

**School of Information Systems
Curtin Business School**

A Semantic Framework for Ontology Usage Analysis

Jamshaid Ashraf

**This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University**

February 2013

DECLARATION

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgement has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Jamshaid Ashraf

Signature:

Date :

Table of Contents

List of Figures	xiv
List of Tables	xix
Abstract	xx
Acknowledgments	xxi
List Of Publications	xxii
Recognition of the work by the Semantic Web Community	xxiv
1 Introduction	1
1.1 Introduction	1
1.2 Semantic Web	3
1.3 Ontologies	6
1.3.1 Different stages in the Ontology Lifecycle	7
1.3.2 Ontology Engineering	8
1.3.3 Ontology Evaluation	8
1.3.4 Ontology Population	9
1.3.5 Ontology Evolution	9
1.3.6 The missing link between the different stages of an ontology lifecycle	10
1.4 Need for Ontology Usage	11
1.4.1 Analysing the use of ontologies	11
1.4.2 Encouraging the reuse of Terminological Knowledge	12
1.5 Thesis contribution	13
1.6 Scope of the Thesis	14
1.7 Significance of the Thesis	14
1.8 Structure of the Thesis	16
1.9 Conclusion	17
2 Literature Review	18

2.1	Introduction	18
2.2	Ontology Focused Work	19
2.2.1	Ontology Development Methodologies and Processes	20
2.2.1.1	Methodologies and Methods for Building Ontologies from Scratch	22
2.2.1.2	Cooperative and Distributed Approaches for Ontology Building	26
2.2.1.3	Ontology Development Approaches using Web2.0 features	29
2.2.1.4	Summarizing Ontology Development Methodologies and Processes.	30
2.2.2	Ontology Lifecycle	30
2.2.2.1	Summarizing the Ontology Lifecycle.	33
2.2.3	Ontology Evaluation Frameworks	34
2.3	Semantic Web (RDF) Data Focused	37
2.3.1	Empirical Analysis of RDF Data on the Web	38
2.3.2	Empirical Analysis of Ontologies and Vocabularies in RDF Data	41
2.4	Critical Evaluation of the existing work on Analysing Ontology Usage	43
2.4.1	Lack of a definition to describe Ontology Usage Analysis	45
2.4.2	Ontology Usage has not been positioned as an area in the Ontology Engineering Lifecycle	46
2.4.3	There is no methodological approach for Ontology Usage Analysis	47
2.4.4	Lack of methods and techniques to measure Ontology Usage Analysis	48
2.4.5	Lack of a model to conceptually represent Ontology Usage Analysis and make it accessible to others	48
2.5	Conclusion	49
3	Problem Definition	50
3.1	Introduction	50
3.2	Key concepts	51
3.3	Problem Definition	57
3.4	Research Issues	65
3.5	Research approach to Problem Solving	67
3.6	Conclusion	69
4	Solution Overview	70
4.1	Introduction	70
4.2	Preliminaries and Notation	70

4.3	Defining Ontology Usage Analysis	72
4.3.1	Positioning Ontology Usage Analysis in Ontology Engineering Lifecycle	74
4.3.1.1	Ontology Usage Analysis (OUA) vs. Ontology Evaluation	75
4.3.1.2	Ontology Usage Analysis vs. Ontology Evolution	75
4.4	Solution overview: Methodological Approach for OUA	78
4.4.1	Identification Phase	78
4.4.2	Investigation Phase	79
4.4.3	Representation Phase	79
4.4.4	Utilization Phase	80
4.5	Ontology Usage Analysis Framework (OUSAF)	80
4.5.1	Identification Phase: Identification of candidate ontologies	81
4.5.1.1	Usage Analysis of a specific Domain Ontology	81
4.5.1.2	Identify and analyse candidate ontologies from dataset:	81
4.5.2	Investigation Phase: Investigating the Ontology Usage	83
4.5.2.1	Empirical Analysis	84
4.5.2.2	Quantitative Analysis	84
4.5.3	Representation Phase: Representing Usage Analysis	86
4.5.4	Utilization Phase: Utilizing Usage Analysis results	87
4.6	Conclusion	87
5	Identification Phase : Ontology Usage Network Analysis Framework (OUN-AF)	88
5.1	Introduction	88
5.2	Social Network Analysis	90
5.2.1	Explicit Relationships	91
5.2.2	Interaction	91
5.2.3	Affiliation	92
5.3	Rationale of using SNA in Ontology Identification	93
5.3.1	Related work on the use of SNA in Semantic Web	95
5.4	Key concepts of SNA relevant to Ontology Identification phase	97
5.4.1	Key Terms and their definitions	97
5.4.2	Types of Networks	100
5.4.2.1	Labelled and Directed Networks	100
5.4.2.2	Labelled, weighted and bi-directional network	101
5.4.2.3	Hypergraphs	101
5.4.2.4	Bipartite (2-mode) Graph	102

5.5	Affiliation Network	102
5.5.1	Representing Affiliation Network	104
5.5.2	Projecting Affiliation Network	105
5.6	Ontology Usage Network Analysis Framework (OUN-AF)	106
5.6.1	Input phase	107
5.6.2	Computation phase	108
5.6.2.1	Ontology Usage Network (OUN)	108
5.6.2.2	Projection	110
5.6.3	Analysis phase	110
5.6.3.1	Analysing (Two-mode) OUN	110
5.6.3.2	Analysing Projected One-mode Network	111
5.6.4	Sequence of OUN-AF activities	112
5.7	Metrics for Ontology Identification	115
5.7.1	Ontology Usage Distribution (OUD)	115
5.7.2	Semanticity	115
5.7.3	Betweenness and Closeness centrality	116
5.7.4	Cohesive Subgroups	117
5.8	Analysis of Ontology Usage Network	117
5.8.1	Dataset and its characteristics	117
5.8.2	Analysing Ontology Usage Distribution (OUD)	120
5.8.3	Analysing Semanticity	122
5.8.4	Formation of Ontology Co-Usability	124
5.8.5	Analysing Betweenness and Closeness	126
5.8.6	Analysing Cohesive Subgroups	129
5.9	Ontology Identification Evaluation	132
5.9.1	Scenario 1: Ontologies and Data Publishers	132
5.9.2	Scenario 2: Ontologies Co-usability	133
5.10	Conclusion	133
6	Investigation Phase: Empirical Analysis of Domain Ontology Usage	
	(EMP-AF)	135
6.1	Introduction	135
6.2	Different ways of Analysing Domain Ontologies	137
6.2.1	Empirical Analysis of Domain Ontology Usage	138
6.2.2	Quantitative Analysis of Domain Ontology Usage	140
6.3	EMPIRICAL Analysis Framework (EMP-AF)	140
6.3.1	Data Collection phase	141
6.3.2	Aspects Analysis phase	142

6.3.3	Sequence of EMP-AF activities	143
6.4	Metrics for EMP-AF	145
6.4.1	Preliminaries	145
6.4.2	Schema Link Graph (SLG)	146
6.4.3	Concept Usage Template (CUT)	147
6.4.3.1	Concept Instantiation	147
6.4.3.2	Vocabs	148
6.4.3.3	Object Property Usage	148
6.4.3.4	Attribute Usage	149
6.4.3.5	Class Usage	149
6.4.3.6	Interlinking	150
6.4.4	Labelling	152
6.4.4.1	Formal Labels	152
6.4.4.2	Domain Labels	153
6.4.5	Knowledge Patterns (Traversal Path)	154
6.4.5.1	Unique paths	154
6.4.5.2	Average Path Length	154
6.4.5.3	Max path length	155
6.4.5.4	Path steps	155
6.5	Case Study: Empirically Analysing Domain Ontology Usage	156
6.5.1	GoodRelations as a domain ontology	156
6.5.2	Conceptual Schema and Pivot Concepts	157
6.5.2.1	Business Entity	158
6.5.2.2	Offering	158
6.5.2.3	Product or Service	159
6.6	Data Collection: Hybrid crawler and Dataset	159
6.6.1	Hybrid Crawler	159
6.6.2	Dataset characteristic	161
6.6.3	Data providers landscape	163
6.6.3.1	Large Size Retailers	163
6.6.3.2	Web shops	163
6.6.3.3	Data Service providers (Data spaces)	163
6.6.4	Use of different Namespace Analysis in GRDS	164
6.7	Empirical Analysis of Domain Ontology Usage	165
6.7.1	Preprocessing	165
6.7.2	Analysing the Schema Link Graph (SLG)	167
6.7.3	Analysing the Concept Usage Template (CUT) and Labelling	168
6.7.3.1	gr:BusinessEntity Analysis	168

6.7.3.2	gr:Offering Analysis	170
6.7.3.3	gr:ProductOrService Analysis	171
6.7.4	Analysing Knowledge Patterns (Traversal Path)	173
6.8	Empirical Analysis Evaluation	175
6.8.1	Scenario 1: Application developers need to know how a given ontology is being used.	175
6.8.1.1	Case 1 : What terminological knowledge is available for application consumption?	175
6.8.1.2	Case 2 : What common data and knowledge patterns are available?	176
6.8.1.3	Case 3 : How are entities being annotated or textually described?	176
6.8.2	Scenario 2: Data publishers need to know what is being used to semantically describe domain-specific information.	177
6.8.2.1	Case 1 : How is a company (or business) being described and what attributes are being used?	177
6.8.2.2	Case 2 : What other entities are a company (entity) linked to?	178
6.9	Conclusion	178
7	Investigation Phase: Quantitative Analysis of Domain Ontology	
	Usage (QUA-AF)	179
7.1	Introduction	179
7.2	QUAntitative Analysis Framework (QUA-AF)	181
7.2.1	Data Collection phase	182
7.2.1.1	Ontology Repository	183
7.2.1.2	Semantic Web (RDF) data	183
7.2.1.3	Semantic Markup Repository	183
7.2.2	Computation Phase	184
7.2.2.1	Ontology Richness Module	184
7.2.2.2	Ontology Usage Module	184
7.2.2.3	Incentive Module	185
7.2.2.4	Ranking different measures	185
7.2.3	Application Phase	186
7.2.4	Sequence of QUA-AF activities	186
7.3	Metrics for quantifying dimensions for OUA in QUA-AF	189
7.3.1	Measuring Ontology Richness	189
7.3.1.1	Concept Richness (CR)	189

7.3.1.2	Relationship Value (RV)	191
7.3.1.3	Attribute Value (AV)	193
7.3.2	Measuring Ontology Usage	193
7.3.2.1	Concept Usage (CU)	193
7.3.2.2	Relationship Usage (RU)	195
7.3.2.3	Attribute Usage (AU)	196
7.3.3	Measuring Incentive	196
7.3.4	Ranking based on Usage Analysis	198
7.4	Case study: Quantitative Analysis of Domain Ontology Usage	199
7.4.1	Dataset and Ontology Identification	200
7.5	GoodRelations Ontology Usage Analysis	201
7.5.1	Computation	201
7.5.2	Observations	202
7.5.2.1	Usage related observations	203
7.6	FOAF Ontology Usage Analysis	207
7.6.1	Observation	207
7.7	Quantitative Analysis Evaluation	209
7.7.1	Requirement 1: Identify the ontologies application to an eCommerce website	209
7.7.2	Requirement 2: Identify the ontological terms to be used	210
7.7.3	Requirement 3 : A summarized view on the prevalent use of ontologies in a given application area	211
7.8	Conclusion	211

8 Representation Phase: Ontology Usage Ontology

	(U Ontology)	213
8.1	Introduction	213
8.2	Different Aspects to be Considered while Representing Ontology Usage	214
8.2.0.1	Different type of user	214
8.2.0.2	Structure and format	215
8.3	Methodology Adopted for U Ontology	216
8.3.1	Specification phase	218
8.3.2	Conceptualization phase	219
8.3.3	Formalization phase	220
8.3.3.1	Implementation phase	220
8.3.4	Steps involved in the development of U Ontology	221
8.3.4.1	Step 1 : Develop Motivation Scenario	221
8.3.4.2	Step 2 : Develop Competency Questions	222

8.3.4.3	Step 3 : Develop Controlled Vocabulary	222
8.3.4.4	Step 4 : Identify Existing Ontologies	222
8.3.4.5	Step 5 : Develop Conceptual Model	223
8.3.4.6	Step 6 : Formalize Conceptual Model	223
8.3.4.7	Step 7 : Integrate with other ontologies	224
8.3.4.8	Step 8 : Ontology encoding (implementation)	224
8.4	Specification phase: Motivation Scenarios and Competency Questions	224
8.4.1	Motivational Scenarios to capture the requirements of users	225
8.4.1.1	Scenario 1 : Ontology Developers (owners)	225
8.4.1.2	Scenario 2 : Application developers	225
8.4.1.3	Scenario 3 : Data publishers	226
8.4.1.4	Scenario 4 : Semantic Web practitioners/users	226
8.4.2	Competency Questions to capture the scope of representation	227
8.5	Conceptualization phase: Controlled Vocabulary, Existing Ontologies and Conceptual Model	230
8.5.1	Controlled Vocabulary to identify the terminological knowledge	230
8.5.2	Identify existing ontologies for reuse	238
8.5.2.1	Ontology Metadata Vocabulary	239
8.5.2.2	Ontology Application Framework	239
8.5.3	Conceptual Model for U Ontology	240
8.5.3.1	Ontology Usage Analysis sub-model	240
8.5.3.2	Concept Usage sub-model	242
8.5.3.3	KnowledgePattern sub-model	243
8.6	Formalization Phase: Ontology Formalization and Integration	244
8.6.1	Formalization of Conceptual Model	244
8.6.1.1	Concepts to represent Analysis Metadata	246
8.6.1.2	Core Concepts to represent Ontology Usage Analysis	247
8.6.1.3	Concepts to represent Knowledge Patterns	248
8.6.2	Integration with other ontologies	249
8.7	Implementation Phase: Ontology Implementation	252
8.8	Conclusion	253
9	Utilization Phase: Utilization of OUSAF Framework	255
9.1	Introduction	255
9.2	Approach to demonstrate the utilization of the OUSAF framework.	256
9.2.1	Types of Users	257
9.2.2	Contributions and Criteria for Utilization Analysis	258

9.2.3	Components and Sequence of activities involved in Analysing the Utilization of each contribution of the OUSAF framework	259
9.3	Dataset for demonstrating the utilization of OUSAF	261
9.3.1	Data Requirements	261
9.3.2	Data Collection Strategy	263
9.3.3	Data Collection Process	264
9.3.3.1	Seed URL Builder	264
9.3.3.2	Semantic Document Downloader	265
9.3.3.3	Snippet Extractor	265
9.3.3.4	Contextualizer	266
9.3.3.5	Loader	266
9.3.4	Crawling	266
9.3.5	Dataset statistics	268
9.4	Utilization of the Identification Framework (OUN-AF)	269
9.4.1	Aim	269
9.4.2	Flow of Activities	269
9.4.3	Use Cases	270
9.4.4	Utilization Analysis	270
9.4.4.1	Req. 1: What is the level of usage of a given ontology? .	270
9.4.4.2	Req. 2: Is the given ontology being used alone or with other ontologies and if yes, what are these?	272
9.4.4.3	Req. 3: What ontologies are being used in a given domain?	273
9.4.4.4	Req. 4 : What data sources are using a given ontology to publish their information?	274
9.4.4.5	Req. 5: What cohesive groups of ontologies have similar usage?	275
9.4.5	Discussion of Findings	277
9.5	Utilization of the Empirical Analysis Framework (EMP-AF)	277
9.5.1	Aim	277
9.5.2	Flow of Activities	278
9.5.3	Use Cases	278
9.5.4	Utilization Analysis	279
9.5.4.1	Req. 1: What is the adoption level of a given ontology? .	279
9.5.4.2	Req. 2: How are the entities of a given concept described?281	
9.5.4.3	Req. 3: How are entities textually described?	283
9.5.4.4	Req. 4: What knowledge patterns are available in the dataset?	284

9.5.5	Discussion on Findings	285
9.6	Utilization of the Quantitative Analysis Framework (QUA-AF)	285
9.6.1	Aim	285
9.6.2	Flow of Activities	286
9.6.3	Use Cases	286
9.6.4	Utilization Analysis	287
9.6.4.1	Req. 1: What is the richness value of the concepts in a given ontology?	287
9.6.4.2	Req. 2: Display the ontology terms based on their usage ranking?	288
9.6.4.3	Req. 3: List the terms that are being recognized by search engines?	290
9.6.4.4	Req. 4: What ontologies are being used in a given application area?	291
9.6.5	Discussion on Findings	293
9.7	Benefits of the Utilization phase	293
9.7.1	Benefits from an Ontology Developer’s perspective	294
9.7.2	Benefits from a Data Consumer’s perspective	294
9.7.3	Benefits from a Data Publisher’s perspective	295
9.8	Comparison of the output from the OUSAF framework with existing approaches in the literature	295
9.9	Conclusion	297
10	Evaluation of U Ontology	298
10.1	Introduction	298
10.2	Methodology for Ontology Evaluation	298
10.2.1	Criteria for Ontology Evaluation	300
10.2.2	Aspects to be analysed for Ontology Evaluation	301
10.2.3	Metrics to quantify the evaluation findings	302
10.3	U Ontology evaluation: Vocabulary aspect	303
10.3.1	Method 1 : Check used protocols	303
10.3.2	Method 2 : Check response codes	304
10.3.3	Method 3 : Look up Names	306
10.3.4	Method 4 : Check Naming conventions	308
10.3.5	Method 5 : Metrics of Ontology reuse	309
10.3.6	Method 6 : Check name declaration	310
10.3.7	Method 7: Check Literals and data type	311
10.3.8	Method 8 : Check Language tag	311

10.3.9	Method 9 : Check labels and comments	312
10.3.10	Method 10 : Check for superfluous blank nodes	313
10.4	U Ontology Evaluation : Syntax aspect	314
10.4.1	Method 11 : Validating against an XML Schema	314
10.5	U Ontology Evaluation : Structural aspect	315
10.5.1	Method 12 : Ontology Complexity	315
10.5.2	Method 13 : Searching for Anti-Patterns	317
10.5.3	Method 14 : OntoClean meta-property check	320
10.6	U Ontology evaluation : Semantics aspect	321
10.6.1	Method 15 : Ensuring a stable class hierarchy	321
10.6.2	Method 16 : Measuring language completeness	322
10.7	U Ontology evaluation : Representation aspect	323
10.7.1	Method 17 : Explicitness of the subsumption hierarchy	323
10.7.2	Method 18 : Explicit terminology ratio	324
10.8	U Ontology evaluation : Context aspect	325
10.8.1	Method 19 : Checking competency questions against results	325
10.8.2	Method 20 : Checking competency questions with constraints	326
10.8.3	Method 22 : Increasing expressivity	327
10.8.4	Method 23 : Inconsistency checks with rules	328
10.9	Summary of U Ontology Evaluation	328
10.10	Conclusion	330
11	Recapitulation and Future Work	331
11.1	Introduction	331
11.2	Recapitulation	332
11.3	Contribution of the Thesis	334
11.4	Future work	339
11.4.1	Expand the dataset and Extend to other Application domains in order to provide (near to) real time Usage Analysis	339
11.4.2	Publish the Ontology Usage Analysis in the form of Ontology Usage Catalogue	340
11.4.3	Explore other Dimensions and Aspects required for Measuring Ontology Usage and provide support for Reasoning over the collected dataset	340
11.4.4	Explore and Incorporate other approaches for Measuring Incentives	341
11.4.5	Explore further ontology evaluation methods to validate U Ontology	341

11.5 Conclusion	342
References	342
Appendix A U Ontology Listing	372
Appendix B Best PhD Symposium Paper Award	384
Appendix C Selected Publications	386

List of Figures

FIGURE 1.1: A simple picture of Web Evolution (Ding, 2007)	2
FIGURE 1.2: The Semantic Web Stack (SW Layer Cake)	5
FIGURE 1.3: Detailed description of ontology definition (Studer et al., 1998)	7
FIGURE 1.4: Two main stages in Ontology lifecycle: Development and In-Use stages	8
FIGURE 1.5: Ontology Lifecycle with a feedback loop based on Ontology Usage.	10
FIGURE 2.1: Ontology Engineering components.	20
FIGURE 2.2: Roles and functions in distributed ontology engineering (cf. (Vrandecic et al., 2005)).	27
FIGURE 2.3: The software architecture. Each box represents a software module, each circled unit is a data/knowledge repository and each arrow represents the call of a program functionality (Euzenat, 1996)	27
FIGURE 2.4: Scenarios for Building Ontology Networks (Gomez-Perez and Suarez-Figueroa, 2009)	28
FIGURE 2.5: A high level view of Ontology lifecycle	31
FIGURE 2.6: Abstract Data Lifecycle Model (Möller, 2012)	32
FIGURE 2.7: Lifecycle model (c.f. (Tran et al., 2008))	33
FIGURE 2.8: Ontology Lifecycle model with Ontology Evaluation.	35
FIGURE 3.1: PingTheSemanticWeb.com Index of vocabulary and ontology usage	57
FIGURE 3.2: Linking Open Data cloud diagram by Richard Cyganiak and Anja Jentzsch. http://lodcloud.net ; retr. 10/09/2011	59
FIGURE 3.3: Set of activities performed and their levels according to science and engineering-based methodology	68
FIGURE 3.4: Set of activities and levels mapped with thesis chapters.	69
FIGURE 4.1: Phases in Ontology Usage Analysis	78
FIGURE 4.2: Ontology Usage Analysis Framework and its components with the process flow	82

FIGURE 5.1: Zachary club network. The links between two nodes represents explicit relationship and nodes are split into two groups one in round white and other in round gray.	91
FIGURE 5.2: Protein to Protein interaction network.	92
FIGURE 5.3: Affiliation Network (two-mode graph) consisting of 10 US computer software firms (red circles) and 54 strategic alliances (blue squares)	93
FIGURE 5.4: Power-law distribution	99
FIGURE 5.5: A simple example of a network (with eight nodes and edges) .	100
FIGURE 5.6: Examples of different types of networks: (a) network with labelled nodes and directed edges; (b) network with labelled nodes, weighted and bidirectional edges.	101
FIGURE 5.7: Hypergraph with three hyperedges	102
FIGURE 5.8: An example of an author-paper affiliation network	105
FIGURE 5.9: Example of projection: (a) authors co-affiliation network, (b) paper's co-authorship network	105
FIGURE 5.10: Ontology Usage Network Analysis Framework (OUN-AF) and its phases.	107
FIGURE 5.11: Ontology Usage Network (affiliation network with one set of nodes representing ontologies and other set of nodes representing data sources).	109
FIGURE 5.12: Flow of activities in OUN-AF.	114
FIGURE 5.13: Ontology Usage Affiliation Network (bipartite graph).	118
FIGURE 5.14: List of ontologies with their prefixes.	119
FIGURE 5.15: Degree distribution of ontology usage (data sources per ontology).	121
FIGURE 5.16: Degree distribution of semanticity (Ontologies per Data source).	123
FIGURE 5.17: Data source collaboration network.	124
FIGURE 5.18: Ontology Co-usability network.	126
FIGURE 5.19: Betweenness centrality of Ontology Co-Usage network.	128
FIGURE 5.20: Closeness centrality of Ontology Co-Usage network.	129
FIGURE 5.21: Stacking of k-cores of Ontology Co-Usage network.	131
FIGURE 6.1: Two different ways to analyse ontology usage.	138
FIGURE 6.2: Empirical Analysis Framework (EMP-AF).	141
FIGURE 6.3: Flow of activities in EMP-AF.	144
FIGURE 6.4: (a) Sample RDF graph from the dataset with blank nodes and (b) the corresponding Schema Link Graph.	146
FIGURE 6.5: Sample RDF code for discussion	151

FIGURE 6.6: Traversal paths	155
FIGURE 6.7: Path steps and their strength	156
FIGURE 6.8: GoodRelations Ontology Adopters (http://wiki.goodrelations-vocabulary.org/References ; Accessed 15 Sept, 2012)	157
FIGURE 6.9: GoodRelations Popularity reported by PingTheSemanticWeb.com (http://pingthesemanticweb.com/stats/namespaces.php ; Accessed 12 Sept, 2012)	157
FIGURE 6.10: Schemata diagram of Hybrid Crawler.	161
FIGURE 6.11: List of Data sources included in GRDS.	162
FIGURE 6.12: Schema Link Graph (SLG) in GRDS	167
FIGURE 7.1: Usage Model of Simmons (2005)	180
FIGURE 7.2: Quantitative Analysis Framework for Ontology Usage Analysis	182
FIGURE 7.3: Flow of activities in QUA-AF	188
FIGURE 7.4: Sample Ontology and its instantiation to explain the metrics defined in QUA-AF to measure richness and usage.	191
FIGURE 7.5: SPARQL query to compute CU metrics value	202
FIGURE 8.1: Different ontology development methodologies and their relationship.	217
FIGURE 8.2: Customized ontology development methodology for the U Ontology.	221
FIGURE 8.3: UML Class diagram for ontology modeling.	223
FIGURE 8.4: Ontology-related information clusters.	238
FIGURE 8.5: Ontology Usage Analysis sub (conceptual) model.	240
FIGURE 8.6: OntologyUsage and related concepts.	241
FIGURE 8.7: Concept Usage sub-model.	242
FIGURE 8.8: KnowledgePattern sub-model.	243
FIGURE 8.9: KnowledgePattern components.	244
FIGURE 8.10: U Ontology Overview (V2.0).	245
FIGURE 8.11: Concepts describing Analysis Metadata.	247
FIGURE 8.12: Concepts describing ontology components (concept, relationship, and attribute) analysis	247
FIGURE 8.13: Concepts to represent Knowledge Patterns in dataset.	249
FIGURE 8.14: Integration of the U Ontology with other ontologies.	250
FIGURE 9.1: Approach for measuring Usability and Adequacy	257
FIGURE 9.2: Components and Sequence of activities involved in analysing the utilization	259
FIGURE 9.3: Components involved in data collection	264

FIGURE 9.4: Seed URL Builder	265
FIGURE 9.5: Basic Flow of the Crawler's Activities	267
FIGURE 9.6: SPARQL query to display the list of ontologies and their usage.	271
FIGURE 9.7: Result of SPARQL query shown in Figure 9.6	271
FIGURE 9.8: SPARQL query to display the names of the ontologies being co-used.	272
FIGURE 9.9: Result of SPARQL query shown in Figure 9.8.	273
FIGURE 9.10: SPARQL query to display the name of the ontologies present in the dataset.	274
FIGURE 9.11: SPARQL query to display the name of the data sources which have used a given ontology.	274
FIGURE 9.12: Result of SPARQL query shown in Figure 9.11.	275
FIGURE 9.13: SPARQL query to extract the k-core value ontologies	276
FIGURE 9.14: Result of SPARQL query shown in Figure 9.13.	276
FIGURE 9.15: SPARQL query to display the terms of the ontology which have usage in the dataset.	279
FIGURE 9.16: Query exploiting RDFS entailment rule (rdfs9)	280
FIGURE 9.17: Result of SPARQL query shown in Figure 9.15.	281
FIGURE 9.18: SPARQL query to display the use of different relationships and attributes.	282
FIGURE 9.19: Result of SPARQL query shown in Figure 9.18.	282
FIGURE 9.20: SPARQL query to access the formal and domain labels used for a concept.	283
FIGURE 9.21: Result of SPARQL query shown in Figure 9.20.	284
FIGURE 9.22: SPARQL query to display the knowledge patterns in the dataset.	284
FIGURE 9.23: SPARQL query to display the concepts and their richness value of a specific ontology	287
FIGURE 9.24: Result of SPARQL query shown in Figure 9.23.	288
FIGURE 9.25: SPARQL query to display the usage of given ontology terms. .	289
FIGURE 9.26: Result of SPARQL query shown in Figure 9.25.	290
FIGURE 9.27: SPARQL query to list the terms that are being recognised by search engines.	291
FIGURE 9.28: Result of SPARQL query shown in Figure 9.27.	291
FIGURE 9.29: SPARQL query to list the ontologies being used in a given application area.	292
FIGURE 9.30: Result of SPARQL query shown in Figure 9.29.	292
FIGURE 10.1: The URI and HTTP protocol used in the U Ontology	304

FIGURE 10.2: Excerpt from U Ontology	306
FIGURE 10.3: SPARQL query to identify an anti-pattern in the U Ontology .	319
FIGURE 10.4: Summary of U Ontology Evaluation	329
FIGURE B.1: Scanned copy of Best PhD Symposium Paper Award	385

List of Tables

TABLE 4.1: Drawing comparison between Ontology Usage Analysis, Ontology Evaluation and Ontology Evolution	77
TABLE 5.1: Affiliation matrix of author-paper affiliation network	104
TABLE 5.2: Distribution of Ontology Usage in data sources	121
TABLE 5.3: Distribution of Semanticity (Ontology used per data source) . . .	122
TABLE 6.1: Sample RDF Graph: CUT of gr:BusinessEntity (ex:cardealer) . .	152
TABLE 6.2: List of vocabularies and their percentage in GRDS	165
TABLE 6.3: CUT of gr:BusinessEntity	169
TABLE 6.4: CUT of gr:Offering	171
TABLE 6.5: CUT of gr:ProductOrService	173
TABLE 6.6: Traversal path of all three pivot concepts	173
TABLE 6.7: Path Steps frequency in Traversal Path	174
TABLE 7.1: List of ontologies identified in the dataset	200
TABLE 7.2: GoodRelations Concepts Usage Analysis and their rank considering richness, usage and incentive measures.	204
TABLE 7.3: GoodRelations ontology terms and their ranking	206
TABLE 7.4: FOAF ontology terms and their ranking	208
TABLE 8.1: U Ontology Controlled Vocabulary	230
TABLE 8.2: Reused Terms	251
TABLE 9.1: Contributions of OUSAF Framework	258
TABLE 9.2: Content Type of HTTP Response	269

Abstract

The Semantic Web envisions a Web where information is accessible and processable by computers as well as humans. Ontologies are the cornerstones for realizing this vision of the Semantic Web by capturing domain knowledge by defining the terms and the relationship between these terms to provide a formal representation of the domain with machine-understandable semantics. Ontologies are used for semantic annotation, data interoperability and knowledge assimilation and dissemination.

In the literature, different approaches have been proposed to build and evolve ontologies, but in addition to these, one more important concept needs to be considered in the ontology lifecycle, that is, its *usage*. Measuring the “usage” of ontologies will help us to effectively and efficiently make use of semantically annotated structured data published on the Web (formalized knowledge published on the Web), improve the state of ontology adoption and reusability, provide a usage-based feedback loop to the ontology maintenance process for a pragmatic conceptual model update, and source information accurately and automatically which can then be utilized in the other different areas of the ontology lifecycle. Ontology Usage Analysis is the area which *evaluates, measures and analyses the use of ontologies on the Web*. However, in spite of its importance, no formal approach is present in the literature which focuses on measuring the use of ontologies on the Web. This is in contrast to the approaches proposed in the literature on the other concepts of the ontology lifecycle, such as ontology development, ontology evaluation and ontology evolution. So, to address this gap, this thesis is an effort in such a direction to assess, analyse and represent the use of ontologies on the Web.

In order to address the problem and realize the abovementioned benefits, an Ontology Usage Analysis Framework (OUSAF) is presented. The OUSAF Framework implements a methodological approach which is comprised of *identification, investigation, representation and utilization* phases. These phases provide a complete solution for usage analysis by allowing users to identify the key ontologies, and investigate, represent and utilize usage analysis results. Various computation components with several methods, techniques, and metrics for each phase are presented and evaluated using the Semantic Web data crawled from the Web. For the dissemination of ontology-usage-related information accessible to machines and humans, The U Ontology is presented to formalize the conceptual model of the ontology usage domain. The evaluation of the framework, solution components, methods, and a formalized conceptual model is presented, indicating the usefulness of the overall proposed solution.

Acknowledgements

First and foremost, I would like to thank Allah Almighty for the immense blessings and strength He bestowed on me throughout my life and particularly during my PhD tenure.

I am indebted to my supervisor, Dr. Omar Khadeer Hussain. He patiently and gracefully provided guidance, encouragement, and advice that were crucial for the completion of this thesis and assisted in each stage of my study. I want to thank other members of my thesis committee, particularly Prof. Tharam Dillon, Dr. Maja Hadzic, and Dr. Farookh Hussain for their support and advice. I would like to thank my colleagues at EU4 with whom I have shared a memorable time and particularly enjoyable lunches with Naeem Janjua and Zia-ur-Rehman.

Special thanks to my friends at DERI, Galway, especially Dr. Sean O'Riain, Richard Cyganiak, and Dr. Michael Hausenblas, who have been generous with their knowledge and gave me the opportunity to work in their Linked Data group. Throughout the six months, the experience and knowledge gained provided the ground for this research.

I gratefully acknowledge the support and motivation I received from Mohammad Amjad Ch. throughout the PhD program. His interest and the cooperation received from the DPS Kuwait and my colleagues have made the journey through these years a comforting experience.

A special thanks to Adil Hammadi and his family for their continuous support and assistance from the beginning of my enrolment in this PhD program to the day this thesis is submitted. Also, I would like to thank Masroor Siddiqui and his family for their prayers and support throughout these years and for visiting us in Perth.

A major part of what I am today is credited to my brother and mentor, Javaid Ashraf, who took the responsibility to educate me since I was in the 7th grade. I would like to thank my sisters, Ghazala Arif and Rizwana Ali, for their care, affection, and prayers.

I am grateful and indebted to my wife Farheen. She has continuously been a source of strength and courage and has done every possible thing to make this a pleasant and comforting journey. To my children, Rahemeen, Ahmad, and Barirah, thanks to all of you for being a wonderful source of joy and pleasure. I am impressed by your level of understanding and support you have demonstrated over the last few months.

Last but certainly not least, I dedicate this thesis to my late parents, Mohammad Ashraf and Zubaida Baigum, who gave the best of themselves to make sure that we have the best life. Without their sacrifices, I would not have made it this far.

List Of Publications arising from this Thesis

Referred Journal Article

1. Jamshaid Ashraf, Omar Khadeer Hussain, and Farookh Khadeer Hussain "A Framework for Measuring Ontology Usage on the Web", The Computer Journal, doi: 10.1093/comjnl/bxs134, 2012. (**ERA Rank A***, **Impact Factor: 0.943**)

Referred Conference Article

2. Jamshaid Ashraf, "A Framework for Ontology Usage Analysis", in Proceedings 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012, pp 813-817. (**Best PhD Symposium Paper Award**)
3. Jamshaid Ashraf, Richard Cyganiak, Sean O'Riain, Maja Hadzic. "Open eBusiness Ontology Usage: Investigating Community Implementation of GoodRelations" In Proceedings of Linked Data on the Web Workshop (LDOW) at World Wide Web (WWW2011), volume 813 of CEUR Workshop Proceedings, pages 111, Hyderabad, India.
4. Jamshaid Ashraf, Maja Hadzic, "Domain Ontology Usage Analysis Framework," Semantics Knowledge and Grid (SKG), Seventh International Conference, Beijing, China, 24-26 Oct., 2011, pp.75-82,
5. Jamshaid Ashraf, Maja Hadzic, "Web Schema Construction Based on Web Ontology Usage Analysis", In, Joint International Semantic Technology Conference, JIST 2011, The Semantic Web, Springer Berlin Heidelberg, 2012, 7185, pp. 376-384
6. Jamshaid Ashraf, Omar Khadeer Hussain, "Integrating Financial Data Using Semantic Web for Improved Visibility", Semantics, Knowledge and Grids (SKG), Eighth International Conference, Beijing, China 22-24 Oct., 2012, pp.265-268,
7. Jamshaid Ashraf, Omar Khadeer Hussain, "Ontology Usage Network Analysis Framework", The 15th Asia-Pacific Web Conference (APWeb), Distributed Processing of Graph, XML and RDF Data track. 4-6 April, 2013, Sydney, Australia (in press)

8. Jamshaid Ashraf, Omar Khadeer Hussain, "Ontology Usage Network Analysis", IEEE/WIC/ACM International Conference on Web Intelligence (WI 2012), 4-7 December 2012, Macau (in press)
9. Jamshaid Ashraf, Maja Hadzic and Hassan Naqvi, "Semantic Web and Enterprises - a Pragmatic Approach", Second Kuwait Conference on e-Services and e-Systems, State of Kuwait, April 5-7, 2011 .

Recognition of the work by the Semantic Web Community

The work arising from this thesis has received the following recognition by the Semantic Web community.

Best PhD Symposium Paper Award

at

**9th Extended Semantic Web Conference (ESWC2012)
May 27-31, 2012, Heraklion, Crete, Greece.**

Chapter 1 - Introduction

1.1 Introduction

In the recent past, the internet has transformed the way we communicate, interact and do business across the globe. Described and dubbed as *information highway*, the internet has provided an unprecedented seamless infrastructure to assimilate and disseminate information at an ease and speed never witnessed by mankind. As a result of this, today 32% of the world's population is using the *internet*¹. Capitalizing on the intrinsic properties of the internet such as simplicity, ubiquity and scalability, Tim Berners-Lee introduced the World Wide Web (Berners-Lee and Fischetti, 1999) as a platform for publishing and consuming information at a universal scale. The World Wide Web (also known as the WWW or *Web*), which without a doubt is one of the most significant computational phenomena to date, has revolutionized information sharing by providing a decentralized information platform, which has enabled and empowered users to be more interactive and participative, turning each user of the Web into a potential publisher (Figure 1.1). Being able to publish information which is accessible to anyone in the world with access to the Web for a low cost has resulted in the proliferation of approximately 50 billion web documents² containing information on a variety of topics, creating a huge amount of diversified information commonly known as Big Data.

As a consequence, we are witnessing an incessant rise in user-generated content that is padded with **metadata** to provide additional (syntactical and structural) information about the content, such as content ownership, provenance detail, content categorization and labelling. This stage of the evolution of the Web, is termed Web 2.0 in the literature (O'Reilly, 2005) which is described as *a concept that takes the internet as a platform for information sharing, interoperability, user-centered design, and collaboration on the Web (Figure 1.1)*. With the ability to interact and participate

¹<http://www.internetworldstats.com/stats.htm>; retr. 25/5/2012

²<http://www.worldwidewebsite.com/>; retr. 29/5/2012

in content generation, Web2.0 has provided the necessary techniques (Vossen and Hagemann, 2007) and approaches (such as Web APIs, mash-ups, blogging softwares, tagging) to link documents with users (whether publisher or consumer) by adding meta-information to user-generated content. One of the major contributions of this evolution is the publication of metadata (describing the content and linking it with users) which, in fact, was the early emergence of structured data on the Web, paving the way for the next possible evolution stages.

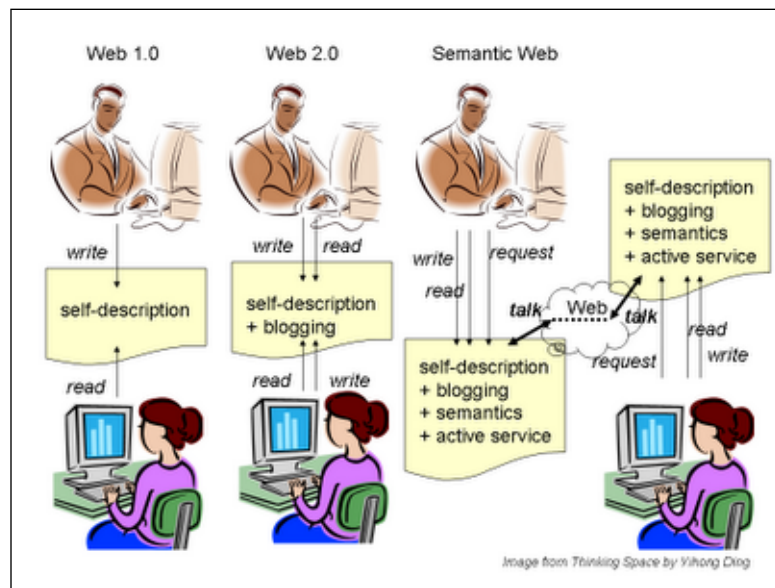


Figure 1.1: A simple picture of Web Evolution (Ding, 2007)

While having such meta-information is useful, its full potential can only be realized if we are able to retrieve the required information off the Web to consume it for our individual or collective needs. While it sounds possible, in reality, it is a daunting challenge to retrieve accurate information when machines do not have any cue to understand the content and structure of web documents. Search engines such as Google and Yahoo! have been working hard on processing and sift through unstructured web documents to classify and index them. This pre-processing of documents, although helping search engines find and return a prioritized list of query-relevant documents (Hogan, 2011b), it falls short in providing answers to specific queries, which is what is needed; *give me what i want when i want it*. In pre-processing, extensive engineering and algorithmic effort (Page et al., 1999; Cooley et al., 1997) has been exerted to understand the information and provide relevant and useful information to users. But this useful information has been limited to only returning prioritized list of relevant documents because presently, the information represented in web documents does not contain necessary metadata needed for machines to understand what the content means.

To address these limitations, after realizing the potential of having structured data available on the Web (Atzeni et al., 1997), there was a push toward developing more sophisticated approaches to access information across the Web with improved accuracy. So, search engines applied information processing techniques to go beyond the keyword-based search and provide support for more complex and adequate queries to allow users access to more precise information. However, the quest for providing answers to complex queries, such as ‘finding the doctors in a city specializing in mental health’ highlighted the need for a more granular and structured representation of information at the data level. The representation of information at the data level on the Web meant that everyone should be able to access, process and interpret the information in a consistent and coherent manner.

To address these challenges and take the Web to its initial envisaged design³ Tim Berners-Lee and colleagues (T. B. Lee and Lassila., 2001) proposed the Semantic Web vision in 1998, which is described as:

I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web the content, links, and transactions between people and computers. A ‘Semantic Web’, which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The ‘intelligent agents’ people have touted for ages will finally materialize.

This vision of the Semantic Web (as shown in Figure 1.1) is to transform the present *web-of-document* into a *web-of-data* where the Web forms a global space for seamless knowledge integration. This global space provides the mechanism to start describing tangible and non-tangible entities such as *people, software modules, projects, concepts, documents*, etc., on the Web. In the next section, Semantic Web, its core technology stack and Linked Data principles are described.

1.2 Semantic Web

The Semantic Web (also dubbed Web 3.0, the Linked Data Web, and the Web of Data) represents the on-going major evolution of the Web in the form of transforming *data* into *meaning*. Such transformation enables data to be linked from a source to any other source and to be understood by computers so that they can perform increasingly

³<http://www.w3.org/History/1989/proposal.html>; retr. 29/5/2012

sophisticated tasks on their behalf. These sophisticated tasks require a knowledge processing capability to realize different applications that come under the rubric of searching, information interoperability, knowledge integration and information retrieval. In order to embrace this major evolution (of the Web) to realize the Semantic Web vision (Berners-Lee, 1998a), the Semantic Web community has taken steps to standardize the underlying foundational components to make them conformant with the original Web architecture (Berners-Lee, 1998b). The guiding principles considered while standardizing information representation at a *syntactic* and *semantic* level are as follows:

- Resources are identified using the Unique Resource Identifier (URI) to make them accessible over the Web
- Resources are described using standard format to make their access and reference consistent across different (consuming) applications.
- Resources are represented using standard data model which is flexible and compatible with Web architecture.
- Resources are semantically described to allow aggregation and combination of data drawn from different resources.

The Resource Description Framework (RDF) (Klyne and Carroll, 2004) is the W3C standard for the representation of data and knowledge on the Web and forms a foundational data model of the Semantic Web, as depicted in Figure 1.2. Figure 1.2, known as the Semantic Web layer cake, shows different layers with their respective roles and proposed technological (standard) components. On a high level, RDF provides the means to connect resources (things, data, documents, abstract idea, etc.) in a structured and meaningful way. Technically, RDF is a framework designed to create statements about resources in the form of subject-predicate-object expressions called RDF triples. RDFS (Resource Description Framework Schema) is the most basic schema language which provides declarative schemata whose semantics are defined within RDF Schema (Brickley and Guha, 2004). RDFS extends RDF vocabulary to allow the description of taxonomies of classes and properties to develop lightweight vocabularies. While RDF provides a standard structured data model and RDFS declarative schema, the OWL (Web Ontology Language) (Dean et al., 2002) provides a highly declarative expressive language to formally conceptualise the knowledge of a given domain. OWL extends RDF and RDFS, its primary aim being to bring the expressive and reasoning power of description logic to the Semantic Web. In order to query information that is semantically described and structured using an RDF data

model, W3C provides SPARQL as a standard query language for RDF data. It contains the SPARQL protocol and RDF query language to allow users and applications to write queries and to consume the results of queries across distributed sources of information (knowledge bases).

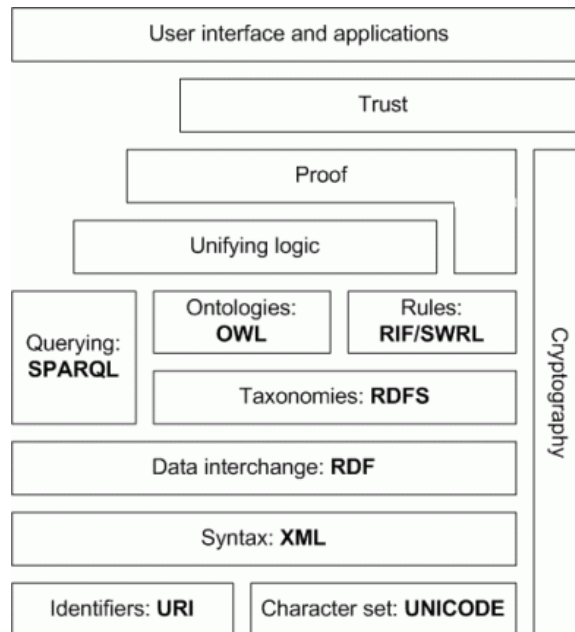


Figure 1.2: The Semantic Web Stack (SW Layer Cake)

While the previously mentioned technological components, discussed in aforementioned paragraph, provide the standards to implement the Semantic Web vision, they do not provide any guidelines to promote the grass-roots adoption of these standards. To address this issue and accelerate the adoption, Tim Berners-Lee, along with the Semantic Web community, introduced **Linked Data** (Bizer et al., 2009) principles to facilitate the publishing and interlinking processes involved in generating semantically rich structured data over the Web. The four Linked Data principles (Berners-Lee, 2006) are as follows:

1. Use URIs as the names for the things
2. Use HTTP URI so that names can be looked up to allow dereferencability
3. Upon look up, return useful information
4. Include links by using URIs which links to other related remote documents

To reap the potential benefits offered by the Semantic Web, many domain-specific industries and their major players, researchers, practitioners and governments, have started adopting the Linked Data principles to disseminate information in a

machine-interpretable way. Notable examples are: different governments⁴ entities (Ding et al., 2010a) such as UK⁵, USA⁶, Australia⁷; different corporations such as the BBC (Kobilarov et al., 2009b), New York Times (Sandhaus, 2010), Thompson Reuters (Kobilarov et al., 2009a), Freebase (Bollacker et al., 2007), Volkswagen⁸, BestBuy (Breslin et al., 2010); community-driven Linked Open Data (LOD2) project⁹ and DBpedia (Auer et al., 2007); biomedical and health-care data sets such as DrugBank¹⁰, UniPort¹¹, LinkedCT¹², PubMed¹³; and several other datasets¹⁴.

However, for machines to interpret information in a common way, distributed ontologies are used to provide machine-processable meta-information enabling automatic information sourcing, retrieval and interlinking. In the next section, ontologies are discussed in detail.

1.3 Ontologies

Ontologies are the main component of the Semantic Web vision as they provide the semantics for the RDF data; that is, transforming data into meaning. In the literature, ontologies are defined by Gruber (1993) as a *formal specification of conceptualization*. Ontologies are viewed as a shared and common understanding of the domain that can be communicated between people and heterogeneous distributed application systems, as shown in Figure 1.3 (depiction appeared in (González, 2005)). Thus, they specify a machine readable vocabulary in computer systems, which is then used to infer and integrate knowledge, based on the semantics they describe.

Ontologies, which are comprised of concepts, relationships, individuals, and axioms, are constructed to formally conceptualise consensual (shared) knowledge about a particular domain. These components of ontologies are identified by Uniform Resource Identifiers (URI) (Berners-Lee et al., 1998) to offer a global naming

⁴To access the updated and extended list of countries participating in the Open Data initiative, visit <http://logd.tw.rpi.edu/> and to obtain the initial analysis visit http://logd.tw.rpi.edu/iogds_data_analytics; retr. 6/7/2012

⁵<http://data.gov.uk/>; retr. 17/06/2012.

⁶<http://www.data.gov/>; retr. 02/5/2012

⁷data.gov.au; retr. 12/9/2012

⁸<http://www.w3.org/2001/sw/sweo/public/UseCases/Volkswagen/>

⁹<http://lod2.eu/WikiArticle/Project.html>

¹⁰<http://www.drugbank.ca/>

¹¹<http://www.uniport.org>

¹²<http://www.linkedct.org>

¹³<http://www.ncbi.nlm.nih.gov/entrez/query.fcgi>

¹⁴Two of the resources which maintain statistics on the different datasets published by following Linked Data principles are: <http://stats.lod2.eu/rdfdocs> (retr. 6/7/2012) and <http://thedatahub.org/group/lodcloud> (retr. 6/7/2012)

scheme. Data publishers use these URIs to describe the information in order to promote consistent and coherent semantic interoperability between users, systems and applications. To reap the benefits of the Semantic Web, several domain ontologies have been developed to describe the information pertaining to different domains such as Healthcare and Life Science (HCLS) (d'Aquin and Noy, 2012), governments¹⁵, social spaces (Breslin et al., 2006; Caire and van der Torre, 2010), libraries (Gradmann, 2005), entertainment (Raimond et al., 2007), financial services (Garcia and Gil, 2009) and eCommerce (Hepp, 2008).

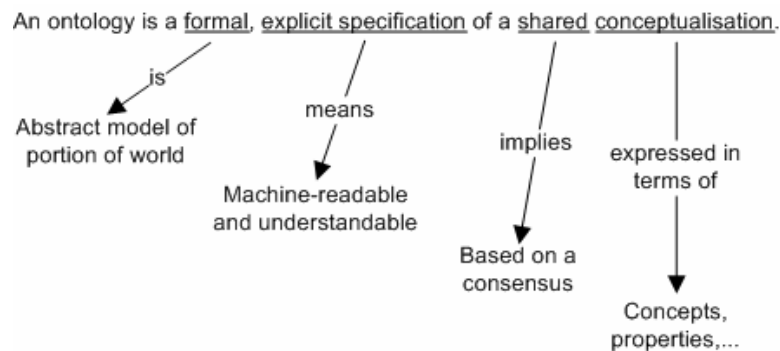


Figure 1.3: Detailed description of ontology definition (Studer et al., 1998)

As is the case with any information system or product, ontologies being the end product go through different stages of building before they can be used. This is discussed further in the next subsection.

1.3.1 Different stages in the Ontology Lifecycle

From a broader and wider perspective, ontologies go through two main stages in their lifecycle, namely *the engineering stage* and *the usage stage*, as shown in Figure 1.4. The engineering stage encompasses the processes and activities involved in the construction of ontologies while the usage stage represents the phase in which ontologies are deployed and used in the real world. The engineering stage (which is also referred to as the development stage in this thesis) deals with *the knowledge meta-process* (Staab et al., 2001) and focuses on knowledge identification which includes all the relevant activities involved in the construction of ontologies such as design, implementation, evaluation and evolution of ontology (left portion of Figure 1.4). The usage stage (which is also referred to as the in-use stage in this thesis), deals with knowledge creation which includes ontology population and the

¹⁵<http://oegov.org/> & <http://dvcs.w3.org/hg/gld/raw-file/default/dcat/index.html>; retr. 12/7/2012

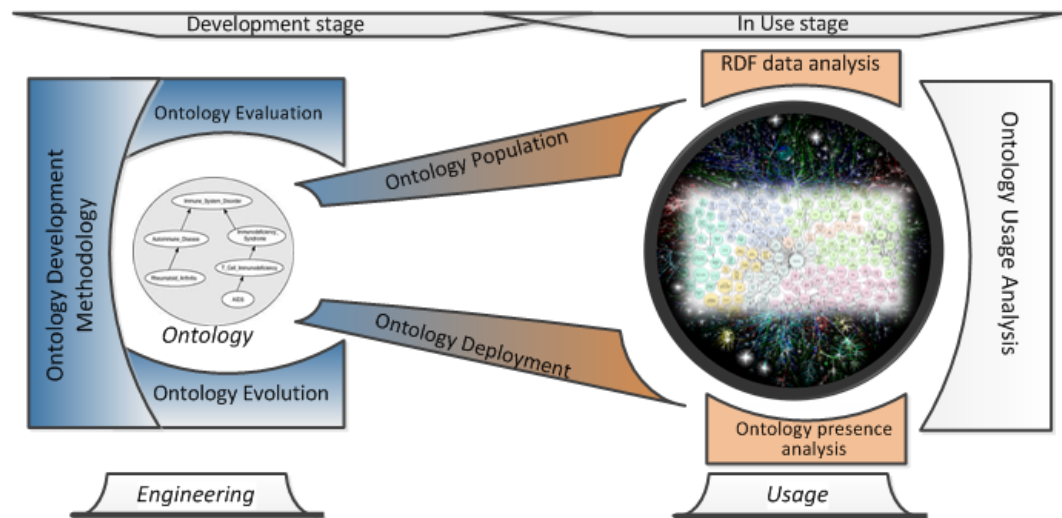


Figure 1.4: Two main stages in Ontology lifecycle: Development and In-Use stages

usage of the ontology (right portion of the Figure 1.4). Each stage comprises many different steps, as explained in the next subsections.

1.3.2 Ontology Engineering

Ontological Engineering refers to the set of activities that concern the ontological development process as well as the methodologies, tools and languages required for building ontologies (Corcho et al., 2007).

In the literature, numerous development methodologies focusing on different aspects have been proposed. For example, Uschold and Kings methodology (Uschold and King, 1995), METHONTOLOGY (Asunción Gómez-Pérez et al., 1996), and On-To-Knowledge (OTK) (Sure, 2002) methodologies are proposed to assist ontology developers in developing new ontologies from scratch. KACTUS (Schreiber et al., 1995) and the Integration-Oriented methodology (Leung et al., 2011) enables ontology engineers reuse existing ontologies to develop new ontologies, and CO4 (Euzenat, 1996) and NeOn Methodology (Presutti et al., 2008) support the collaborative and distributed construction of ontologies.

1.3.3 Ontology Evaluation

Since ontologies explicitly represent domains in the form of entities, properties, and relationships that exist in the real world and constitute the domain in focus, it is a practical requirement to evaluate the developed ontologies to see whether they are fit for the purpose. **Ontology evaluation** is the area which focuses on measuring

the quality of developed ontologies. There are different approaches and preferences for evaluating the ontologies. For example, one approach is to measure formal properties such as consistency and completeness, another may look at the coverage and scope of the ontology, and another perspective might be to map some specific upper ontologies. Functionally, ontology evaluation includes aspects of ontology validation and verification which covers structural, functional and usability issues (Obrst et al., 2007).

1.3.4 Ontology Population

Once an ontology has been developed and evaluated, it is then moved into the in-use stage, as shown in Figure 1.4 (which can also be viewed as the run-time phase) with the help of bootstrapping activities such as Ontology Population and Ontology Deployment. **Ontology Population** (Amardeilh, 2006) refers to the set of activities which use automatic (Geleijnse and Korst, 2005) or semi-automatic (Celjuska and Vargas-vera, 2004) techniques to populate ontologies with instance data, whereas **Ontology Deployment** refers to the set of informal techniques often used by data publishers to make use of ontologies such as Semantic Annotation (Oren et al., 2006) and Web Forms (Tao et al., 2009a) to populate ontologies.

1.3.5 Ontology Evolution

Developed ontologies are meant to evolve. Changes in ontologies, as described by Noy and Klein (2004), can be triggered by three possible elements: (a) change in the domain; (b) change in conceptualization; or (c) change in formal specification (for example, change in RDF/RDFS/OWL specifications). Changes in ontologies are the focus of the **ontology evolution** research area. Ontology evolution is described as the activity of adapting the ontology to new knowledge that occurs as a result of domain changes, while preserving its consistency (Zablith, 2011). Ontology evolution, in general, encompasses relevant processes such as data validation, ontology changes, evolution validation and evolution management to implement the complete change management process for ontologies.

1.3.6 The missing link between the different stages of an ontology lifecycle

As mentioned earlier, ontologies are the backbone of the Semantic Web and for them to remain useable, they need to be kept up-to-date. Ontology evolution approaches proposed in the literature have focused more on syntactical and logical aspects of ontologies to ensure their validity and consistency in their conceptual model (Zablith, 2009). While these aspects are important in terms of providing tools and techniques to incorporate changes (change management) in the knowledge conceptualized by the ontology, they do not provide any assistance to the ontology developers and knowledge experts in obtaining feedback on how effective and beneficial (if at all) the existing implementations are. As shown in Figure 1.5, such feedback will be of paramount importance to the ontology lifecycle and will provide pragmatic input into the different steps in order to evolve a product (ontology) which is closely aligned with the users. In order to obtain such a feedback loop for knowledge change (ontology evolution), in in-use stage - where it experiences instantiation by different users - requires a different set of techniques to evaluate and measure how ontologies are actually being used. This intermediary place in the ontology lifecycle, where such a set of activities is employed to analyse the ontologies while in-use, is described in this thesis as **Ontology Usage** and the analysis activity is called **Ontology Usage Analysis (OUA)**.

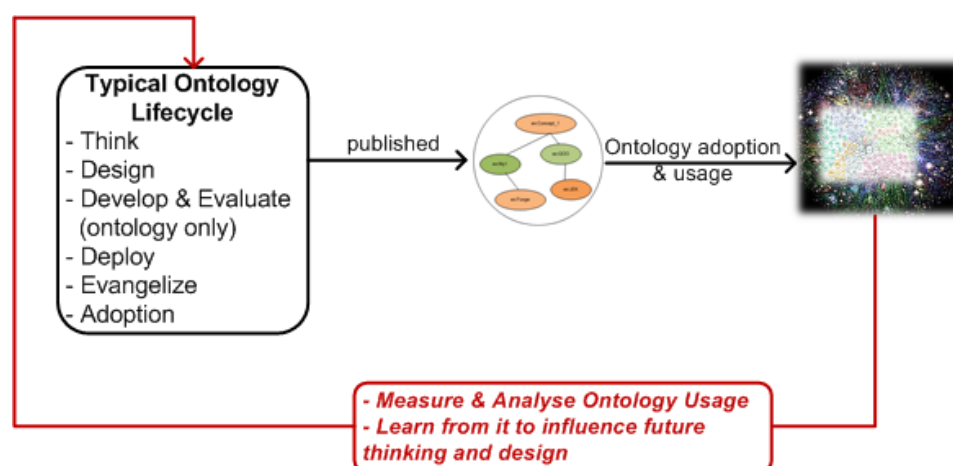


Figure 1.5: Ontology Lifecycle with a feedback loop based on Ontology Usage.

In the literature, there is extensive work around the development stage of the ontologies covering ontology development, evaluation and evolution; however, less work has been done in analysing their usage. Therefore, in order to improve

the realization and increase the effectiveness of ontologies, analyzing the usage of ontologies is an important step. Thus, an understanding and analysis of ontologies while in-use helps to obtain pragmatic insight and a feedback loop for evolving ontologies and encourages reusability. Further discussion is presented in the next section.

1.4 Need for Ontology Usage

In this section, the need to analyse the usage of ontologies is discussed.

1.4.1 Analysing the use of ontologies

As discussed in Section 1.1, the vision of the Semantic Web is to provide a standard means for publishing data on the Web that is identifiable, accessible and understandable by machines as well as humans. Semantic Web standards provide the foundational arrangements for the proliferation of RDF data which is semantically rich and enables data interoperability at a syntactic and semantic level. With the acceptance of Linked Data principles ([Heath and Bizer, 2011](#)) as the best approach for publishing and interlinking structured data on the Web, we are observing a continuous growth in the publication of Semantic Web data, often dubbed as Web-of-Data. Ontologies, being the formal and standard way for adding semantics to data, are also becoming widely adopted ([Ashraf et al., 2011](#)). According to [PingTheSemanticWeb.com](#) which maintains a list of namespaces used in RDF documents, there are around 1965 namespaces (URIs) of ontologies (vocabularies) being used on the Web¹⁶. Another source, though not considered up-to-date, is referred to in the literature is Swoogle ([Ding et al., 2004](#)) which automatically crawls the Web and has an index¹⁷ comprising approximately 10,000 ontologies. Ontologies are being adopted in different domains such as the Healthcare and Life Science domain, Gene Ontology ([Ashburner et al., 2000](#)) which is widely used to semantically describe gene-related data, Music Ontology ([Raimond et al., 2007](#)) which provides a formal framework to describe music-related information on the Semantic Web, the FOAF ([Brickley and Miller, 2004](#)) ontology which describes people, their interests and social networking aspects, and GoodRelations ([Hepp, 2008](#)) which is being adopted as a vocabulary to semantically describe business entities, offers and products.

However, while ontologies are being adopted in different domains, research has

¹⁶as of 25th June 2012

¹⁷<http://swoogle.umbc.edu/index.php>; retr. 12/9/2012

shown that the rate of adoption of ontologies is not occurring at the same pace at which it is being developed. This was highlighted by [Jain et al. \(2010\)](#) who conducted a study on the available LOD dataset and the ontologies they contained. They state that “linked data is merely more data because of the limited use of ontologies in LOD”. Other authors, too, have raised different issues which impede the use of existing ontologies and the reasons for this; for example [d’Aquin and Noy \(2012\)](#) suggest that the difficulty in finding a relevant ontology is the main factor hindering the adoption of ontologies.

The fact that greater advantage is being taken of the availability of domain-specific ontologies is encouraging, but for this to continue, we need to facilitate their adoption and reusability by empowering users with the required knowledge. This includes providing data publishers with the current ontology uptake status and the trends being observed in knowledge and data patterns. Similarly, ontology developers or domain experts need to be made aware of the variations present in the domain conceptualisation and adopt them either by specializing or generalizing the respective concepts. However, **there is currently no formal approach in the literature to evaluate, measure, and analyse the use of ontologies on the Web** in order to provide the required visibility as described above. The lack of such a methodical approach to performing empirical analysis on the use of ontologies will impact the effective and efficient utilization of Semantic Web (RDF) data made available on the Web.

This is important considering the fact that large internet companies such as Google and Yahoo, after realizing the benefits of explicit semantics, have started supporting Semantic Web standards as well as Web ontologies with reasonable adoption and maturity (for example, the GoodRelations ontology ([Hepp, 2008](#))). As a result, billions of RDF triples published on the Web (either as part of the LOD cloud or embedded within Web documents using RDFa) and thousands of ontologies will be used to annotate the data. Having an insight into how ontologies are used will assist such endeavours.

1.4.2 Encouraging the reuse of Terminological Knowledge

Ontologies are developed, published and instantiated to describe information and enable information interoperability among diverse application. It is desirable and also recommended by the Linked Data community to encourage the reuse of terms defined in existing vocabularies/ontologies (where possible) to provide coherent and consistent terminological knowledge to make it more data integrated and consumer friendly. For example, a Semantic Web application can perform a simple RDF query

to retrieve all the relevant data, where consistent terminology is used to describe the information and map other similar concepts. The reuse of terms (in RDF, this means reusing the same URIs), particularly of classes and properties, enables the ideal situation where highly reused concepts and properties become a de facto standard for the given domain (Hogan, 2011a). In a given domain, once an ontology is accepted by the community, this further encourages others data publishers to reuse the adopted ontologies, which produces network effects. As highlighted in (Hepp, 2007) ontologies exhibit positive network effects, such that their perceived utility increases with the number of people who commit to them, which comes with wider usage. The aforementioned discussion signifies the importance of ontology reuse which is linked to the adoption of ontologies by the community requiring a better understanding on **how ontologies are being used and what exactly is being used**. Presently, information regarding the use of ontologies available to the community is merely limited to the ontologies that are out there and how to access them. Therefore, the required insight and detailed ontology usage insight will be achieved by Ontology Usage. In this thesis, I investigate two problems related to the usage of ontology and believe that their resolution will (directly or indirectly) help in enabling data interoperability and subsequently data integration on the Web.

1.5 Thesis contribution

Significant contributions presented in this thesis are as follows:

- to highlight the role of ontology usage analysis in the ontology lifecycle model and propose Ontology Usage Analysis as a significant component of ontology management. Furthermore, in order elucidate its position and relationship with ontology engineering and the ontology lifecycle, I compare it with complementary areas such as Ontology Engineering and Ontology Evolution. Thus, I have tried to advance research in the ontology engineering and management research field.
 - to define a set of metrics to measure ontology usage from two perspectives, namely the ontology perspective and RDF data perspectives. An ontology perspective allows us to focus on ontology as an engineering artefact to consider its functional, structural and semantic characteristics, whereas an RDF data perspective allows triples to be evaluated to understand the data and knowledge patterns.
 - to develop techniques to measure and analyse the relationship between different ontologies based on their co-usage in describing domain specific entities.
-

- to develop a usage network model to measure the semanticity and co-usability of ontologies among different data publishers.
- to develop a conceptual knowledge architecture that facilitates the extraction of usage patterns and populate the Ontology Usage Catalogue based on the usage analysis.
- to demonstrate the application of the results obtained through the developed metrics for measuring ontology usage on the Web.

1.6 Scope of the Thesis

This thesis concentrates only on developing and evaluating *a semantic framework for analysing ontology usage on the Web*. To base the framework on empirical grounding, in this thesis, the ontologies which need to be analysed and the RDF data on which usage analysis is performed are collected by crawling the Web. This means that wherever possible, I have avoided using test data and factitious scenarios to make it closer to the real world.

However, ‘ontology usage’ is a wide concept and there is a need to clarify the scope in which it is considered in this thesis. If it is not explicitly quantified, one can implicitly assume all the usage scenarios of ontologies are being considered in this thesis. In this thesis, “on the Web” refers to the usage scenarios in which ontologies (vocabularies) are used to semantically annotate the information published on the Web. Other usage scenarios in which ontologies are used but not considered in the scope of this thesis are: Semantic Web Services (SWS) (McIlraith et al., 2001), in which ontologies (e.g. (Martin et al., 2004)) are used, but their specialized nature constrains their reusability on the Web for any other purpose; ontologies that are formalized using non-W3C ontology representation languages such as Ontolingua (Gruber, 1993).

1.7 Significance of the Thesis

As mentioned earlier, in this thesis, I propose and evaluate a framework to obtain an empirical view on how ontologies are actually being used on the Web. From the outset, this study benefits the Semantic Web community in general and specifically offers significant benefits to the ontology developer and Semantic Web application developers. Aside from being beneficial to the ontology user (which will be discussed in subsequent chapters in detail), the significance of the thesis is as follows:

-
- It proposes a solution to analyse the use of ontologies in a real world setting, therefore all the variables involved in this research such as RDF data and Web ontologies are real instances (actual data) of the usage collected by crawling the Web. Most of the Semantic Web technologies used in the implementation of the framework are W3C Semantic Web standards with the exception of a data store which is an open source application (i.e. open source version of the Virtuoso database).
 - It provides a methodology based on Semantic Web technologies to support the full process of crawling the Web (for RDF data), populating the dataset, identification of ontologies, analysing the use of ontologies, representing the usage analysis and utilizing the results.
 - It analyses ontologies from different perspectives to provide an erudite insight on the state of semantic structured data. Set of metrics are developed to measure the usage, richness and commercial advantage of the terminological knowledge of a given ontology.
 - It represents ontology usage as a bipartite graph which provides a microscopic level insight on how different data sources use domain ontologies. Such insight such as the semanticity level of different data sources helps in evaluating the conceptual model based on the actual prevalent usage
 - The obtained analysis from the developed framework provides the pragmatic feedback loop to the ontology evolution process for updating the formalized conceptual model to reflect the changes in a particular domain.
 - It applies social network analysis techniques to identify the ontology usage patterns in the RDF dataset. Based on the large scale corpus of RDF data, ontology usage network is constructed as a bipartite network to study the usage patterns hidden in the network.
 - The output of the usage analysis is represented in an ontological model to allow different applications to utilize it automatically or with little human interaction. This means that the usage patterns and the prevalent knowledge patterns are represented in the RDF data model in the form of an Ontology Usage Catalogue, which can be accessed, utilized and queried by any Semantic Web client application, hence increasing its utilization.
-

1.8 Structure of the Thesis

The remainder of the thesis is structured as follows in the subsequent chapters:

Chapter 2 : In Chapter 2, a survey of the current state of ontology engineering in general is presented, particularly focusing on Ontology Development Methodologies and Ontology Evaluation techniques. Under Ontology Development Methodologies, different proposed methodologies, methods and frameworks are described, including a discussion of the different ontology lifecycle models in the literature. Different Ontology Evaluation and Evolution approaches are discussed and the need for Ontology Usage Analysis is highlighted.

Chapter 3 : In Chapter 3, the background and the problem definition is formally presented. To provide a detailed discussion on the problem addressed through this thesis, problem definition is broken down into different research issues. Key terms and their definitions are also given to provide the background and context. The research methodology followed in this research is discussed at the end of this chapter.

Chapter 4 : In Chapter 4, the solution overview is presented for the problem defined in Chapter 3. Ontology Usage Analysis is defined and its components are discussed. The phases involved in carrying out the usage analysis in a methodological fashion are presented after the definition. After this, the framework (i.e OUSAF) which is developed to implement the activities involved in each phase is described.

Chapter 5 : This chapter deals with the identification phase of the methodological approach presented in Chapter 4. This chapter explains the detail of the method and techniques followed to identify the ontologies present in the dataset. Different techniques that are used to identify the usage patterns are discussed. Further, in this chapter, a dataset is used to understand the use of ontologies in a vertical application area.

Chapter 6 : This chapter deals with the investigation phase of the methodological approach presented in Chapter 4. A framework is developed to perform empirical analysis and measure the ontology usage on the Web. The set of metrics developed to measure the usage on empirical grounding are introduced along with their formal representation. The developed metrics are then used on a dataset built by crawling the Web to measure the usage of a domain ontology. The results are then presented.

Chapter 7 : This chapter deals with the investigation phase of the methodological approach presented in Chapter 4. A framework is developed to perform quantitative analysis on the use of ontologies on the Web. A new set of metrics to cover other important aspects of usage analysis are developed and implemented as part of the framework. An extended dataset is then used to measure the usage analysis based on the use case requirement introduced in this chapter.

Chapter 8 : This chapter deals with the representation phase of the methodological approach presented in Chapter 4. A conceptual model is developed to represent the ontology usage analysis domain. Further, in this chapter, the conceptual model is formalized (an ontology is developed) using ontology language.

Chapter 9 : This chapter deals with the utilization phase of the methodological approach presented in Chapter 4. The components developed in Chapters 5-7 are analysed by accessing them using the formalized conceptual model developed in Chapter 8. Using different use cases, the obtained results are analysed to see how these results help users to obtain the required information.

Chapter 10 : In Chapter 10, the formalized conceptual model developed in Chapter 8 is evaluated. Ontology evaluation methodology is used to analyse the different aspect of the developed ontology.

Chapter 11 : In Chapter 11, the thesis concludes with a summary of the solution developed to address the problem introduced in this thesis, followed by a discussion of future work directions.

1.9 Conclusion

The Web is transforming from a Web-of-Documents to a Web-of-Data. This transformation is enabled by Semantic Web technologies to promote data interoperability achieved through the use of ontologies. In this chapter, the role of ontologies in the realization of the Semantic Web vision was highlighted. With the continuous rise in the use of ontologies and the proliferation of Semantic Web data, the need for a solution to understand the “usage” of ontologies was highlighted. The research problem being addressed through this thesis was discussed, followed by an overview of the contribution of this thesis and the scope. The objectives of this study and the significance of this work were also discussed. Finally, the structure of this thesis was presented.

Chapter 2 - Literature Review

2.1 Introduction

As mentioned in the previous chapter, the aim of this thesis is to present a framework to measure and analyse the *usage* of *ontologies*. **Usage** and **ontology** are the two key words signifying the focus of this work. “Ontology” is an engineering artefact produced by using appropriate development methodology which comes under the definition of ontology engineering. “Usage” of an ontology is an orthogonal process to ontology development and refers to the situation or scenarios in which ontology is used for knowledge creation and knowledge representation. The knowledge creation and representation process essentially means the instantiation of ontologies, where terminological knowledge defined by the ontology is used to (semantically) describe the instance data. This instance data which contains terminological statements (schema-level information) as well as assertional statements (data-level information) is syntactically encoded in the RDF data model (also known as Semantic Web data and/or web-of-data).

In order to provide sufficient broader background and pertinent literature synopsis, in this chapter, a comprehensive survey of the literature is presented which is focused on ontology engineering (to describe “ontology” focused research work) and RDF data analysis (to discuss “usage” focused research work) that goes beyond the specific focus of our thesis. The discussion is categorised into two main categories to delineate the work based on its primary focus. First, ontology development-related work is discussed followed by work which analyses Semantic Web (RDF) data including both schema-level and instance-level data. The structure of this thesis is as follows:

- Section 2.2 presents ontology focused work which includes:
 - a discussion of the ontology engineering discipline (Section 2.2.1)
-

- an analysis of different ontology evaluation frameworks and the use of instance data in evaluation (Section 2.2.2).
- Section 2.3 presents RDF data focused work which includes:
 - a discussion of work that performs empirical analyses of RDF data on the Web (Section 2.3.1)
 - a discussion of analysis work that evaluates the presence and use of different ontologies and vocabularies (Section 2.3.2)
- Section 2.4 then concludes the chapter by giving an integrated critical view on the current state of ontology and RDF usage analysis.

2.2 Ontology Focused Work

The word *Ontology* has two different views depending on whether the person is interested in its philosophical root or its application in computer science. In this thesis, I am interested in its role in the context of computer science (ontology is typed using lowercase contrary to its use in the philosophical world where Ontology is typed using uppercase).

The use of ontology in computer science started around 1991 at DARPA as part of the Knowledge Sharing Effort ([Neches et al., 1991](#)). Obvious from the name, the aim of this project was to find ways to develop a knowledge-based system in which knowledge is represented and used as reusable components ([Corcho et al., 2007](#)). Since then, *Ontology Engineering* as a discipline has matured and provides an extensive body of knowledge to assist in the development process of ontologies. Ontologies have now become an important component of a large number of applications in different areas which includes knowledge management, customer relationship management, eCommerce, biomedical, health care, data integration and eLearning to name.

Immediately after the emergence of the Semantic Web ([Berners-Lee and Fischetti, 1999](#)), the significance and importance of ontologies came to the fore as a knowledge representation and knowledge sharing approach suitable for the Semantic Web. This applicability motivated the Semantic Web community to focus on ontologies, thus most of the work during the early days of the Semantic Web (from 1999-2006) were centered around them. This includes:

- methodologies and frameworks to develop ontologies under the name of *Ontology Engineering* ([Mizoguchi and Ikeda, 1996](#); [Jarrar and Meersman, 2002](#); [Sure et al., 2002a](#)))

- formal languages to represent ontologies under the name of *Ontology Languages* (Horrocks et al., 2003; McGuinness and van Harmelen, 2004)
- methodologies to evaluate ontologies under the name of *Ontology Evaluation* (Brewster et al., 2004; Brank et al., 2005; Gangemi et al., 2005a; Tartir and Arpinar, 2007)
- methodologies to evolve ontologies under the name of *Ontology Evolution* (Noy and Klein, 2004; Vrandeovic, 2010)
- formal logic for reasoning with ontologies under the name of *RDFS/OWL Reasoning* (Sirin and Parsia, 2004; Meditskos and Bassiliades, 2010; De Bruijn et al., 2005).

Each abovementioned area has a focused research effort around it however, the research community group them under the rubric of Ontology Engineering. Ontology engineering normally covers three sets of activities; (a) ontology development methodologies and processes; (b) ontology lifecycle models; and (c) tools and languages for supporting and automating ontological development as shown in Figure 2.1.

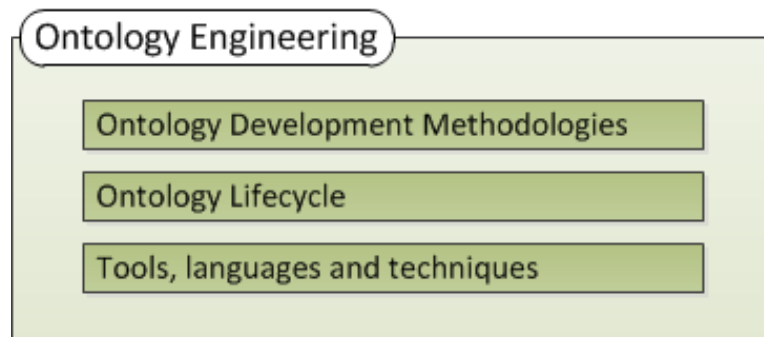


Figure 2.1: Ontology Engineering components.

In this section, the first two sets of activities, (a) and (b), are the focus because of their relevance and overlap with our work.

2.2.1 Ontology Development Methodologies and Processes

Similar to software engineering and software development lifecycle models (Boehm, 1987), ontologies are developed and maintained using ontology development methodologies which are important component of Ontology Engineering (Gómez-Pérez et al., 2004). Most of the present methodologies such as On-To-Knowledge Management (Sure et al., 2004) and METHONTOLOGY (Fernández-López et al., 1997) tend to cover the engineering aspects of the lifecycle which includes *requirement*

analysis, ontology development, evaluation and maintenance (Tran et al., 2008). However, the common limitation in most of these methodologies (further discussion follows in the next subsection) is the shallow consideration of the usage aspects of the developed ontologies which is often placed under the post development (maintenance) stage.

Before the different methodologies, and the methods and processes proposed, developed and deployed for the development of ontologies are discussed, the definition of methodologies and methods standardised by the IEEE Standard Glossary of Software Engineering Terminology (Radatz et al., 1990) is firstly reviewed.

Methodology : A comprehensive, integrated series of techniques or methods creating a general systems theory of how a class of thought-intensive work ought be performed.

Method : A method is an orderly process or procedure used in the engineering of a product or performing a service.

Technique : A technique is a technical and managerial procedure used to achieve a given objective.

Process : A sequence of interdependent and linked procedures which, at every stage, consume one or more resources to convert inputs into outputs.

There are a growing number of methodologies proposed and used to address the issue of ontology development. Few methodologies attempt to cover the whole lifecycle of ontology engineering, ranging from requirement elicitation, development, evaluation and maintenance, while others focus on a specific stage or process of the ontology engineering.

In the literature, over two dozen methodologies and methods supporting ontology development were found presented in the following in chronological order. A few of the methodologies which are relevant to this work will be discussed in reasonable detail. The methodologies and methods have been categorised based on the following classifications:

- methodologies which develop ontologies from scratch
 - methodologies which support cooperative and distributed construction of ontologies
 - methodologies which use Web 2.0 features to provide social networking aspects in ontology development
-

Each of these groups is discussed in subsequent subsections.

2.2.1.1 Methodologies and Methods for Building Ontologies from Scratch

Different methodologies which support the creation of ontologies from the scratch are briefly described as follows:

- Cyc (Elkan and Greiner, 1993; Lenat et al., 1990): Cyc methodology was the result of the experience gained through the development of the Cyc knowledge base comprised of common sense knowledge. A detailed description of Cyc methodology and knowledge bases is available at www.cyc.com (retr.; 21/04/2012).
- Uschold and King's methodology (Uschold and King, 1995; Uschold and Gruninger, 1996; Uschold, 1996): This methodology is the result of research done on the development of Enterprise Ontology to model the enterprise processes. This ontology represents the terms and definitions relevant to business enterprises. The detail on Enterprise ontology and the methodology can be accessed from <http://www.aiai.ed.ac.uk/project/enterprise/> (retr.; 21/04/2012)
- Toronto Virtual Enterprise Methodology (Gruninger and Fox, 1994b,a; Uschold and Gruninger, 1996): In the literature, this is also known as Gruninger and Fox's methodology. Part of the TOVE project, the methodology comprises several steps: (a) motivation scenarios; (b) informal competency questions; (c) first-order logic-based terminology; (d) formalization of competency questions; and (e) definitions of semantics and constraints. One of the significant elements of this work is that it is considered the first reported use of competency questions in defining the scope of ontology.

Further detail on TOVE can be found at <http://www.eil.utoronto.ca/enterprise-modelling/entmethod/index.html> (retr.; 21/04/2012)

- KACTUS (Schreiber et al., 1995; Schreiber and Terpstra, 1996; Wielinga et al., 1995): This is a European ESPRIL -III project aimed at evaluating the feasibility of knowledge reuse in complex technical systems and the role of ontologies in giving explicit structure to the knowledge. Using the methodology, the authors developed three ontologies and applications: fault diagnosis in electrical networks, scheduling service resumption after a fault appears and control of electrical networks. Further detail on KACTUS can be found at <http://hcs.science.uva.nl/projects/NewKACTUS/home.html> (retr.; 21/04/2012)
-

- METHONTOLOGY (Asunción Gómez-Pérez et al., 1996; Fernández-López et al., 1997; López et al., 1999; Vega et al., 2001) : This is one of the most famous ontology development methodologies which defines a comprehensive set of activities needed for the development and maintenance of ontologies. In addition to the activities, the authors also describe the lifecycle of an ontology starting from requirement gathering to the evolution of the ontology. The stages through which the ontology passes are: *specification, conceptualization, formalization, integration, implementation*. In addition to these core development centric activities, a few umbrella activities such as *evaluation* and *documents* are used which run through the lifecycle stages. One of the significant achievements of METHONTOLOGY is its consideration for the development of ontologies by the Foundation for Intelligent Physical Agents (FIPA)¹, which promotes communication across agent-based applications. This methodology will be discussed in more detail in subsequent sections. Further detail on METHONTOLOGY can be found at: <http://www.oeg-upm.net/> (retr.; 25/04/2012).
- SENSUS (Swartout et al., 1997; Knight and Luk, 1994; Knight et al., 1995; Valente et al., 1999): This is one of the early approaches toward knowledge sharing using ontologies. This approach is based on the assumption that if two knowledge bases are using the same base ontology, then knowledge can be easily shared between these knowledge bases since they share a common structure. The SENSUS methodology comprises the following steps: (a) a list of terms are identified as seed terms that are particular to the domain; (b) seed terms are then linked with the SENSUS ontology (the SENSUS method makes use of the SENSUS ontology which has more than 70,000 concepts organized in hierarchy according to their abstraction level (Fernandez-Lopez et al., 2002)); (c) then, all the concepts in the path from the seed terms to the root of SENSUS are included; (e) relevant terms which are missing are then added manually; and (f) at the end, for those nodes with a high betweenness value, the entire sub-tree under this node is added.

Using the abovementioned approach, an ontology for military air campaign planning was built which describes basic elements such as the air campaign plan, scenarios, commanders, participants (Valente et al., 1999). Further detail on methodologies and ontologies can be found at <http://www.isi.edu/natural-language/projects/ONTOLOGIES.html> (retr.; 24/04/2012)

¹<http://www.fipa.org>

- On-To-Knowledge (OTK) (Sure, 2002; Sure et al., 2002a, 2003, 2004): Part of the EU IST-1999-10132 project, the On-To-Knowledge (OTK) methodology was developed for the introduction and maintenance of knowledge based applications in enterprises focused on knowledge processes and knowledge meta processes, based on ontologies. This methodology comprises the following stages: (a) *kick-off*: requirements are identified, competency questions are identified and the final draft of the ontology is developed either from scratch or reusing possible existing ontologies, (b) *refinement*: the ontology is refined to meet the application requirements; (c) *evaluation*: the ontology is evaluated using competency questions to measure its usefulness; and (d) *ontology maintenance* the ontology is updated to reflect changes.

This project was later joined by the Ontotext company² in 2001 to develop ontology middleware and a reasoning module based on the work that went into the On-To-Methodology. Ontology middleware developed through this collaboration (Broekstra et al., 2002) provided the administrative layer on top of On-To-Knowledge to make this research work more integrateable with real-world application.

On-To-Knowledge methodology details are available at <http://www.ontotext.com/research/otk> and www.ontoknowledge.org/ (retr.; 18/09/2012)

- DOLCE (Claudio et al., 2005; Oberle et al., 2005; Stuckenschmidt, 2003) : DOLCE stands for a Descriptive Ontology for Linguistic and Cognitive Engineering. The main idea behind this project was to develop first-order logic based ontologies for inclusion in the WonderWeb foundation Ontologies Library (Horrocks, 2005). DOLCE, an upper level ontology, was the first module of this library to be built by firstly introducing the concepts informally along with the basic categories, functions and relations. Later, detailed axiomatization was added to impose the constraints on the model and clarify the assumptions through the illustration of formal consequences (Masolo et al., 2003). As part of this project, the KAON open-source ontology management infrastructure (Volz et al., 2002) was also developed to provide tools to manage ontologies. It includes a comprehensive tool suite allowing easy ontology creation and management, as well as building ontology-based applications (Horrocks, 2005).

Further detail on the WonderWeb project and its deliverables can be found at <http://wonderweb.semanticweb.org/> (retr.; 28/08/2012) and for DOLCE visit <http://www.loa.istc.cnr.it/DOLCE.html> (last accessed; 28 April 2012)

²<http://www.ontotext.com/research/otk>

- KBSI IDEF5 (Benjamin et al., 1994; Grover and Kettinger, 2000) : IDEF5, which stands for the Integrated Definition for the Ontology Description Capture Method is an ontology engineering approach toward the development, modification and maintenance of domain ontologies. The IDEF5 method is part of the IDEF family of modeling languages in the field of ontology engineering. This method considers ontology development as open-ended work which cannot be effectively adopted using a "cookbook" approach, therefore they published a general procedure with a set of guidelines comprising the following activities:
 - *organizing and scoping*: the purpose, viewpoint and context for the ontology development project is identified and roles are assigned to team members.
 - *data collection*: the raw data required for the development is gathered using typical knowledge acquisition techniques such as protocol analysis and expert interviews.
 - *initial ontology development*: the data obtained from the previous activity is used to build a prototypical ontology which contains proto-concepts (concepts, relations and properties).
 - *ontology refinement and validation*: the proto-concepts are iteratively refined and tested. This is essentially a deductive validation procedure to refine and validate the ontology to complete the development process.

According to the IDEF5 methodology, the initial ontology is defined with a schematic language which is a set of graphical notations used to express the most common form of ontological information.

Further details on the IDEF5 methodology and the IDEF5 schematic language can be found at <http://www.idef.com/IDEF5.htm> (retr.; 29/05/2012)

In addition to the abovementioned methodologies, many approaches have been proposed to address a specific aspect of ontology development. In the following, brief details on these methods is provided.

- MENELAS (Zweigenbaum, 1994; Medicale, 1995; Zweigenbaum et al., 2001): MENELAS is based on four principles: similarity, specificity, opposition and unique semantic axis, which helps in the development of taxonomic knowledge in ontologies. Based on these four principles, the MENELAS ontology was designed as part of a natural language understanding system. MENELAS was then used to develop an access system for medical records using natural language. Further details on MENELAS can be found at <http://estime.spim.jussieu.fr/Menelas/> (retr.; 25/07/2012)

- ONION (Gangemi et al., 1999a, 1996; Steve et al., 1997): The ONIONS (ONtological Integration Of Naive Sources) project was initiated in 1990 to address the problem of conceptual heterogeneity, particularly in the medical domain. The object of the project was to develop a large-scale ontology library for medical terminology. The terminological knowledge in this approach is acquired by conceptual analysis and ontology integration. The ONIONS methodology exploits a set of formalisms, a set of computational tools that implement and support the use of the formalisms, and a set of generic ontologies taken from the literature in either formal or informal status and translated or adapted to the formalism proposed by (Gangemi et al., 1999b).

For more detail visit <http://ksi.cpsc.ucalgary.ca/KAW/KAW96/steve/introduction.html> (retr.; 29/05/2012)

- COIN (Madnick et al., 2003; Madnick and Lupu, 2008) : The COntext INterchange (COIN) strategy, developed at MIT's Sloan School of Management, is an approach to solve the problem of inter-operability of semantically heterogeneous data sources through context mediation. It provides the notations and syntax to represent AN ontology. This approach attempts at resolving semantic conflicts among heterogenous systemS by defining the context axioms corresponding to the systems involved in the interaction. It also provides formal characterization and reasoning underlying the context interchange strategy (Goh et al., 1999).

In the next subsection, ontology development methodologies that support cooperative and distributed approaches are described.

2.2.1.2 Cooperative and Distributed Approaches for Ontology Building

An ontology is a shared and common understanding of some domain which is built by establishing an agreement among domain experts on the conceptual model of the domain. Since ontologies have to be available on the Web and the end users of the ontologies may be from different locations, in order to arrive at consensus on the ontological model, it is important to have methodologies which support the development of an ontology distributedly, as shown in Figure 2.2.

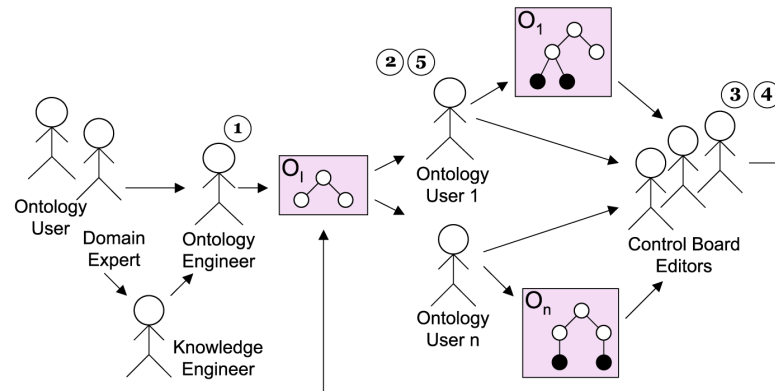


Figure 2.2: Roles and functions in distributed ontology engineering (cf. (Vrandečić et al., 2005)).

Selective methodologies which support the distributed and collaborative ontology development process are briefly described below.

- CO4 (Euzenat, 1996, 1995, 1997): CO4 is one of the earliest work started at INRIA³ toward developing ontologies cooperatively. It enables different people to discuss, share and establish agreement on the domain model to represent consensual knowledge in the knowledge base. Consensus on the content of the knowledge base is achieved by a protocol which integrates knowledge, based on the reached consensus. The knowledge base architecture is shown in Figure 2.3

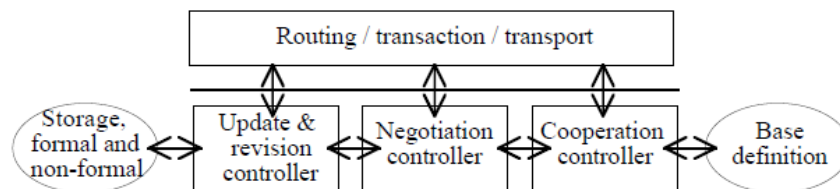


Figure 2.3: The software architecture. Each box represents a software module, each circled unit is a data/knowledge repository and each arrow represents the call of a program functionality (Euzenat, 1996)

- NeOn Methodology (Presutti et al., 2008; Blomqvist et al., 2009; Suárez-Figueroa et al., 2012): NeOn is the latest methodology which supports the collaborative aspects of ontology development, reuse and evolution in distributed environments. It is considered a scenario-based methodology for building ontology networks which makes it a flexible approach, providing variety of pathways for ontology development (Suárez-Figueroa et al., 2012). The key components of the NeOn methodology include: (a) a set of commonly occurring

³<http://www.inria.fr/>

nine scenarios for building ontologies, such as when to re-engineer the available ontology, and align, modularize and localize this with ontology design patterns; (b) NeOn glossary of processes and activities; and (c) methodology guidelines for each process which includes: (i) a filling card, (ii) a workflow, and (iii) examples. The nine possible scenarios and expected output and existing knowledge resources to be reused is shown in Figure 2.4.

Further details on the NeOn methodology are available at : <http://mayor2.dia.fi.upm.es/oeg/index.php/en/methodologies/59-neon-methodology> (retr.; 20/06/2012)

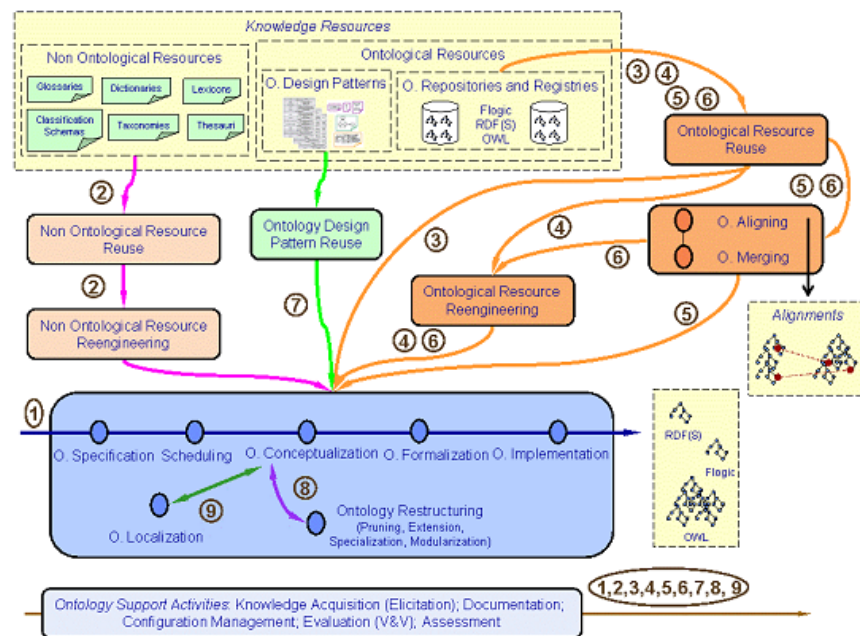


Figure 2.4: Scenarios for Building Ontology Networks (Gomez-Perez and Suarez-Figueroa, 2009)

- HCOME (Kotis et al., 2004; Kotis and Vouros, 2006; Kotis, 2008) : The HCOME methodology is a human-centered approach which integrates argumentation and ontology engineering in a distributed setting, where ontologies are considered living artifacts, ontology development is considered a dynamic process and special focus is given to ontology evolution throughout the ontology lifecycle. In HCOME, expert users formalize their own ontology first before sharing with others (through the Shared Space) to evolve the conceptual model. After deliberation by the domain-specific community, agreement is achieved before moving it into the Agreed Space. While the HCOME methodology enables consensus-building on ontologies right from the start of development, the actual benefits of such an approach are unknown since the authors did not report any

experience or adoption of the methodology.

- MeltingPoint ([Garcia et al., 2010](#)) : The Melting Point (MP) methodology provides a collaborative ontology development environment in decentralized settings. MP methodologies are the result of the experience the authors obtained through their work in the biology domain. This methodology reuses some of the components of several ontologies and analyses their reusability in the MP methodology.

In the next sub-section, the third category of methodologies in which Web2.0 approaches are used is briefly described.

2.2.1.3 Ontology Development Approaches using Web2.0 features

A few of the methodologies which support collaboration-based ontology development in decentralized settings have been described. The emergence of implicit semantics based on social interaction on the Web (i.e. social web sites) has motivated ontology researchers to use Web2.0 technologies in developing ontologies. Several techniques, such as social tagging systems (STS) ([Heymann and Garcia-Molina, 2006](#)) which allow users to freely associate terms to the resources are being used to allow users provide implicitly conceptual structures and semantics. Such conceptual structures are known as folksonomies and are increasingly being used for information retrieval, discovery and clustering on the Web. In the following sub-section, few of the methodologies which have considered social interaction and Web 2.0 approaches in the ontology engineering process are presented.

- FolksOntology ([Van Damme et al., 2007](#)) : In this research, the authors presented a comprehensive approach for driving ontologies from folksonomies by integrating multiple techniques and resources. Folksonomies are analysed using statistical analysis techniques to measure the usage, structure and the implicit social networks to compare them with different knowledge resources such as Wikipedia, WordNet and online dictionaries. After data analysis, ontology mapping and matching techniques are used to create correspondence between terms to develop consensus over ontology elements.
 - Ontology Maturing ([Braun et al., 2007](#)) In the Ontology Maturing approach, the authors consider ontology engineering more as a collaborative informal learning process and less of a specialized knowledge engineering approach. Therefore, they proposed the Ontology Maturing process in which users engage
-

in ontology engineering in their everyday work processes by integrating tagging and folksonomies with formal ontologies. This makes ontology development a learning process which is a continuous evolution process. The development process is structured into four phases known as the knowledge maturing process: (a) emergence of ideas; (b) consolidation in communities; (c) formalization; and (d) axiomatization.

2.2.1.4 Summarizing Ontology Development Methodologies and Processes.

Most of the methodologies, as part of the development process, include different management and maintenance-related processes to provide a complete ontology development framework. There are several other classifications used by researchers to draw a comparison between different methodologies to establish a better understanding of their similarities and peculiarities. For example, [Fernández-López et al. \(2002\)](#) analyzed different methodologies, grouped on the basis of whether the methodology supports the development of new ontologies from scratch, reusing other ontologies without transforming them and re-engineering ontologies. In other similar work, [Jones et al. \(1998\)](#) discusses different methodologies which provide the complete ontology lifecycle support and the methods which address a specific aspect of ontology development. [Corcho et al. \(2003\)](#) presented a comparative analysis of different methodologies based on the set of processes used in the development phase. More recently, the work of [Suárez-Figueroa et al. \(2012\)](#) consolidated the research done pertaining to ontology development as part of the NeOn project. In their work, they proposed the nine most commonly occurring scenarios and the solution offered by the NeOn methodology framework. Restricted in scope, [Changrui and Yan \(2012\)](#) published comparative research on methodologies for domain ontology development.

However, it was observed that there is no consistent ontology lifecycle model implemented to understand the different stages through which an ontology passes. The overall focus of the methodologies are centered around development-related processes with a few exceptions, such as On-To-Knowledge and METHONTOLOGY which provide project management and integration-related processes as part of the methodology.

2.2.2 Ontology Lifecycle

As mentioned in Section 2.2 and depicted in Figure 2.1, the second component of ontology engineering is the ontology lifecycle. It is important to understand the ontology lifecycle from a high level perspective to group the related set of activities

to generalize the lifecycle stages. Understanding the ontology lifecycle helps in identifying the stages through which an ontology passes from its inception to its utilization, either in knowledge-driven applications or on the Web for information annotation.

According to the Oxford English Dictionary, the generic definition of lifecycle is *"a course or evolution from a beginning, through development and productivity, to decay or ending"*. In the context of ontologies, the ontology lifecycle is considered different from the ontology development lifecycle model. This difference has emerged from the very fact that ontology (specifically in the Semantic Web) contains the formalized representation of domain knowledge (statements expressed using OWL) but at the same time, it can contain the RDF statements using the terms defined by the ontology. In other words, any document or set of statements can contain both the statements describing the terminological knowledge (T-Box) and/or the statements describing instance data (A-Box). Therefore, in this thesis, the document which describes the conceptualized domain model is considered to be the formalized representation of the ontology. Thus, ontology lifecycle refers to the evolution through which the ontological model passes during its different stages, especially during development and usage. This thesis is particularly interested in the usage of ontologies to identify how ontologies have been received and treated after the development phase. In Figure 2.5 (ontology-lifecycle -high level) two phases of ontologies are shown, first as the "design time" in which the ontology is being developed and second, the "run time" in which the ontology is used in either a knowledge-drive application or for annotation.

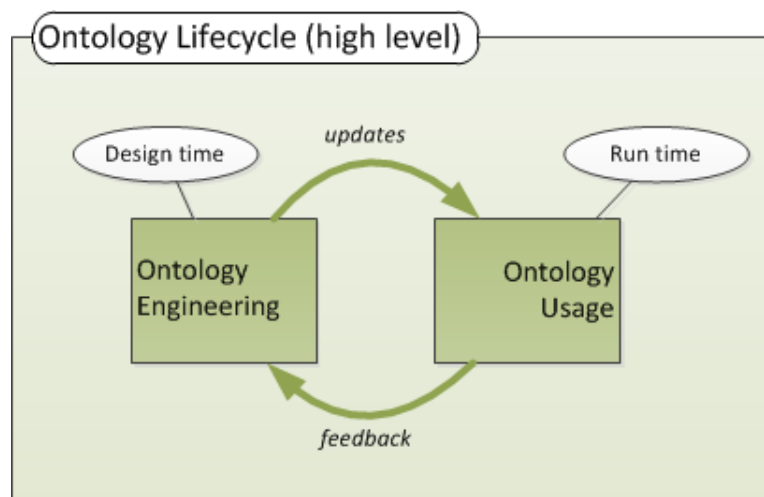


Figure 2.5: A high level view of Ontology lifecycle

In the literature, the ontology lifecycle is mostly discussed as part of ontology development methodology, such as Cyc (Elkan and Greiner, 1993), Ushold & King (Ushold and King, 1995), METHONTOLOGY (Asunción Gómez-Pérez et al.,

1996) and On-To-Knowledge (Sure, 2002). The lifecycle model discussed in these methodologies primarily relates to the lifecycle models found in software engineering disciplines such as waterfall (Schwaber et al., 1995), spiral (Boehm, 1988) and prototypical (Aoyama et al., 1998). The most recent survey on the lifecycle models of data and knowledge-centric systems is presented by Möller (2012). In this work, the author first describes different lifecycle models used in data-centric domains, such as digital libraries, multimedia, eLearning, knowledge and Web content management and ontology development. Based on the comparative analysis of the existing models, the author then proposes a meta-vocabulary of lifecycle models for data-centric systems. Using the meta-vocabulary, the Abstract Data Lifecycle Model (ADLM) is developed, along with additional actor features and generic features of data and metadata, as shown in Figure 2.6.

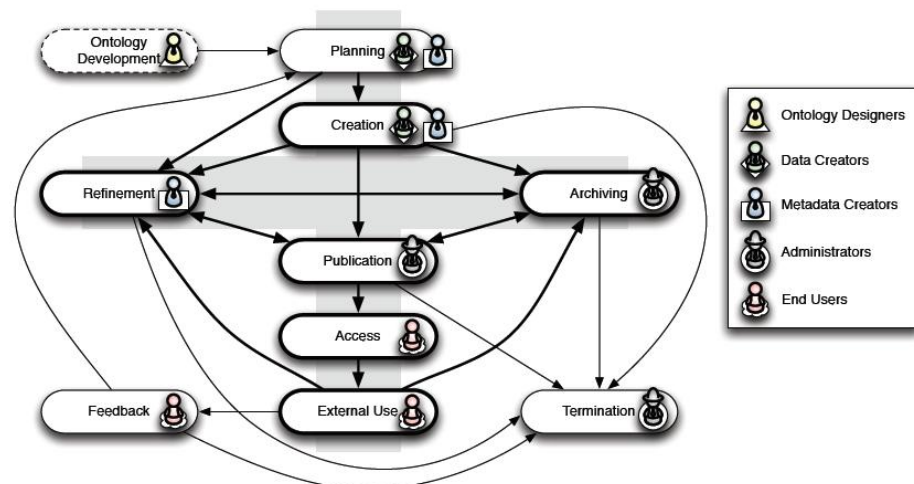


Figure 2.6: Abstract Data Lifecycle Model (Möller, 2012)

ADLM, being the meta-model used to describe the related but different lifecycle models, provides a comprehensive coverage to the usage aspect of ontologies (run-time stage of the ontology lifecycle; see Figure 2.5). A number of the processes of the ADLM model are focused on the run-time activities through which ontologies pass while in use.

In other work, Tran et al. (2008) discusses the role of the ontology lifecycle in ontology-based information systems (OIS). In relation to the management of ontology lifecycle, they proposed a simplified lifecycle model shown in Figure 2.7. In this lifecycle model, contrary to those published under the label of methodologies (of ontology development), an equal emphasis is given to ontology usage and ontology engineering. In ontology engineering activities, after reviewing the different methodologies, they proposed that the main ontology engineering lifecycle activities were requirement analysis, development, evaluation, and maintenance. Ontology

usage encompasses all the activities which are performed on the ontology after it is developed and in use i.e. in run-time state. In the lifecycle model proposed by Tran et al. (2008) in relation to ontology usage, they cover all the services and processes which are involved in accessing and manipulating an ontology, such as search, retrieval and cleansing. A reasoning service is also included to infer implicit knowledge which helps in expanding and refining the query and expanding the search results, including statements deductable through inferencing. One of the important services/activities in ontology usage is *ontology population* which populates the knowledge base with the instance data marked (annotated) with the ontology. In their work, they considered it to be a manual intensive work carried out by collecting instance data from the user via online forms. Obviously, this will impose substantial overhead and in a practical setting, could become burdensome.

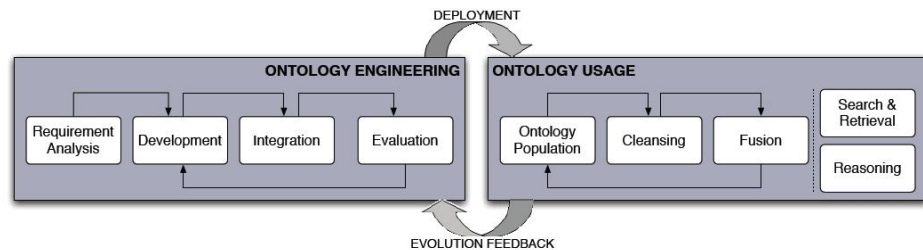


Figure 2.7: Lifecycle model (c.f. (Tran et al., 2008))

Therefore, an efficient mechanism needs to be proposed to make it more automated and requiring less manual work. Additionally, their lifecycle model is based on a layered architecture for ontology-based applications. The three proposed layers are: the *presentation layer* (this layer contains the presentation-related component capable of generating interfaces for diverse target devices such as the browser, desktop and mobile), the *logic layer* (in this layer, application-specific services are implemented for a particular use case); and the *data layer* (this layer contains different kinds of data sources such as databases and file systems).

2.2.2.1 Summarizing the Ontology Lifecycle.

In this sub-section, a discussion of work related to the ontology lifecycle model is discussed. First, a simplified and generalized lifecycle model is described to discern the different components and aspects involved in the ontology lifecycle model. As shown in Figure 2.5, the simplified lifecycle model comprises design time and run-time stages and communication between these two stages to make the model adaptive. In the literature (and also pointed out in (Möller, 2012)), it was found that the ontology lifecycle is implicitly *described and labeled under ontology development*

methodologies work. Therefore, *the lifecycle models discussed in the literature are more representative of development-related activities rather than equally covering ontology usage aspects*. Like any other engineering artifact, ontologies are developed to be used by its end-users and its value increases as it becomes more highly used. While producing well engineered ontologies is important, consideration also needs to be given to ontology usage as an integral part of ontology lifecycle models to make them a more living artifact.

2.2.3 Ontology Evaluation Frameworks

Ontologies, when developed, are evaluated to measure their quality and fitness based on the requirements specification. It is very important to evaluate the ontology while in the development phase to ensure that when the ontology is used, it is *fit* to serve the purpose. There are already several frameworks available for ontology evaluation. Before proceeding with the discussion of different ontology evaluation approaches, how ontology evaluation relates to the ontology lifecycle is discussed, as shown in Figure 2.8

Ontology Evaluation (OE), often described as a sub-area of ontology engineering, covers research pertaining to the measurement of the quality, usefulness and fitness of the developed ontology, *with or without considering the instance data*. Since ontologies are an important component of the Semantic Web, ontologies could have diverse usage scenarios not known to the ontology developer. Therefore, it is important to also evaluate how a particular ontology is being received and used in the real world. There are two points in the ontology lifecycle where the ontology needs to be evaluated to provide a comprehensive feedback loop to all ontology stakeholders. First, *during and after ontology development* and second, *while the ontology is in use*. Most of the literature covers the first evaluation point where the ontology is assessed prior to its actual utilization. In the following, the existing ontology evaluation frameworks are surveyed and the stages within the lifecycle where they are analysed are identified.

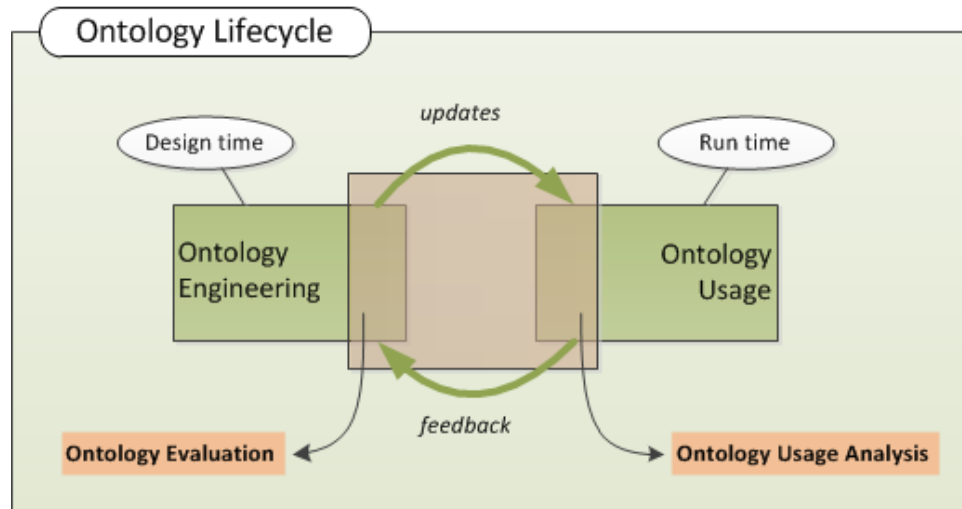


Figure 2.8: Ontology Lifecycle model with Ontology Evaluation.

From the beginning, ontology evaluation frameworks were proposed as part of ontology engineering to assess ontologies from different dimensions. One of the classifications of ontology evaluation approaches is reported in (Brank et al., 2005) in which the authors categorized them, based on their objectives as follows:

- Golden standard: where an ontology is used to evaluate other ontologies
- Application-based: using the ontology in an application and evaluating the results
- Data driven: analysing the content of an ontology to measure its domain coverage
- Assessment by humans: conducted by humans to assess whether the ontology meets the required criteria

Other research has studied the structural aspects of ontologies to understand the relationship between ontology usefulness (fit for the purpose) and its topological properties. For example, Tartir et al. (2005) presented a framework and a set of measurements to evaluate the richness, connectivity, fullness and cohesiveness of a given ontology. They proposed metrics and measures in their study to determine whether an ontology is domain-specific or generic. While the metrics are interesting, however, their actual usefulness is not well known. The proposed metrics are evaluated on a very small data set which by no means reflects the actual instantiation.

In (Guarino and Welty, 2004), the authors proposed the OntoClean methodology in their approach to **evaluate** and **validate** the ontology's taxonomical relationship by employing formal notions from philosophy such as essence, identity and unity. Four

meta-properties (rigidity, identity, unity and dependence) and operators (+, -, ~) to symbolically specify the characteristics of ontology components, such as classes and relationships, were used to validate the assumption which influenced the ontological model. Similar to the abovementioned approaches, this study has made use of examples often used in presentation and teaching materials which, in our thinking, cannot sufficiently represent the actual ontologies being used on the Web. In other research, [Alani et al. \(2006\)](#) proposed a method for the evaluation and ranking of ontologies based on four ranking measures: class match measure (evaluates the coverage of an ontology for a given search term); density measure (measures the density based on its neighbourhood i.e. sub-classes, super-classes, relationships and sibling); semantic similarity measure (how close the concepts of interest are laid out in the ontology); and betweenness measure (how far a concept is from the root concept of its hierarchy). Evaluation approaches ([Guarino and Welty, 2004](#)) which either relied largely on the structural aspect of the concepts or the integrated evaluation of ontologies approach ([Alani et al., 2006](#)) provide insight on how knowledge is distributed and the existence of different thematic hierarchies in a given ontology, but it is believed that without incorporating the actual usage data of the ontologies in such studies, this will provide only shallow observations without any empirical findings.

Recent work in this area was reported by Vrandečić in his dissertation ([Dasgupta et al., 2007](#)) which answers the research question, *how to assess the quality of an ontology on the Web?* He proposes concrete measures relating to quality, computational efficiency, accuracy, usefulness and adaptability of the ontology which he defines as a formal artifact comprising classes, properties and instance data. He argues that the right approach for ontology evaluation is to find the methods and metrics which tell us whether the ontology is flawed and if so, in which way, instead of merely evaluating an ontology to measure its goodness.

Another different but related ontology evaluation approach is **ontology evolution** which implements change management in ontology engineering. [Stojanovic \(2004\)](#) presented the theory and practice of ontology evolution. The evolution of ontologies has been addressed by different researchers by defining change operations and change representations for ontology languages. [Stojanovic \(2004\)](#) presented evolution strategies to handle inconsistencies for evolving ontologies in a centralized setting and for the handling of ontology changes in a distributed setting. However, change operations have been proposed for specific ontology languages, such as OKBC ([Chaudhri et al., 1998](#)), OWL ([Patel-Schneider et al., 2004](#)) and the KAON ontology language ([Bozsak et al., 2002](#)). Based on work by ([Stojanovic, 2004](#)), [Haase and Stojanovic \(2005\)](#) extended the work for OWL-DL ontologies which focused on inconsistencies while investigating the ontology evolution. They also developed a tool

called evOWLution which implements their approach.

In the aforementioned sub-section, a few of the ontology evaluation frameworks which have been proposed in the literature were described. Different authors proposed different methods to focus on specific aspects of ontology during evaluation. While a few authors, such as [Fox and Gruninger \(1998\)](#); [Guarino and Welty \(2004\)](#); [Dasgupta et al. \(2007\)](#) focused on functional completeness, generality, efficiency, perspicuity, precision granularity, and minimality of ontologies, others authors such as [Hoser et al. \(2006\)](#); [Zhang \(2008\)](#) considered the **structural aspects** of an ontology to measure their topological characteristic to infer their effectiveness and usefulness. A significant amount of literature focusing on evaluating ontologies before they are even used highlights the absence of work on evaluating ontologies while they are being used. The assessment of ontologies while in use could provide pragmatic and empirical assessment on how vocabularies are being used and adopted which, in return, could help in improving the ontological model and effective knowledge utilization.

In the following section, work in which ontologies and their usage is analyzed to measure the usage, data patterns and knowledge patterns in the RDF dataset is presented.

2.3 Semantic Web (RDF) Data Focused

In this section, work which has analysed RDF data published on the Web, also known as the *web-of-data* is described. As mentioned in Section 2.1, after the introduction of Linked Data principles ([Heath and Bizer, 2011](#)) by Sir Tim Berners-Lee in 2006, there has been tremendous growth in the publication of structured data on the Web. The growth in the RDF is credited to the simplicity of the four linked data principles (see Section 1.2) which have provided an easy-to-follow approach for publishing Semantic Web data on the Web. Another effort which has significantly contributed to the proliferation of RDF data on the Web is the research community effort dubbed the Linked Open Data (LOD) project ([Kobilarov et al., 2009a](#)) which has contributed billions of schema-level and instance level triples on the Web, covering myriad application areas.

It is well known that Semantic Web data is loosely comprised of two types of statements: one in which terminological knowledge is described; and in the second, where instance data is defined. Though both types of statements are encoded and stored in the same documents, their classification helps in conducting more focused analysis depending on the objectives. In order to provide sufficient coverage on the work in which RDF data is analysed to find the data and knowledge patterns and

share best practices, the relevant work has been categorised, based on its focus. In the first sub-section, I present the work in which *empirical analysis is performed on RDF data*; and in the second sub-section, the work that *evaluates the presence and use of different ontologies and vocabularies* is described.

2.3.1 Empirical Analysis of RDF Data on the Web

In this section, the work in which RDF data is analyzed to understand the data and knowledge patterns available is described. Several research efforts have made use of the Linked Open Data (LOD) cloud datasets to perform empirical analysis which is published as part of the ISWC (International Semantic Web Conference)'s Semantic Web Challenge⁴.

The simplicity of Linked Data principles (Bizer et al., 2009), introduced by Tim Berners-Lee in 2006, and the consequential monumental success of the Linked Open Data project (Heath and Bizer, 2011) transformed the Web into a structured data space. This new data space, comprising self-describable data based on a standard model (RDF), provided a test bed for researchers⁵ to unleash and exploit the potential of Semantic Data on the Web. Researchers have analysed the Web of Data to understand the nebulous nature of the **quality of data**. One of the early attempts to analyse the quality of RDF data was made by Hogan et al. (2010), who reported the common errors made by the early RDF data publishers. While highlighting the shortcomings, issues and findings, the authors provided guidelines for both data publishers and consumers to assist in generating and consuming high quality semantic data. An analysis experiment was done on a dataset comprising data crawled from 150,000 URIs. While the prime focus was to measure noise and inconsistency in the dataset, they classified the errors into four categories: (a) *accessibility*; (b) *syntactical errors*; (c) *reasoning*; and (d) *non-authoritative contributions*. One of the significant findings was that 14.3% of the properties (URIs used as a predicate in triple) and 8.1% of the classes were used in ways for which their declaration and description is not available. In addition to this, they also reported the use of certain vocabulary terms against their original semantics and purpose, known as ontology hijacking. Their work reports on the quality of the RDF data which is very subjective, therefore it cannot be generalized since each application has its own data requirements and specific modelling choices. Hence, it is believed that while analysing

⁴<http://challenge.semanticweb.org/>

⁵For example, Linked Open Data Around-The-Clock (LATC) is an European-funded project, to "create an in-depth **test-bed** for data intensive applications by publishing datasets produced by the European Commission, the European Parliament, and other European institutions as Linked Data on the Web" as one of their objectives (website : <http://latc-project.eu/about>, Last accessed 21 March, 2012)

“data”, one also needs to look at the traces on “knowledge” on the dataset to allow the maximum utilization of analysis findings.

In other work, [Auer and Lehmann \(2010\)](#) tried to **identify the shortcomings** of the LOD Cloud and suggest ways in which it could be improved and used for practical purposes. Specifically, they highlighted: (a) the need to improve the performance of RDF data management tools to provide efficient data processing; (b) the need to improve the state of “interlinking” among diverse datasets to provide a typed linked dataspace; (c) the need to improve the algorithms and tools to enhance the quality of the linked data; and (d) the need to provide an adaptive user interaction experience to support linked data management services. The authors identified potential areas for the focus of future research work, along with its expected outcomes; however, they did not highlight the role of ontologies and vocabularies in improving the quality and quantity of Semantic Web data.

In 2008, [Hausenblas et al. \(2008\)](#) attempted to empirically **gauge the size of the Semantic Web** when the surge in RDF data was gaining attraction. This was also the first work in which authors tried to study schema level data and instance level data to understand the hidden patterns they hold. Instance level data was further classified into single-point-of-access and distributed datasets. In their analysis, they report on the number of triples available, the frequency of the subject, object and predicates and the level of external linkage (external linkage refers to the triples in which the subject and object refers to the resource hosted in different domain names. They found that FOAF data is well connected internally and sparsely with external resources. Though they did not mention any effective size of the Semantic Web, they provided an estimate on how well the Semantic Web is linked and what type of datasets are available.

In another study, [Mika et al. \(2009\)](#) identified the **semantic gap** which is essentially the divide between the supply of the data on the Semantic Web and the demand of a typical web user. They provided a generic method to extract the attributes that Web users are searching for regarding particular classes of entities. Through this, they contrast class definitions found in Semantic Web vocabularies with the attributes of objects in which users are interested. The was conducted on data comprising different data formats, such as eRDF, RDFa data and certain popular microformats. Although they argued that RDFa is becoming more popular compared with other formats, in their dataset, RDFa was 0.6%, much less popular than the other formats, particularly hcard which was the most popular during 2008 and 2009. Their work found that Semantic Web technologies could play an important role in web searches if web sites published structured information to target a particular category of queries.

Aside from looking at the **data quality issues** in RDF data, Semantic Web

effective size and **interlinking** between decentralized datasets, identifying the **semantic gap** between the available data and users expected (web search) results, several researchers also looked at the generic **characteristic** of Semantic Web data, described as follows.

Semantic Web data includes a wider range of topics such as quantifying the RDF data patterns (Hogan et al., 2010), analysing the distribution of schema level details in RDF documents (Hausenblas et al., 2008), and the statistical properties of the LOD cloud (Bizer et al., 2011). However, the early work on **characterizing Semantic Web data** on the Web was reported by Ding and Finin (2006). In this work, they estimated the number of RDF documents available on the Web, based on the search engine result pages (SERP) returned by Google. The estimated number of RDF documents found at that time (i.e 2006) was in the range of $10^7 - 10^9$. The authors also provided an in-depth analysis, conducted on 300M triples (mostly consisting of FOAF with some RSS1.0 documents), on the landscape of RDF web data, including the number of files, provenance in terms of website, use of RDFS primitives and use of class and properties. They found that 2.2% of classes and properties had no definition and that 0.08% of terms had both class and property meta-usage. Other work based on a similar analysis approach but from Linked Data perspectives is reported by Hausenblas et al. in (Hausenblas et al., 2008). The motivation of their study was based on the argument that understanding the size of the current Semantic Web is critical to the development of scalable Semantic Web applications. Empirical analysis comprising syntactical and semantic aspects was conducted on a LOD dataset which was viewed as an interlinked (single-point-of-access) dataset and distributed (FOAF-o-sphere) datasets. For the interlinked datasets, they reported on the number of triples available and automatic interlinking which yields a high number of semantic links but of shallow quality. As such, no quantitative measure is reported by authors to size the Semantic Web data but this helps to understand the importance of creating semantic links among distributed datasets.

In the aforementioned sub-section, the literature in which RDF data is analysed from different dimensions, such as **quality, data patterns, structural properties, interlinking** and general **characteristics** of datasets is described. The insight into how RDF data, in general, is being published and the state of its quality is very useful, however, it is equally important to understand how vocabularies and ontologies are being used on the Web. An overview of the literature in which schema level data, which includes ontologies and vocabularies, are analysed and assessed is described in the following section.

2.3.2 Empirical Analysis of Ontologies and Vocabularies in RDF Data

In this sub-section, the reported work on the use of ontologies, including both **W3C vocabularies** (i.e RDF, RDFS, OWL) and **domain ontologies** on the Web is discussed.

A large amount of work has been reported on evaluating the usage of W3C-based standard vocabularies. [d'Aquin et al. \(2007\)](#) surveyed 1300 OWL ontologies and RDF schemas and reported some of the trends observed during the investigation. They observed that most of the ontologies from the OWL family are OWL DL and OWL Lite ontologies. [Cheng et al. \(2011\)](#) conducted a study on roughly 3000 vocabularies, comprising 396,023 classes and 59,868 properties in total. In addition to vocabulary documents, the authors also considered 15 million instance documents to investigate the relatedness between ontologies. They reported that 72% of vocabularies contain no more than 24 terms and also investigated the relatedness indicators between vocabularies, the textual content of vocabularies, and the explicit linking among vocabularies.

[d'Aquin et al. \(2007\)](#) reported on the characterization of the Semantic Web, based on the WATSON repository which represents a snapshot of the online semantic documents available during 2006. One of their findings is similar to the one we reported in ([Ashraf et al., 2011](#)), that a large number of small and lightweight ontologies are used with some instances where large scale heavyweight ontologies are used. They highlighted the need for an effort to improve the quality and usefulness of existing ontologies and the need to develop ontologies for diverse domains.

Also reported in the earlier sub-section, [Ding et al. \(2005\)](#) collected 1.5 million RDF/XML documents from the Web and reported on the use of different namespaces and the concepts and properties defined by these namespaces. Their particular emphasis was on the documents in which information (data) is semantically described with FOAF and DC vocabularies. They found that the majority of the RDF data is published by a few of the social network sites such as `livejournal.com`, `academy.com` and `deadjournal.com`. Aside from reporting that a large amount of RDF data is, in fact, published by only a few of the data publishers, they also analysed the network properties of the FOAF network such as connected components and the distribution of nodes. They detected various forms of Zipf distribution such as the number of `foaf:Person` described in each document and the number of aliases found by using `foaf:mbox_sha1sum` predicate.

[Ding and Finin \(2006\)](#) presented another analysis conducted on 300 million RDF

triples, on the use of FOAF and DC vocabularies on the Web. They described the number of global metrics, properties and usage patterns for the study and also observed the presence of the power law distribution of different metrics. They reported that most of the classes (>97%) are not instantiated on the Web (this means, not used to define instances) and likewise, more than 70% of the properties are also not used to describe resources. One of the significant contributions of their work was a discussion on whether or not the traditional monolithic ontologies are the best solution for the Web or should the research community proceed with lightweight vocabularies and encourage maximization of reusability of existing vocabularies.

Another important focus in analysing the schema level data on the Web is to look into the use of **standard W3C meta-vocabularies** such as RDF, RDF Schema and OWL/OWL2. In the following, several of the studies which empirically analyse the use of such vocabularies are presented. For example, several researchers investigated the use of the `owl:sameAs` predicate which allows two resources to refer to the same things, on the Web (meaning that two co-referent resources talk about the same real-world object).

[Ding et al. \(2010b\)](#) presented work in which they explored the presence of `owl:sameAs` to combine and retrieve additional information during crawling. They reported on quality issues observed, such as the casual use of `owl:sameAs` without giving due attention to ensure that the symmetric semantic of `owl:sameAs` can, in fact, create a lot of Web discrimination.

[Hogan et al. \(2012\)](#) looked at the **co-referencing issues** keeping in mind the OWL features that allow inferences, including inverse functional property and functional property. They explored the use of `owl:sameAs` in the same dataset by computing inference closure and found that URIs with at least one alias had an average of 2.65 aliases due to the incorrect use of `owl:sameAs` linkage. They also reported that 57% of alias groups contained URIs from multiple domains. They also looked at the implicit `owl:sameAs` relations which were produced through inferencing over inverse-functional properties, finding that the majority of additional aliases had blank-nodes coming from the same domain. Overall, the finding was that the quality of Linked Data is high if undertaken carefully otherwise it could be a burden on the applications which consume such data.

In recent work, [Cheng et al. \(2011\)](#) performed an empirical study on a dataset collected from 261 pay-level-domains comprising 2,996 vocabularies. These vocabularies further contain 396,023 classes and 59,868 properties. They took 15 million instance documents to investigate the relatedness between vocabularies, measured with respect to how terms are defined, the textual content of vocabularies

and co-occurrence in instance documents. They also looked at the relationship between relatedness and popularity and its usage for Falcons ontology search recommendation service. The significant finding of their work is that several related vocabularies are not interlinked and those which are interlinked are often co-used in the same instance document.

In a previous study, (Ashraf et al., 2011) analysed the usage and adoption of the GoodRelations ontology in eCommerce domain. To base the findings on empirical ground, a purpose-built dataset was used containing RDF (most represented using RDFa) of the data from 105 different pay-level domains. The co-usability factor of the domain ontology with other ontologies is analysed to observe how different vocabularies are being co-used to semantically describe the entities (pertinent to eCommerce domain). Using real use cases, the use of different object properties and attributes of pivotal concepts (`gr:Offering`, `gr:BusinessEntity` and `gr:ProductOrService`) are analysed to understand the data and knowledge patterns available on the Web (in eCommerce domains). One of the findings of this works is that a small portion of the ontology is hugely used by a large number of data sources. This supports the previous findings and recommendations (Ding and Finin, 2006; Hogan et al., 2010) that Web ontologies, in order to be successful on the Web, should be small Web ontologies rather than monolithic ontologies.

2.4 Critical Evaluation of the existing work on Analysing Ontology Usage

In this section, a critical evaluation of the existing approaches in the literature is presented and the main issues that need to be addressed for measuring ontology usage are identified. As can be seen from the discussions in the previous sections, that there are several approaches in the literature proposed by different researchers by which *ontologies are developed* (Asunción Gómez-Pérez et al., 1996; Uschold and King, 1995; Sure, 2002) and the *RDF data is analysed* (Ding and Finin, 2006; Hausenblas et al., 2008; Hogan et al., 2010) from different perspectives. Relating to the development of ontologies, approaches have been proposed to *develop* (Mizoguchi and Ikeda, 1996; Jarrar and Meersman, 2002; Sure et al., 2002a) and *evaluate ontologies* (Brewster et al., 2004; Brank et al., 2005; Gangemi et al., 2005a; Tartir and Arpinar, 2007) and measure their quality and assess their compliance to the requirements. Also approaches have been proposed to *evolve ontologies* (Noy and Klein, 2004; Vrandevcic, 2010) to implement change management to ensure ontologies remain useful and adapt

to the new requirements.

However, none of the approaches analysed ontologies based on their usage in a real world setting which results in bringing the *using* element of ontologies to the fore. This means that while evaluating ontologies and analysing Semantic Web data, after an ontology has been developed, there is a need to bring the “usage” aspect of ontologies and RDF data to the equation to better understand their adoptability and uptake in the actual instantiation. Similarly, in the literature, RDF data has been analysed to assess the quality (Hogan et al., 2010) and understand the use of W3C-based vocabularies’ constructs (Hogan et al., 2012) in the published instance data. The approaches analysed the instance data but not from the perspective where the use of different domain ontologies is measured. Therefore, while analysing the RDF data from quality perspectives, it should be analysed from the ontology usage perspective to measure the semantic level information in the instance data. The availability of such insight will help to provide pragmatic insights about the state of ontology usage, its adoption level and develop evolution strategies. So, the shortcomings in the existing literature related to ontology usage analysis that have been identified are:

- Most of the ontology lifecycle models are centered on the construction and evaluation of ontologies (Asunción Gómez-Pérez et al., 1996; Tran et al., 2008; Möller, 2012). Here, the emphasis remains on developing approaches for ontology development which closely match their anticipated “usage” and hence once they are developed, they are evaluated according to this factor. But, no emphasis is given to actually evaluating the “usage” of the developed ontologies in real world settings from the viewpoint of their instantiation.
 - Most of the ontology evaluation approaches only consider the (formalized) conceptual model to evaluate ontologies (Guarino and Welty, 2004; Alani et al., 2006; Dasgupta et al., 2007). Since ontology evaluation, in most methodologies, is considered part of the development phase in order to measure their effectiveness of developed ontology, “usage” is not measured due to their lack of implementation in real world applications. Therefore, the concept of “usage” often refers to the evaluation of the use of different constructs to describe the concepts and other components of the ontologies and not their “usage” in annotating information.
 - Most of the RDF analysis work focuses on analysing the quality aspect of published RDF triples (Ding and Finin, 2006; d’Aquin et al., 2007; Hausenblas et al., 2008). Here, the “usage” concept is used to analyse the different
-

W3C-based vocabularies and their compliance with the linked data principles but do not analyse the “use” of domain ontologies in semantically describing the information.

- Most of the work in which ontologies are analysed study the structural and typological aspects of the ontology graph (Mika, 2005; Ding et al., 2010b; Cheng et al., 2011; Erétéo, 2011). Here, “usage” is again considered from the point of evaluating how the concepts are hierarchically arranged in the ontology graph but do not provide any insight on the “usage” of those ontologies by creating relationships with ontology users.

Considering the abovementioned observations, the main shortcomings of the existing approaches in the literature pertaining to measuring ontology usage are identified as follows.

1. Lack of a definition to describe ontology usage analysis.
2. Ontology usage has not been positioned as an area in the ontology engineering lifecycle.
3. There is no methodological approach proposed toward ontology usage analysis
4. Lack of methods and techniques to measure ontology usage
5. Lack of a model to conceptually represent ontology usage analysis and make it accessible to others so that its analysis can be considered in the different areas of ontology engineering.

Each of these points is discussed in detail in the following subsections.

2.4.1 Lack of a definition to describe Ontology Usage Analysis

As discussed in Sections 2.2 and 2.3, in the present literature, extensive work has been done on knowledge representation and several approaches and methodologies have been proposed (in the early days of Semantic Web (circa 1999-2006)) to develop and maintain ontologies. As a result of these efforts, ontologies have been developed in a huge quantity but their application is somewhat limited. Due to the lack of their application in real world scenarios, their instantiation was inadequate to provide the actual instance data needed for the evaluation and analysis of ontologies.

In order to overcome this situation, test data was often used to perform the evaluation of ontologies (Tao et al., 2009b). Therefore, the focus in the early days of ontology-specific research was centered around building methodologies and a formal model for ontologies and limited focus was given to the utilization of ontologies (Auer and Lehmann, 2010).

As highlighted in Section 1.4, recently the focus has shifted toward publishing data using domain ontologies on the Web. This shift is credited to few things such as the recognition of explicit semantics by search engines and the simplicity of Linked Data principles. With numerous ontologies and their instantiation generating a large number of triple, it now provides a platform to actually analyse the use of ontologies. Consequently, the presence of these triples raises the need to consider evaluating the "use" of ontologies and measure usage as it provides a usage-based feedback loop to the ontology lifecycle model, make effective and efficient use of formalized knowledge and insight on the state of semantic structured data.

Therefore, there is a need to have a specific area of research focusing on the "usage" aspect of ontologies. This requires some form of formalization to precisely define the area and its scope. Defining ontology usage as a focused area will help to identify the work that needs to be carried out to achieve the required objectives. In order to bridge the gap, in Chapter 3 the need for ontology usage analysis is presented and in Chapter 4 the definition of ontology usage analysis is presented to specify its scope and highlight the key terms representing its definition.

2.4.2 Ontology Usage has not been positioned as an area in the Ontology Engineering Lifecycle

Ontology engineering, as shown in Figure 2.1, represents a group of activities geared toward the development of ontologies. In the normal course of action, ontologies once developed are then evaluated using ontology evaluation techniques (discussed in Section 1.3.3). To ensure ontologies remain useful, ontology evolution (discussed in Section 1.3.5) which supports the evolution process of the ontologies is used. These major activities and other supporting activities provide an ecosystem in which ontologies grow from their inception to their implementation. In the literature, various approaches for different areas of ontology engineering have been proposed. However, as mentioned in Section 2.4.1, these areas were developed when the focus was on knowledge representation which has shifted now to publishing data. Therefore, the implemented ontologies need to experience their utilization in order to receive the benefits, which comes through their usage

In order to provide a detailed definition of ontology usage, it is important to discuss its role through the reference of other related (existing) activities such as ontology evaluation and evolution. In addition to this, it is important to specify the position of ontology usage within the ontology lifecycle model to understand its application and the stage at which it is applicable.

The abovementioned discussion highlights the need to specify the relationship of ontology usage with other related activities and its placement within the ontology lifecycle model, which has not been examined in the literature. To address this, in Chapter 3, the need to include ontology usage as an area of ontology engineering is defined. In order to provide a solution, in Chapter 4, ontology usage analysis is discussed by specifying its relationship with other activities of the ontology engineering lifecycle, such as ontology evaluation and evolution.

2.4.3 There is no methodological approach for Ontology Usage Analysis

After defining ontology usage analysis and positioning it with other relevant areas of ontology engineering, there is need for its evaluation and implementation. In order to support the implementation of ontology usage analysis, a methodological approach is required which provides the guidelines to carry out usage analysis in a systematic and repeatable way. To achieve this, the identification of the major “stages” that can facilitate an integrated series of activities to analyse the usage of ontologies need to be objectively identified, followed by the order in which they are required to operate need to be specified.

In order to support the different techniques and methods for different stages of the envisioned approach, each stage needs to be specified in reasonable detail to facilitate the development of techniques and methods. The stages need to cover the initiation phase to bootstrap the analysis activity, the execution phase to perform the analysis and the implementation phase to obtain the results of usage analysis. No existing approaches in the literature identify the stages needed for ontology usage analysis and the series of activities for these stages. Therefore, in Chapter 3, the need for such a methodological approach is presented and its high level requirements are discussed. In Chapter 4, the proposed solution comprising the stages which provide the flow of activities is presented.

2.4.4 Lack of methods and techniques to measure Ontology Usage Analysis

Once the different stages for usage analysis are identified, the methods and techniques required for the implementation of the each stage need to be defined. For the implementation of each stage, their input, output, and the processes which will manipulate the input data and perform the required operations need to be identified. Aside from considering the techniques, it is also necessary to consider the communication requirements between stages.

The identification of different techniques for each stage heavily depends on the perspectives from which ontologies need to be analysed. Each perspective has certain technical requirements to address in order to provide perspective-based analysis. This means that after the identification of stages and their specification, one needs to consider the perspectives from which the analyses are performed.

In the literature, to the best of my knowledge, no approach has been proposed which empirically and quantitatively measures the use of ontologies from different aspects and dimensions. The lack of methods and techniques to measure and analyse ontology usage is described in Chapter 3 and in Chapter 4, the solution is presented in which different methods and techniques are proposed to systematically measure the use of ontologies.

2.4.5 Lack of a model to conceptually represent Ontology Usage Analysis and make it accessible to others

The aim of the methodological approach for ontology usage analysis is to measure and analyse the use of ontologies in order to provide the feedback loop to the ontology lifecycle model and provide quantitative insight into the use of ontology and its components for different types of users. So, once the methodology is proposed and its stages are identified, there is a need for a formal mechanism to represent the output of the obtained usage results for the respective users. The proposed formal mechanism should provide the required granularity to enable different types of users to access the information applicable to their role. For example, data publishers need to know what terms to use to semantically describe information and on the other side, ontology developers/owners are interested to know the usage level of specific terms in their ontology.

To the best of my knowledge, no conceptual model is proposed in the literature which represents the domain of ontology usage and its usage analysis. In Chapter

3, the need for a conceptual model which represents and formalizes ontology usage and analysis domain knowledge is presented. The solution in the form of a formalized conceptual model represented using a formal approach is presented in Chapter 4.

2.5 Conclusion

In this chapter, a survey of the existing literature relevant to the work of ontology usage analysis is presented. Two streams of work are presented: the first stream covers the work in which “usage” from the ontology perspective is covered; and in the second stream, “usage” from the RDF (Semantic Web) data perspective is covered. Then, under each category, the relevant literature is discussed to provide the necessary background and context to support the gaps identified pertaining to ontology usage analysis. Literature is then summarised by identifying the gaps in the critical evaluation. Each identified gap and the possible approach to address this is discussed to provide guidelines for subsequent chapters to propose the solution.

In the next chapter, the problem which is being addressed in this thesis is formally described and issues arising from the main problem and sub-problems are discussed.

Chapter 3 - Problem Definition

3.1 Introduction

As mentioned in Chapter 1, the vision of the Semantic Web is to extend the current Web in such a way that it makes data located anywhere on the Web accessible and understandable to both people and machines. Having such a model will allow machines to use data not only for display purposes but for automation, integration, reasoning, intelligent processing and reuse across various applications (Fensel et al., 2002). Ontologies, which are the main component of the Semantic Web, provide the formalized mechanism to associate semantics with the data that is published on the Web.

Ontologies are developed by domain experts or ontology engineers following an appropriate ontology development methodology. In Chapter 2, some of the common ontology development methodologies are discussed and highlighted the different approaches and methods being used by the community. Although each approach may be different, they follow the ontology lifecycle model that broadly comprises two stages, namely *engineering (also known as the development stage)* and *usage (also known as the In-Use stage)* as shown in Figure 1.4. In the literature, while the development stage is largely explored by ontology engineering which has different ontology development methodologies, evaluation and evolution frameworks, the In-Use stage which mainly covers ontology use is largely unexplored. As discussed in Chapter 2, due to the early concentration of the research community on the Semantic Web and ontologies, the focus has largely remained on ontology construction chores (i.e. the development stage), whereas the post construction facets (i.e the usage stage) which are crucial to the realization of ontology utilization have not been addressed, thus leaving a gap which hinders the adoption of ontologies and hence the Semantic Web and also results in a missing link in the ontology lifecycle, as shown in Figure 1.5.

In order to address this gap, in in this chapter, the problem that will be the

focus of this thesis is described in detail. Specifically, the problem focuses on the run-time stage of the ontology lifecycle with an view to understanding how ontologies are being used and exactly what is being used from a given ontology to describe the data on the Web. This will help in obtaining insight on how Semantic Web data is produced/generated, which can then provide feedback on the ontology development process and usage patterns to the data publishers and encourage ontology reuse.

This chapter is organized as follows. In Section 3.2, the set of definitions used throughout the thesis is presented. In Section 3.3, after providing a brief background, the research motivation and the definition of the problem to be addressed in this thesis is given. Section 3.4, the problem defined in the previous section is further broken down into different issues that need to be resolved in order to propose a solution for the problem presented in Section 3.3. The research methodology followed to address the research issues and formulate a solution is presented in Section 3.5. Finally, in Section 3.6, the conclusion of the chapter is presented.

3.2 Key concepts

In this section, the definition of the terms used in this and the rest of the chapters of this thesis are presented. For more details on these terms and other terms which are used but are not defined in this chapter, readers can refer to the following resources¹.

ABox

An ABox (for assertions; the basis for A in ABox) is an assertion component; that is, a fact associated with a terminological vocabulary within a knowledge base. ABox are TBox-compliant statements about instances belonging to the concept of an ontology.

Attributes

These are the aspects, properties, features, characteristics, or parameters that objects (and classes) may have. They are the descriptive characteristics of a thing. Key-value pairs match an attribute with a value; the value may be a reference to another object, an actual value or a descriptive label or string. In an RDF statement, an attribute is expressed as a property (or predicate or relation)

¹<http://www.mkbergman.com/1017/glossary-of-semantic-technology-terms/>
<https://wiki.base22.com/display/btg/Glossary+of+Semantic+Web+Terms>
http://swoogle.umbc.edu/index.php?option=com_swoogle_manual&manual=glossary
http://agtrivity.com/semantic_web_glossary.htm
<http://semanticalley.com/semanticwebglossary/>
<http://blogs.ubc.ca/dean/2010/09/aglossaryforweb30thesemanticweb/>
(Retrieved at 20 Aug 2012)

Axiom

An axiom is a premise or starting point of reasoning. In an ontology, each statement (assertion) is an axiom.

Class

A class, in general, is a representation of a concept. It is an abstract representation of some specific classification of things (hence the name class). The name used to identify a class is the perceptual symbol or word used to denote a concept. In an ontology, a class is more specifically a formal definition of a type of information object that may possess a given set of attributes or properties and specific types of relations to other things. The ontology class is the template for an instance or individual of that type. In other words, the class is the schema or model for information of a given type while an instance of the class is considered to be the actual data.

Domain ontology

Domain (or content) ontologies embody more of the traditional ontology functions such as information interoperability, inferencing, reasoning and conceptual knowledge capture of the applicable domain.

Dataset

An aggregation of similar kinds of things or items, mostly comprising instance records.

Dublin Core (DC)

Dublin Core (DC) (<<http://dublincore.org/>>) is a metadata standard created by the Dublin Core Metadata Initiative (DCMI); it provides a semantic vocabulary for describing the core properties of digital objects.

Entity

An individual object or member of a class; when affixed with a proper name or label it is also known as a named entity (thus, named entities are a subset of all entities).

FOAF

FOAF (Friend of a Friend) is an RDF schema for machine-readable modeling of homepage-like profiles and social networks.

Individual

An object or instance of a class.

Inferencing

Inference is the act or process of deriving logical conclusions from premises known or assumed to be true. The logic within and between statements in an ontology is the basis for inferring new conclusions from it, using software applications known as inference engines or reasoners.

Instance

Instances are the basic, ground level components of an ontology. An instance is an individual member of a class, also used synonymously with entity. The instances in an ontology may include concrete objects such as people, animals, tables, automobiles, molecules, and planets, as well as abstract instances such as numbers and words. An instance is also known as an individual, with member and entity also used somewhat interchangeably.

Knowledge base

A knowledge base (abbreviated KB or kb) is a special kind of database for knowledge management. A knowledge base provides a means for information to be collected, organized, shared, searched and utilized. Formally, the combination of a TBox and ABox is a knowledge base.

Linked data

Linked data is a set of best practices for publishing and deploying instance and class data using the RDF data model, and uses uniform resource identifiers (URIs) to name the data objects. The approach exposes the data for access via the HTTP protocol, while emphasizing data interconnections, interrelationships and context useful to both humans and machine agents.

Mapping

Mapping is a considered correlation of objects in two different sources to one another, with the relation between the objects defined via a specific property. Linkage is a subset of possible mappings.

Metadata

Metadata (meta content) is supplementary data that provides information about one or more aspects of the content at hand such as means of creation, purpose, when created or modified, author or provenance, where located, topic or subject matter, standards used, or other annotation characteristics. It is data about data, or the

means by which data objects or aggregations can be described. In contrast to an attribute, which is an individual characteristic intrinsic to a data object or instance, metadata is a description about that data, such as how or when created or by whom.

Ontology

An ontology is a data model that represents a set of concepts within a domain and the relationships between those concepts. Loosely defined, ontologies on the Web can have a broad range of formalism, expressiveness or reasoning power.

OWL

The Web Ontology Language (OWL) is designed for defining and instantiating formal Web ontologies. An OWL ontology may include descriptions of classes, along with their related properties and instances. There are also a variety of OWL dialects.

Property

Properties are the ways in which classes and instances can be related to one another. Properties are thus a relationship, and are also known as predicates. Properties are used to define an attribute relation for an instance.

RDF

Resource Description Framework (RDF) is a family of World Wide Web Consortium (W3C) specifications originally designed as a metadata model but which has come to be used as a general method of modeling information, through a variety of syntax formats. The RDF metadata model is based upon the idea of making statements about resources in the form of subject-predicate-object expressions, called triples in RDF terminology. The subject denotes the resource, and the predicate denotes traits or aspects of the resource and expresses a relationship between the subject and the object.

RDFa

RDFa 1.0 is a set of extensions to XHTML that is a W3C recommendation. RDFa uses attributes from meta and link elements, and generalizes them so that they are usable on all elements, allowing annotation mark-up with semantics. A W3C working draft is presently underway that expands RDFa into version 1.1 with HTML5 and SVG support, among other changes.

RDF Schema

RDFS or RDF Schema is an extensible knowledge representation language, providing

basic elements for the description of ontologies, otherwise called RDF vocabularies, intended to structure RDF resources.

Reasoner

A semantic reasoner, reasoning engine, rules engine, or simply a reasoner, is a piece of software able to infer logical consequences from a set of asserted facts or axioms. The notion of a semantic reasoner generalizes that of an inference engine, by providing a richer set of mechanisms.

Reasoning

Reasoning is one of many logical tests using inference rules as commonly specified by means of an ontology language, and often a description language. Many reasoners use first-order predicate logic to perform reasoning; inference commonly proceeds by forward chaining or backward chaining.

Semantic Web

The Semantic Web is a collaborative movement led by the World Wide Web Consortium (W3C) that promotes common formats for data on the World Wide Web. By encouraging the inclusion of semantic content in web pages, the Semantic Web aims at converting the current Web of unstructured documents into a web of data. It builds on the W3Cs Resource Description Framework (RDF).

SPARQL

SPARQL (pronounced sparkle) is an RDF query language; its name is a recursive acronym that stands for SPARQL Protocol and RDF Query Language.

Statement

A statement is a triple in an ontology, which consists of a subject predicate object (S-P-O) assertion. By definition, each statement is a fact or axiom within an ontology.

Subclass

The child of a parent class. In OWL, subclass means necessary implication. In other words, if Child is a subclass of Person then ALL instances of Child are instances of Person, without exception if something is a Child then this implies that it is also a Person.

Subject

A subject is always a noun or compound noun and is a reference or definition to a

particular object, thing or topic, or groups of such items. Subjects are also often referred to as concepts or topics.

TBox

A TBox (for terminological knowledge, the basis for T in TBox) is a terminological component; that is, a conceptualization associated with a set of facts. TBox statements describe a conceptualization, a set of concepts and properties for these concepts. The TBox is sufficient to describe an ontology (best practice often suggests keeping a split between instance records and ABox and the TBox schema).

Taxonomy

In the context of knowledge systems, taxonomy is the hierarchical classification of entities of interest of an enterprise, organization or administration, used to classify documents, digital assets and other information. Taxonomies can cover virtually any type of physical or conceptual entities (products, processes, knowledge fields, human groups, etc.) at any level of granularity.

Triple

A triple is a basic statement in the **RDF** language, which is comprised of a subject property object construct, with the subject and property (and object optionally) referenced by **URIs**.

Type

Used synonymously herein with **Class**.

URI

A Uniform Resource Identifier (URI) is a compact sequence of characters that identifies an abstract or physical resource. A Uniform Resource Identifier (URI) is a string of a standardized form that allows the unique identification of resources (e.g., documents). A subset of URI is a Uniform Resource Locator (URL), which contains access mechanism and a (network) location of a document - such as <http://www.example.org/>. An international variant to URI is the Internationalized Resource Identifier (IRI) which allows the use of Unicode characters in the identifier and for which a mapping to the URI is defined. In the rest of this text, whenever URI is used, IRI can be used as well as a more general concept.

Vocabulary A vocabulary, in the sense of knowledge systems or ontologies, is a controlled vocabulary. These provide a way to organize knowledge for subsequent

retrieval and are used in subject indexing schemes, subject headings, thesauri, taxonomies and other forms of knowledge.

3.3 Problem Definition

As mentioned in earlier chapters, to take advantage of the benefits offered by the Semantic Web, several disciplines and vertical industries are developing Web ontologies with the anticipation that they will become a de facto standard to conceptually represent their respective domain models. Consequently, the Semantic Web community have seen the emergence of Web ontologies in diverse domains such as in Healthcare and Life Science (HCLS) (Ruttenberg et al., 2007; d'Aquin and Noy, 2012), government², social spaces (Formica and Missikoff, 2002; Breslin et al., 2006), libraries (Gradmann, 2005), entertainment (Raimond et al., 2007), financial services (Garcia and Gil, 2009), eCommerce (Hepp, 2008), and academia (Tokosumi et al., 2006)^{3,4}, etc. As shown in Figure 3.1, the use of domain (Web) ontologies is also in continuous rise, as depicted by the number of vocabularies/ontologies indexed by PingTheSemanticWeb.com index. The presence of such schema level (meta-) data describing the instance data promotes consistent and coherent semantic interoperability between users, systems and exchange data.

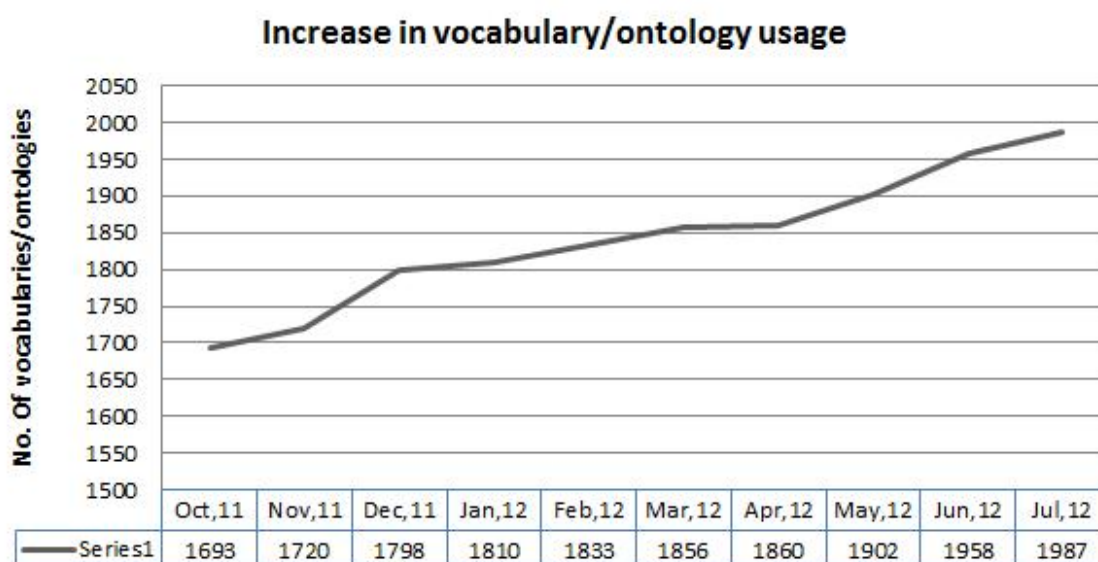


Figure 3.1: PingTheSemanticWeb.com Index of vocabulary and ontology usage

²<http://oegov.org/> & <http://dvcs.w3.org/hg/gld/raw-file/default/dcat/index.html> (retr. 12 July 2012)

³http://colab.mpg.de/mediawiki/images/8/84/ESci08_Sem_3_Ontology_for_Academic_Disciplines_Stricker.pdf; retr. 12/7/2012)

⁴<http://vocab.org/aiiso/schema>; retr. 12/7/2012)

In parallel to the growth in ontologies, there is also continuous growth in the proliferation of RDF data on the Web. Significantly, Googles announcement (Steiner and Hausenblas, 2010) that it will start recognizing the presence of structured data embedded within web documents in their index provided a long needed and awaited motivation for data publishers to publish their information in a structured data format on the Web. After realizing the benefits of semantically annotated structured data and partly seeing the incentive offered by the search engines (e.g. Google), various web publishers have adopted RDF and Linked Data principles as a means to publish data on the Web. As is the case with ontologies, several vertical industries and institutions such as government entities (UK⁵, USA⁶, Australia⁷ and others⁸), enterprises (BestBuy (Breslin et al., 2010) and FreeBase (Bollacker et al., 2008)), biomedical and healthcare (such as PubMed (Doms and Schroeder, 2005) and LinkedCT (Hassanzadeh et al., 2009)), social network sites (Facebook (Graham and Graham, 2012)), content management systems (CMS) (Drupal (Corlosquet et al., 2009)), eCommerce ((Ashraf et al., 2011)^{9,10}) are publishing structured data on the Web, primarily using the RDF data model. As evidence of its continuous growth, the latest estimation of the LOD snapshot (Figure 3.2) reports the presence of approximately 31 billion triples, which does not include the structured data embedded within Web documents either using Facebook Open Graph¹¹, Twitter Card¹² and GoodRelations snippets data¹³.

⁵<http://data.gov.uk/>; retr. 17/06/2012.

⁶<http://www.data.gov/>; retr. 02/5/2012

⁷data.gov.au; retr. 12/9/2012

⁸To access the updated and extended list of countries participating in Open Data initiative, visit <http://logd.tw.rpi.edu/>; retr. 4/7/2012, and to obtain the initial analysis visit http://logd.tw.rpi.edu/iogds_data_analytics; retr. 6/7/2012

⁹lists 105 web sites publishing eCommerce-related data in RDFa syntax

¹⁰<http://wiki.goodrelations-vocabulary.org/References>

¹¹Graph Protocol, Facebook Inc. (2011),<http://developers.facebook.com/docs/opengraph/>; retr. 15/8/2012)

¹²Twitter Cards (2012), <https://dev.twitter.com/docs/cards> (retr. 15/8/2012)

¹³<http://wiki.goodrelations-vocabulary.org/Datasets> (retr. 15/8/2012)

with little¹⁵ or no use of ontologies (Jain et al., 2010). This shift in focus is credited to the Linked Open Data (LOD) Project which has published billions of assertions on the Web using well known Linked Data principles. The Semantic Web research (with these advancements) has reached the point where the pendulum has swung from *'knowledge-centered'* to *'data-centered'* and is now settling down at the point where domain ontologies are being used to publish real-world data on the Web.

Therefore, currently, we are experiencing a unique situation; that is, witnessing an increasing use of ontologies (existing or newly developed) in RDF data on the Web. However, we do not have any analysis on ontology use to help us understand this. In other words, we do not have an analysis that shows how well these ontologies can represent the currently used real world data. As discussed in the previous chapter, most of the relevant work pertaining to ontology construction, evaluation and evolution primarily focuses on ontologies without much involvement of (real world) instance data in to the loop. For example, most of the ontology evaluation approaches presented in Chapter 2 do not include instance data in their evaluation with the exception of a few approaches (Tao et al., 2009b) in which test data was generated to include data in ontology evaluation. The work on ontology engineering methodologies and evaluation techniques has definitely reinforced the ontology construction process, but due to their inadequate instantiation, no actual instance data was available for these approaches to consider. In this respect, while ontology development and evaluation methodologies which are essential components of ontology engineering were carried out, their coverage is somewhat restricted to test the effectiveness of the developed ontology which was mainly covered by the engineering stage of the ontology lifecycle model, leaving the In-Use (run-time) stage partially uncovered.

Likewise, on the RDF (instance) data front, as discussed in Chapter 2, most of the analysis studies have focused on performing statistical analysis on the utilization of RDF/RDFS vocabularies in general and learning what data patterns are available in the given corpus. In (Ding et al., 2005), vocabulary specific analysis is performed to measure the frequency of their appearance in documents and the different topological networks emerging from the implementations. In such vocabulary-focused analysis and more generic studies on understanding the quality and usage patterns (Hogan et al., 2010), the emphasis has been on understanding the current publishing practices and obtaining statistics on the use of different terms (URIs) within the corpus of data rather than establishing a comprehensive understanding of how a domain ontology is being used in real world implementations. This has resulted in the limited use of

¹⁵The Semantic Web community has recommended publishing data first and then worrying about semantics later. This reflects James Hendler's famous quote 'a little semantics goes a long way' <http://www.cs.rpi.edu/hendler/LittleSemanticsWeb.html>

existing ontologies in relation to their full potential and there is a lack of information regarding this.

Likewise, in Figure 3.1, though it is clear that there is an increase in the use of ontologies, empirical studies conducted by [Bizer et al. \(2011\)](#) and critical analysis by [Jain et al. \(2010\)](#) on LOD clouds suggest that the use of ontologies is still limited, even though numerous ontologies are present. Furthermore, one of the earliest empirical studies on RDF data ([Ding et al., 2005](#)) highlights this chronic issue and states that *“among a large number of ontologies that have been published on the Web, however, only a few are well populated”*, which highlight the need to have a more formal mechanism to understand **how ontologies are being used** to disseminate usage-related information back to the ontology lifecycle model. Having such information will be beneficial to the ontology owners to understand the applications (usage scenarios) of their ontologies and based on the obtained visibility, plan improvements in the evolution phase (if needed). Likewise, for the optimal utilization of ontologies, the end-user needs to know what is out there in order for them to benefit from the current implementation of ontologies. The availability of facts about ontologies, their components and usage patterns help in developing routines to effectively and efficiently access the (semantically annotated) structured data on the Web.

The need for such erudite insight accessible through ontology usage analysis has been reported by several researchers, recognizing the potential benefits of ontology usage. For example, in ([Zimmermann, 2010](#)), the author highlighted the need for a detailed understanding on how ontologies are being used and their level of usage in order to recommend an appropriate ontology to the data publishers or Semantic Web application developers. Ontology usage analysis can help to find answers to the following questions:

- Given several choices, which ontology should I use?
- How do I decide which ontology or term is suitable for reusability?
- Which Web ontology should be used or recommended, given that it provides the adequate terms describing my data?
- Which ontologies are prevalent and prominent in a specific application domain?

To answer the above and other similar questions, a comprehensive understanding of ontology usage is needed to provide empirical-based evidence of usage analysis.

Hence, the development of a framework to measure ontology use is pivotal in order to increase its adoption and promote its use in the Semantic Web. Research

has been conducted to better understand RDF data in general (Hogan et al., 2010; Hausenblas et al., 2008) and assess RDF data quality (Fürber and Hepp, 2010) and the use of ontologies to represent social networks (Ding and Finin, 2006). Such work has helped highlight quality-related issues, and enhance an understanding of the use of W3C-based vocabularies (RDFS, OWL) in RDF data, however, from an ontology usage perspective, there is no such work presented in literature.

In order to understand how ontologies are being used, there is a need for a more focused (or recognized) area to formalize the discipline of ontology analysis, consequently raising questions that need to be answered. In order to perform ontology usage analysis, one needs to have a **set of methods and techniques** to carry out this intensive work and achieve the required objectives.

The methods need to look into the aspect of the “usage” of ontologies by measuring factors such as instantiation, usefulness, semanticity and co-usability to obtain a more pragmatic and realistic understanding of ontology use. These methods, whether quantitative or qualitative, provide the building blocks to monitor the observable properties of the subject (ontologies in our case) by defining **different metrics** to obtain the quantified results. To measure the use of ontologies on the Web, I need to look into the aspects generating the patterns of usage by defining appropriate metrics. Metrics are required not only to understand the structural characteristic of the ontological model but also the semantic richness of its components. The structural characteristics of an ontology tell us how knowledge is being structured in the model and relate it with usage attempts to highlight the prominent knowledge patterns and unveil the relationship between usage and structure (if any). Such insight is important for ontology owners to know which structural arrangement of the ontology components are working and to learn from it to influence future thinking and design. The semantic richness of each concept helps in understanding how the entities will be describable and the possible semantic relationship and entity could have other different entities. Metrics to measure ontology use in general and of their components specifically are needed to provide a wider understanding on ontology implementations. For ontology users as well for the ontology owner, it is a key insight to learn which of the components of the ontology are more instantiated and what semantic descriptions associated with them are being used often. The metrics quantifying such understanding help users perceive the data and knowledge patterns expected to be available for consumption.

In the literature, different metrics have been proposed in the context of ontology evaluation to assess ontologies from different aspects. Tartir et al. (2005) and Sabou et al. (2006) have employed several metrics to measure the quality of an ontology by considering its structure, richness and performance. Likewise, different search engines such as Swoogle (Ding et al., 2004) and OntoSelect (Buitelaar et al.,

2004) have used popularity-based metrics to rank and compare different ontologies. The aspects which have been considered by several ontology evaluation techniques and the related problems of ontology discovery, ontology selection (Sabou et al., 2006), ontology selection, and ontology summarization (Zhang et al., 2010) include vocabulary, structure, performance, quality, annotation and semantics. While all these characteristics of an ontology are valuable and help in assessing the quality of an ontology, they do not assist in performing an empirical analysis on how ontologies are being used and the prevalent knowledge patterns to facilitate fine-grained knowledge reuse.

Once the different aspects of the ontology and its use have been measured by different metrics, there is a need to **combine the analysis to obtain a comprehensive insight** on how an ontology is being used and its level of adoption in the Semantic Web.

While the metrics help to measure and quantify the aspects need to be observed, for the dissemination of the obtained analysis to different ontology users, a **formal and structural approach is needed to represent the analysis** for its further utilization in ontology development and In-Use stages. To offer the most from the findings, the diffusion of analysis results preferably needs to be available in a structure processable by computers. Having empirical results in machine processable format will not only help in the automatic retrieval of information but also interlining with other relevant information sources.

Having a combination of such techniques will provide us with an integrated framework comprising of series of methods and processes to make ontology usage analysis computable and communicable with other components (or systems). Ontology usage analysis provides a comprehensive understanding of the prevalent knowledge patterns available in RDF data and provides the quantitative indicators needed to be made available to data publishers or ontology users.

So based on the above mentioned discussion, there is a **need of a framework** in the literature that assists in analysing and representing how ontologies are being used. The availability of such a framework comprising of a series of techniques, methods and processes assist in making ontology usage analysis communicable and computable for its implementation.

In summary, despite the fact that there is an increase in RDF data and a steady increase in the use of ontologies, the pace at which ontologies are being used remains limited. In order to realize the benefits of the Semantic Web vision, it is not sufficient to have a large number of ontologies being developed but rather, the use of ontologies on the Web should be increased. To facilitate an increase in the use of ontologies, it is

important to establish a formal approach toward understanding the use of ontologies. Hence, based on the aforementioned discussion, the problem addresses in this thesis is defined as:

to develop an approach to evaluate, measure and analyse domain ontology usage on the Web and provide a usage-based feedback loop to ontology owners and information on usage patterns and statistics to ontology users for querying (accessing) the Web-of-Data.

In order to provide a methodical solution, the above mentioned problem is divided into several sub-problems. The identification of different related sub-problems helps in understanding the problem in detail and finding the most appropriate solution to address it. The **sub-problems** are as follows:

1. Define Ontology Usage Analysis as a focused research area in the ontology lifecycle and its role and utilization. The definition should allow an understanding of the anticipated role of Ontology Usage Analysis and its utilization for different types of users such as ontology engineers, domain experts and application developers. To position it within or alongside the ontology engineering discipline, discuss its particularities and role in promoting ontology usage
 2. Define a methodology to carry out Ontology Usage Analysis. The proposed methodology should carry out the analysis on empirical grounding. The methodology should provide clear steps and define the role of each step in analysing the usage of ontologies. In order to base the analysis on a real instantiation of ontologies, a dataset should be collected and utilized to generate data and knowledge patterns.
 3. Define the set of metrics to measure ontology usage considering its relevant aspects. It is imperative to measure ontology use from qualitative and quantitative perspectives. To obtain a wider insight, the ontology should be evaluated in terms of its structural, functional and semantic aspects.
 4. Propose a formal conceptualized model to represent ontology usage. To increase the utilization of ontology usage analysis and ensure the results are accessible to both humans and machines, a formal model is needed to conceptually represent the analysis and support auto discovery and dissemination.
 5. Validate the proposed methodology by focusing on a domain-specific application area to evaluate its effectiveness.
-

3.4 Research Issues

Deliberation on the research problem and subsequent sub-problems to devise a solution has raised several issues which need attention. The resolution of these issues will help in determining the appropriate solution and the relevant methodology. In the following, each issue and its relevance to the research problem and sub-problems addressed in this thesis is discussed.

Issue 1: How can the use of ontologies be measured and analysed and is there any formal approach available for this?

As discussed in Chapters 1 and 2, the initial focus of the Semantic Web and knowledge management community was centered on knowledge and as a result, the ontology engineering discipline matured and several methodologies were proposed to develop, evaluate and evolve ontologies (Auer and Lehmann, 2010). The ontology evaluation methodologies proposed in the literature (see Chapter 2) consider only the ontology graph and assesses its structural and functional characteristics. However, the emergence of RDF data which contains resources defined in ontologies to provide semantics has provided a new type of data space comprising of schema and instance level information. Due to the primary focus of existing evaluation and analysis approaches to ontologies, the desired results cannot be simply obtained by applying them on RDF data. Therefore, there is a need for a focused study in which ontologies are evaluated from a different perspective i.e. from their usage point of view, contrary to other related disciplines such as ontology evaluation and evolution.

This thesis aims to highlight the need for such an area of study and draw a comparison with other relevant disciplines. It will discuss the requirements of this new area and define its core focus and responsibility. Chapter 4 focuses on measuring ontology usage and its role and responsibilities will be defined in subsequent chapters.

Issue 2: Lack of a methodology to perform empirical analysis on how ontologies are being used?

In order to carry out empirical analysis on how ontologies are being used and the usage patterns embedded in a given dataset, a methodological approach is needed. The availability of a methodology provides a comprehensive approach with an integrated series of techniques and methods to perform thought-intensive work for the achievement of the desired outcomes.

After defining the area of study to perform usage analysis, a methodology is needed to streamline the stages of the empirical experiments. For the implementation of the methodology, pertinent details such as the roles involved, the set of activities, and the applicable and appropriate methods are needed. Chapter 4 briefly describes

the proposed methodology which is elaborated further in subsequent chapters as the framework is implemented and the implementation phases progress.

Issue 3: What are the appropriate sets of metrics to measure ontology usage analysis from different perspectives?

It is imperative to measure what we would like to manage or improve (Drucker, 1958). The metrics used in ontology evaluation focus on ontologies, however, aspects relating to usage are missing. Therefore, there is need to identify the perspective and aspects which impact or affect ontology usage. The appropriate metrics need to be identified, defined and implemented to qualitatively and quantitatively measure the use of ontologies.

The proposed set of metrics is introduced in Chapter 4 as part of the solution overview and their implementation details are discussed further in subsequent Chapters 5,6, and 7.

Issue 4: How can usage analysis results be represented to increase its utilization?

The objective of this thesis is to propose an approach to measure the use of ontologies. The perceived output of this activity is analysis results which are then shared by different applications to encourage its utilization; thus, making usage analysis a means to an end and not an end itself. Therefore, the representation of usage results is of central importance toward the realization of analysis utilization. The conceptual representation of results needs to be modelled in such a way that it can be easily disseminated across different applications. Aside from considering the standard data model, it is also necessary to select a model which is extendable and flexible to accommodate changes in future.

In next chapter, the solution aligned with the requirements mentioned earlier will be presented and in Chapter 8, implementation details are discussed.

Issue 5: How can the proposed framework be evaluated?

To understand the applicability and usefulness of the implemented solution, it is required to evaluate the methodology implemented to address the research problem and the sub-problems, that is, research issues 1 to 4. The validation need to be based on a concrete use case to measure the effectiveness of the framework in a real implementation. Since there are several areas in which usage analysis can contribute, one which is of a more generic nature and applicable to a large audience was chosen.

The use cases are described in Chapters 9 with a detailed discussion of their implementation and obtained results.

3.5 Research approach to Problem Solving

Proposing a solution to research problems such as the ones posed in this thesis requires a world view and a holistic approach. In terms of a world view, the researcher needs to understand the area of research, define the relevant aspects of interest and acquire pertinent knowledge. In order to carry out a series of activities to establish a sound understanding of the problem, a systematic approach is required to keep the research within its boundaries and parameters, thus ensuring the research is based on well tested and trusted methods to increase its impact and share-ability with the larger community. In science research there are two broad approaches: (a) the science and engineering approach ([Galliers, 1992](#)); and (b) the social science approach ([Gomm, 2004](#)). The former approach is relevant to our discipline and will be employed for this thesis. The science and engineering-based research approach mainly supports and facilitates theoretical prediction through solution development. Research in information and computer science is populated by information artefacts which are produced as the result of solving some theoretical research problem. Following are the key steps involved in research producing information artifacts:

- Identifying the problem or realising the need
- Reviewing existing literature and identifying the gaps in the literature
- Proposing the research problem based on the need and identified gap
- Proposing the solution to the research problem
- Implementing the solution
- Evaluating the solution

The science and engineering-based research method proposed by Galliers ([Galliers, 1992](#)) suggests three levels at which research is performed, namely conceptual, perceptual and practical. The conceptual level deals with creating new ideas and concepts through analysis, the perceptual level deals with new methods and approaches to design and develop a solution; and the practical level deals with carrying out the evaluation of the new methods, approach or system through experiments, test cases, usage scenarios or through implementations.

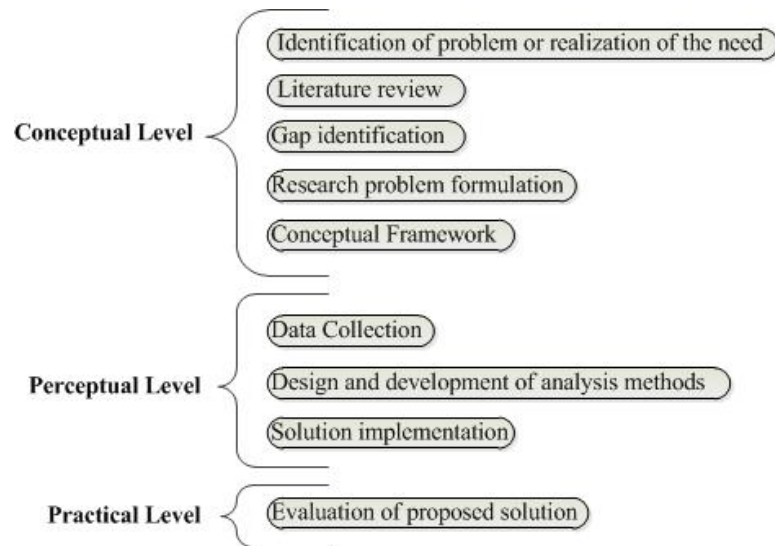


Figure 3.3: Set of activities performed and their levels according to science and engineering-based methodology

The methodology used in this research and the set of activities performed are depicted in Figure 3.3. Scientific research begins with the identification of the problem or by posing questions in the context of existing knowledge. The answers to these questions or the solution to the problem may be obtained from old theories or else a formulation of new theories is required. Therefore, to begin the research at a conceptual level after identifying the problem, the relevant literature is explored to investigate the existing body of knowledge and identify the gaps in the literature. The identified gaps provide the context to define the research problem addressed through this thesis. Research issues are identified while investigating the problem and understanding the required solution components. The last activity at a conceptual level is the development of a conceptual framework to provide a mental model of the solution. The perceptual level involves the construction of the solution, the collection of the data needed for the experiment and the implementation of the solution via the selected methods, techniques or tools. The implemented solution is then evaluated using an appropriate approach at the practical level of the research methodology.

Figure 3.4 shows a schematic representation of the corresponding chapters of the thesis in which the different activities of each level are discussed.

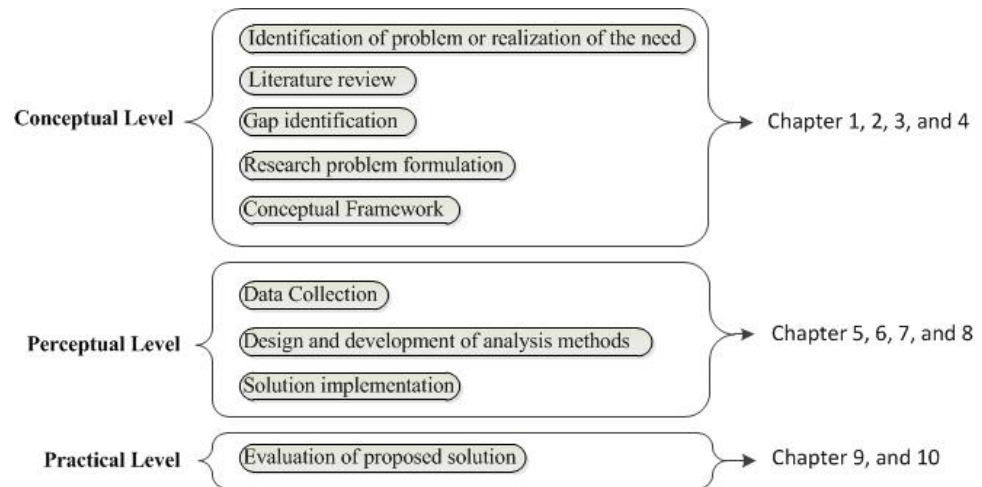


Figure 3.4: Set of activities and levels mapped with thesis chapters.

3.6 Conclusion

In this chapter, the research problem being addressed through this thesis was presented. The key terms used in this chapter and throughout the thesis were defined and the research issues which need to be tackled in order to realize the solution were presented. Further, the different research methodologies applicable to this work were discussed and the research methodology adopted for this thesis was presented.

Chapter 4 - Solution Overview

4.1 Introduction

As discussed in the previous chapter, Web ontologies are being developed and deployed to describe information in a semantically rich fashion. In order to benefit from the deployment of ontologies, it is important to understand which components of an ontology are being used and how they are being used. Such understanding can improve the utilization of Semantic Web data and allow its potential benefits to be realized ([Baker and Herman, 2009](#)).

Chapter 3 discussed that the existing literature which measures “usage”, with a focus more on understanding RDF data in general and rather than presenting an ontology usage perspective. To address this, five research issues were identified. In this chapter, an overview of the solution is presented to address the identified research issues. This chapter is organized as follows. Section 4.2 presents the preliminaries and notation which are used in this thesis. In Section 4.3, ‘Ontology Usage Analysis’ is defined and key terms used in the definition are discussed in detail. Furthermore, the placement of ontology usage in the ontology lifecycle and its relationship with the other subareas of ontology engineering is discussed. Section 4.3 describes the different phases of Ontology Usage Analysis and the purpose of each phase. In Section 4.5, the proposed framework is presented along with a discussion on its components. The conclusion of the chapter is presented in Section 4.6.

4.2 Preliminaries and Notation

In this section, the core preliminaries and notations used throughout this thesis are explained precisely. However, the detailed background and formal discussion on RDF-related terms are available in ([Hayes, 2004](#)). The models of ontology and

knowledge base used in this thesis are primarily based on (Maedche and Zacharias, 2002).

URI Reference: On the Semantic Web, all information has to be expressed as statements about Resources. Resources are identified by Uniform Resource Identifier (URI). URIs identify not just Web documents, but also real-world objects like people and cars, and even abstract ideas and non-existing things like a mythical concepts. All these real-world objects or things, in Semantic Web are called resources and URI Reference is a compact string of characters for identifying an abstract or physical resources.

RDF Term. Given the set of URI references U , the set of blank nodes B , and the set of literals L , the set of RDF terms is denoted by $RDFTerm := U \cup B \cup L$. Such that:

- The set of **blank nodes** B is a set of existentially qualified variables.
- The set of **literals** is given as $L = L_p \cup L_t$, where L_p is the set of **plain literals** and L_t is the set of **typed literals**. A typed literal is the pair $l = (s, t)$, where s is in the lexical form of the literal and $t \in U$ is a datatype URI.

The above mentioned sets U, B, L_p, L_t are pairwise disjoint.

RDF Triple. A triple $t := (s, p, o) \in (U \cup B) \times U \times (U \cup B \cup L)$ is called an RDF triple, where s is called subject, p predicate, and o object.

Data level position: Data level position identifies the position where instance data can be found on a RDF Triple. It refers to the position of the subject of a triple and object of the triple *iff* the predicate is not `rdf:type`

Schema level position: Schema level position identifies the position where schema elements (terminological knowledge) can be found in an RDF Graph. It refers to the object *iff* the predicate in `rdf:type` and predicates other than `rdf:type`.

Ontology Structure (O). An ontology structure is a 6-tuple $O := \{C, P, A, H_C, prop, att\}$ consisting of two disjoint sets C and P whose elements are called concepts and relation identifiers, respectively, a concept hierarchy $H_C : H_C$ is a directed, transitive relation $H_C \subseteq C \times C$ which is also called concept taxonomy. $H_c(C_1, C_2)$ means that C_1 is a sub-concept of C_2 , a function

$prop : P \rightarrow C \times C$, that relates concepts non-taxonomically (The function $dom : P \rightarrow C$ with $dom(P) := \Pi_1(rel(P))$ gives the domain of P , and $range : P \rightarrow C$ with $range(P) := \Pi_2(rel(P))$ gives its range. For $prop(P) = (C_1, C_2)$ one may also write $P(C_1, C_2)$). A specific kind of relations are attributes A . The function $att : A \rightarrow C$ relates concepts with literal values (this means $range(A) := STRING$).

Dataset (ontology-based metadata). A metadata structure is a 6-tuple $Dataset := \{O, I, L, inst, instr, instl\}$, that consists of an ontology O , a set I whose elements are called instance identifiers (correspondingly C , P and I are disjoint), a set of literal values L , a function $C \rightarrow 2^I$ called concept instantiation (For $inst(C) = I$ one may also write $C(I)$), and a function $inst : P \rightarrow 2^{I \times I}$ called relation instantiation (For $instr(P) = \{I_1, I_2\}$ one may also write $P(I_1, I_2)$). The attribute instantiation is described via the function $instl : P \rightarrow 2^{IXL}$ relates instances with literal values.

Data source: In this thesis, data sources refer to the unique pay-level domains (PLD) hosting RDF of some form (RDF document, RDFa snippets within HTML pages). I may interchangeably use the terms “data source”, “site”, “data provider”, “data publisher” to refer to the “data source” that hosts RDF (in any serialization and form, i.e RDFa) data.

In the next section, Ontology Usage Analysis is defined

4.3 Defining Ontology Usage Analysis

As shown in Figure 1.4, ontologies mainly go through two stages in their lifecycle. The first stage is the development stage that covers the set of activities relevant to the construction of the ontologies and their evaluation. The second stage is the in-use stage which covers the life span in which ontologies are being used to perform the intended tasks. This latter stage represents the run-time environment for ontologies which is described as Ontology Usage. Ontology Usage Analysis (OUA) is a task in this stage that provides insight into how ontologies are being used. This will lead to the better utilization of ontologies and effectual access to their instantiated data. Ontology Usage Analysis is defined as:

Definition (Ontology Usage Analysis), A study that examines the use of an ontology on the Web after it has been *instantiated* in a real world setting by measuring its *usefulness*, *usage* and *the commercial advantages* it offers.

This definition is comprised of many important terms (underlined) in measuring the usage of ontologies on the Web. Each will be explained in detail.

Instantiation

Instantiation means that a *term* defined in an ontology is being *used* in different usage scenarios (e.g. semantic annotation, knowledge representation, Semantic Web applications). The term could be a concept, or an object property (relationship), or a datatype property (attribute)). Also, the term used refers to the event when an instance of a concept type is created, or when an object property is used to relate two individuals, or when a datatype property is used to associate data values. The instantiation of ontologies provides access to a corpus of semantically rich structured data comprising schema-level and instance-level data. Since the intrinsic value of ontologies is associated with their increased reusability (Hepp, 2007), the instantiation of ontologies helps in attaining increasing perceived value and utility of ontologies in use. This provides the usage trends of ontologies to promote reusability. The (re)usability – being the utmost quality of any reusable component – of ontologies is facilitated by gaining an insight into how an ontology is actually being instantiated and used.

Usefulness

Usefulness means measuring the structural characteristics of ontological components to understand the distribution of relationships among different concepts and the attributive characteristics of the data properties. Measuring usefulness quantifies how the (ontological) model is conceptualized and organized structurally to arrange the relationship, including the taxonomical and non-taxonomical relationships with other concepts.

Usage

Usage refers to the state when an ontology is in use and it measures the statistical characteristics of the ontological components that are being used through ontology instantiation. The usage of an ontology helps in understanding how correctly the model is conceptualized to represent the real world entities and the relationship those entities hold. Usage encompasses the use of concepts, the use of object properties to create typed relationships with other entities and the use of certain attributes to describe entities.

Commercial Advantage

The *commercial advantage* quantifies the incentives available to the users of the ontology as a result of using it. This helps in incorporating the driving factors behind the adoption of the ontology by the users to further promote and encourage its reusability. It captures the benefits available to the adopters of ontologies in publishing semantically structured data on the Web.

In other words, ontology usage analysis provides qualitative and quantitative insight into how an ontology is being adopted, the common patterns of its usage in the real world setting, how useful it is and what benefits it provides.

Before presenting the methodological approach for analysing ontology usage, in the next subsection how ontology usage analysis is related with other relevant subareas of ontology engineering, namely ontology evaluation and ontology evolution, is detailed. Discussing the overlapping and non-overlapping roles of these disciplines will help in appreciating and positioning the role and significance of ontology usage analysis within the realm of ontology engineering.

4.3.1 Positioning Ontology Usage Analysis in Ontology Engineering Lifecycle

Analysing ontology usage on the Web is different from assessing and evaluating the quality of an ontology. Most of the work by which ontologies are modified, accessed or assessed are considered auxiliary if not an integral component of ontology development methodologies. The reason is because the research community working on ontologies is historically rooted in the knowledge engineering community, therefore their emphasis has been more on envisaging a conceptual representation of the domain knowledge. Thus, most of the early work published under the rubric of ontology engineering focuses on the development (design-time) stage of ontologies (see Figure 1.4) and little emphasis is given on the in-use (run-time) stage of ontologies. However, ontology usage analysis is concerned about the in-use stage of the ontology in which ontologies are viewed as a digital engineering artefact and their adoption, update and utilization is assessed. In the following sub-section, the aspects in which Ontology Usage Analysis (OUA) are different from its adjacent areas, such as ontology evaluation, ontology maintenance and ontology evolution are discussed.

4.3.1.1 Ontology Usage Analysis (OUA) vs. Ontology Evaluation

Ontology Usage Analysis is different in many ways from Ontology Evaluation in spite of there being an overlap. To understand the difference, let us recall the definition of OUA proposed in Section 4.3 and then compare it with the definition of ontology evaluation in the context of an ontology development framework. OUA analyses the use of ontology on the Web in a real world setting by measuring its usefulness, usage and commercial advantages. Even though no formal definition for ontology evaluation is available in the literature, it is commonly referred to as a set of tools and methods to compare, validate and rank similar ontologies (Vrandevecic, 2010; Lozano-Tello and Gomez-Perez, 2004). Ontology evaluation and other ontology quality approaches (Tartir et al., 2005) are important, however their emphasis is more on guaranteeing that what is built will meet the requirements (ontology developers view) and that the final product (ontology artifact) is error free. Therefore, in some ontology engineering methodologies, ontology evaluation is a built-in process, while in others, it is considered as an independent component (Brank et al., 2005).

So, Ontology Evaluation focuses on the post-development phase of an ontology whereas OUA focuses on a post-implementation assessment scenario where actual utilization of a particular ontology in the Semantic Web context is observed and its adoption, co-use and reuse is analyzed after being instantiated. OUA focuses on the instantiated structured data on the Web, based on a domain ontology. For this reason, OUA can be viewed as a separate and independent activity from Ontology Evaluation. While, Ontology Evaluation helps in answering questions such as whether the built ontology matched the purpose, whereas Ontology Usage analysis provides the information needed to answer questions such as, *Given a lot of choices, which ontology should I use to describe the (domain-specific) information on the Web?*. Ontology usage can help in identifying the number of ontologies presently being used (adopted) by different publishers and their frequency of usage provides assistance in quantifying ontologies in term of their usage. Therefore, OUA is a post-implementation process and a part of ontology maintenance which can help in ontology evaluation as explained in Section 1.3. In Table 4.1, both OUA and Ontology Evaluation is compared against different factors to highlight the particular role and scope each have on ontologies.

4.3.1.2 Ontology Usage Analysis vs. Ontology Evolution

As mentioned earlier, the emphasis of OUA is to understand and measure ontology (vocabulary) usage in terms of its population, semantic relationship between different concepts, conformance with linked data principles and possible inferencing, depending

on the axioms of the ontology. On the other hand, Ontology Evolution, which is closely related to ontology change and versioning, covers the change management process to keep the ontology artifact up-to-date and increase its effectiveness and usefulness. Ontology Evolution is defined as the timely adoption of an ontology to the changes which have arisen and the consistent management of these changes (Haase and Stojanovic, 2005). The sources of change that trigger ontology evolution are explicit requirements or the result of some automatic change discovery method. A comparison between OUA and Ontology Evolution, considering different factors, are presented in Table 4.1. In this regard, while both OUA and Ontology Evolution focus on the run-time phase of an ontology lifecycle model, they differ in scope. The current approaches (Stojanovic et al., 2002) ignore an important source of information, that is, information about the actual utilization of an ontology on the Web. Actual utilization needs to be analysed, based on metrics and measurements, to qualitatively and quantitatively describe usage.

But even though they have different scopes, Ontology Evolution can benefit from OUA. For example, ontology usage analysis can provide the experiential evidence to gauge the anticipated impact of the proposed change in ontology. Recently, GoodRelations (Hepp, 2008) a well-known and adopted eCommerce ontology has gone through a few revisions¹ to evolve their conceptual model and implement changes in their model. Usage-based analysis provides the practical perspectives to the ontology evolution which are obtainable through ontology usage analysis and helps in maintaining logical consistency in an ontology.

In next section, the different phases involved in carrying out Ontology Usage Analysis are presented.

¹<http://www.heppnetz.de/ontologies/goodrelations/v1.htmlchangelog>; retr. 17/10/2012

Table 4.1: Drawing comparison between Ontology Usage Analysis, Ontology Evaluation and Ontology Evolution

	Ontology Usage Analysis	Ontology Evaluation	Ontology Evolution
<i>Scope</i>	Analyse how ontologies are being used	Evaluate how fit is an ontology to serve its purpose	Timely adaptation of an ontology to the arisen changes Haase and Stojanovic (2005) .
<i>Ontology lifecycle</i>	Run-time	Design-time	Design-time (part of maintenance process in ontology development methodology)
<i>Perspective covered</i>	Usage, Structural, Semanticity, Incentives	Functional, structural, logical consistency, annotation property usage.	Logical consistency, backward compatibility
<i>Provide answers to</i>	How to decide which ontology or term is suitable for reusability Zimmermann (2010) ?	How to measure the quality of an ontology for the Web Vrandevcic (2010) ?	How to evolve/update the conceptual model of ontology?

4.4 Solution overview: Methodological Approach for OUA

The aim of this thesis is to develop and implement an Ontology Usage Analysis framework known as **Ontology USage Analysis Framework (OUSAF)**. To develop such a framework, there is a need to create a process that is complete and contains the necessary detailed descriptions to communicate the methodological approach which needs to be followed for ontology usage analysis. To make the methodology more practical (that is, easy to follow in real world setting), it should provide fine grained descriptions of the steps involved, the methods or techniques applicable, assign roles to activities and present a clear idea about the input and outputs of the involved processes (Simperl, 2009). The developed methodological approach should provide the necessary detail to make it implementable; however, at the same time, it should be kept generic enough to allow the provision of different methods and techniques to be adopted for different application scenarios, when such a need arises. Keeping these requirements in view, I propose the OUSAF framework which has four broad phases namely; *identification*, *investigation*, *representation* and *utilization*. Each of these phases (see Figure 4.1) are discussed in the next sub- sections.

4.4.1 Identification Phase

Identification phase refers to the selection of the ontology(ies) that have to be analyzed. There are two common scenarios in which ontologies which need to be analysed are identified; (a) to determine the usage analysis of a specific domain ontology already known for the application area, for example FOAF for social networking; and (b) to investigate and identify the interesting ontologies available in the domain-specific dataset. These two scenarios require different types of solutions. The former type

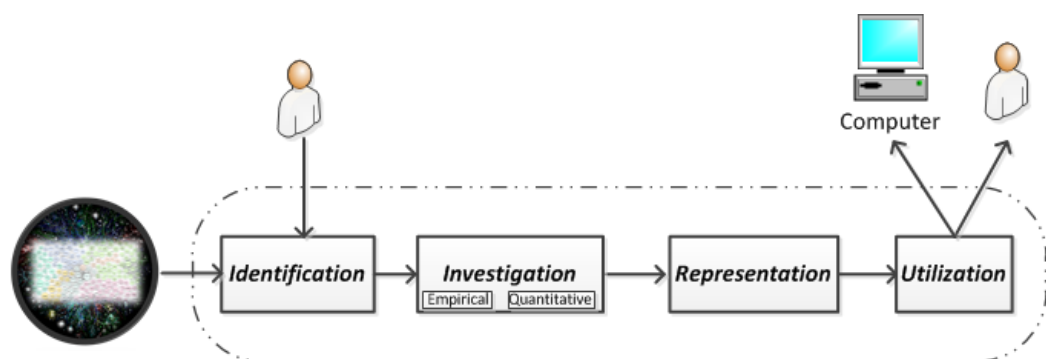


Figure 4.1: Phases in Ontology Usage Analysis

of scenario is trivial and can be addressed by looking up the available semantic search engines to directly access the namespace of the domain ontology. However, the latter case needs some exploration mechanism to identify the use of different ontologies in a dataset or those which are pertinent to domain-specific application areas. Therefore, before proceeding with the usage analysis, the candidate ontologies are identified from the corpus of the dataset. The aim of this phase is to develop the technique and approach to identify the potential candidate ontologies for usage analysis. Further discussion on this phase is given in Section 4.5.1.

4.4.2 Investigation Phase

The *Investigation* phase refers to the analysis of the use of a particular ontology. The aim of this step is to analyse the identified ontology(ies) to measure its usage and usage patterns. The usage analysis investigates how the conceptual model and the ontology components, such as Classes, Relationships, Attributes and Axioms are being used to annotate real world data. In order to obtain a comprehensive insight into the usage of a given ontology, the analysis is required to be performed at two levels. In the first level, empirical analysis needs to be performed in order to understand and highlight the key aspects contributing to the usage of ontologies. In the second level, based on the key aspects obtained through empirical analysis, ontologies need to be quantitatively analysed. The dimensions that represent the usage of ontologies considered in the investigation phase: are (a) usefulness; (b) usage; and (c) commercial advantage. The obtained results based on the analysis from these aspects are then combined to ascertain the usage analysis. From a methodological point of view, it is important to point out the key requirement of the framework is to support the different techniques and methods required to measure the statistical properties of ontology adoption. This requirement allows the adoption of feasible support methods, tools and techniques to improve the applicability and effectiveness of the usage analysis in a real world setting.

Further discussion on this phase and the metrics used for investigation is given in Section 4.5.2.

4.4.3 Representation Phase

The purpose of investigating ontology usage is to understand how an ontology is being used by different users and to exploit this information to utilize Semantic Web data effectively and efficiently. Therefore, analysis results obtained in the investigation phase have to be represented in a structured format to allow a larger

number of applications to use it for further information processing. This is done in the *representation* phase, in which analysis results are represented for further exploitation. Information processing, in this context, may include information retrieval, interlinking with other datasets, mash-ups and the automatic generation of prototypical queries. Additionally, for the optimal utilization of analysis in Semantic Web, the results need to be represented in a format which is equally accessible to both human and machine actors. Further discussion on the representation phase is given in Section 4.5.3.

4.4.4 Utilization Phase

Utilization phase refers to that phase in which the output of the usage analysis is further utilized to achieve conceivable benefits. Since there are different areas in which ontology usage analysis is helpful (as discussed in Section 1.4), the utilization phase covers the activities related to the exploitation of results, by different use case scenarios. To facilitate the utilization of the analysis in different application areas, the results are represented through a structured format developed in the representation phase, allowing the wider dissemination and exploitation of findings. To improve the usability of the methodology, I implement the use case which uses the ontology usage analysis information to assist applications in either accessing precise information from the Web or the assimilation of information to offer wider perspectives. Further discussion on this phase is given in Section 4.5.4.

In the next section, various steps that need to be achieved in each phase are discussed.

4.5 Ontology Usage Analysis Framework (OUSAF)

The main role of the OUSAF framework (depicted in Figure 4.2) is to empirically analyse the RDF data on the Web with a focus on domain vocabularies and ontologies. The framework supports conducting empirical analysis from two dimensions: one from an ontology perspective and the second from the RDF data perspective. From the ontology perspective, I consider the ontology as an engineering artifact (ontology document) to characterize the components defined in the document such as vocabulary, hierarchical and non-hierarchical structure, axioms and attributes. From the RDF data perspective, RDF triples are analysed to understand the patterns and the structure of the data available in the dataset. As mentioned in Section 4.4, the methodological approach followed for the analysis comprises of four different phases

namely: *identification*, *investigation*, *representation* and *utilization*. In Figure 4.2, which provides the schematic diagram, each of the steps is marked using a dotted rectangular box. The overview of the solution for each phase is discussed in the following subsections.

4.5.1 Identification Phase: Identification of candidate ontologies

As mentioned earlier, that there are two different ways by which the ontology usage analysis process can be initiated; first, the domain ontology which needs to be evaluated is known or given; and second, there is a need to identify the ontologies being used in the corpora/dataset.

4.5.1.1 Usage Analysis of a specific Domain Ontology

In a typical scenario, a user would like to analyse the domain ontology which conceptually represents the application area of interest. There are two possibilities: first, the user knows the specifics of the required domain ontologies such as the namespace of the ontology and the URI hosting the formal authoritative document of the ontology; and second, the user would like to search for a specific domain ontology. For the second case, the user can search for the domain-specific ontology using different services such as ontology search engines (Swoogle (Ding et al., 2004), Watson (d'Aquin and Motta, 2011)), ontology libraries (OntoSelect (Buitelaar et al., 2004), Cupboard (d'Aquin and Lewen, 2009), BioPortal (Noy et al., 2009)), Semantic Web search engines (Sindice (Tummarello et al., 2007), SWSE (Hogan et al., 2011)), and other applications built on Linked data corpora (FactForge (Bishop et al., 2011), Sig.ma (Tummarello et al., 2010)). Almost all of these services return the URI of the ontology to retrieve the authentic ontology source document.

4.5.1.2 Identify and analyse candidate ontologies from dataset:

It is also practically desirable to investigate the prevalence of different ontologies in a vertical application domain. This helps to know what different but related ontologies are being used to conceptualize the domain data. It is more advantageous from the view point of data publishers to know the availability of different ontologies being used by the community to describe the information on the Web. To address such common requirements, I propose the use of a domain-specific dataset to be used for the identification of ontologies and their usage in the dataset. The use of a domain-specific

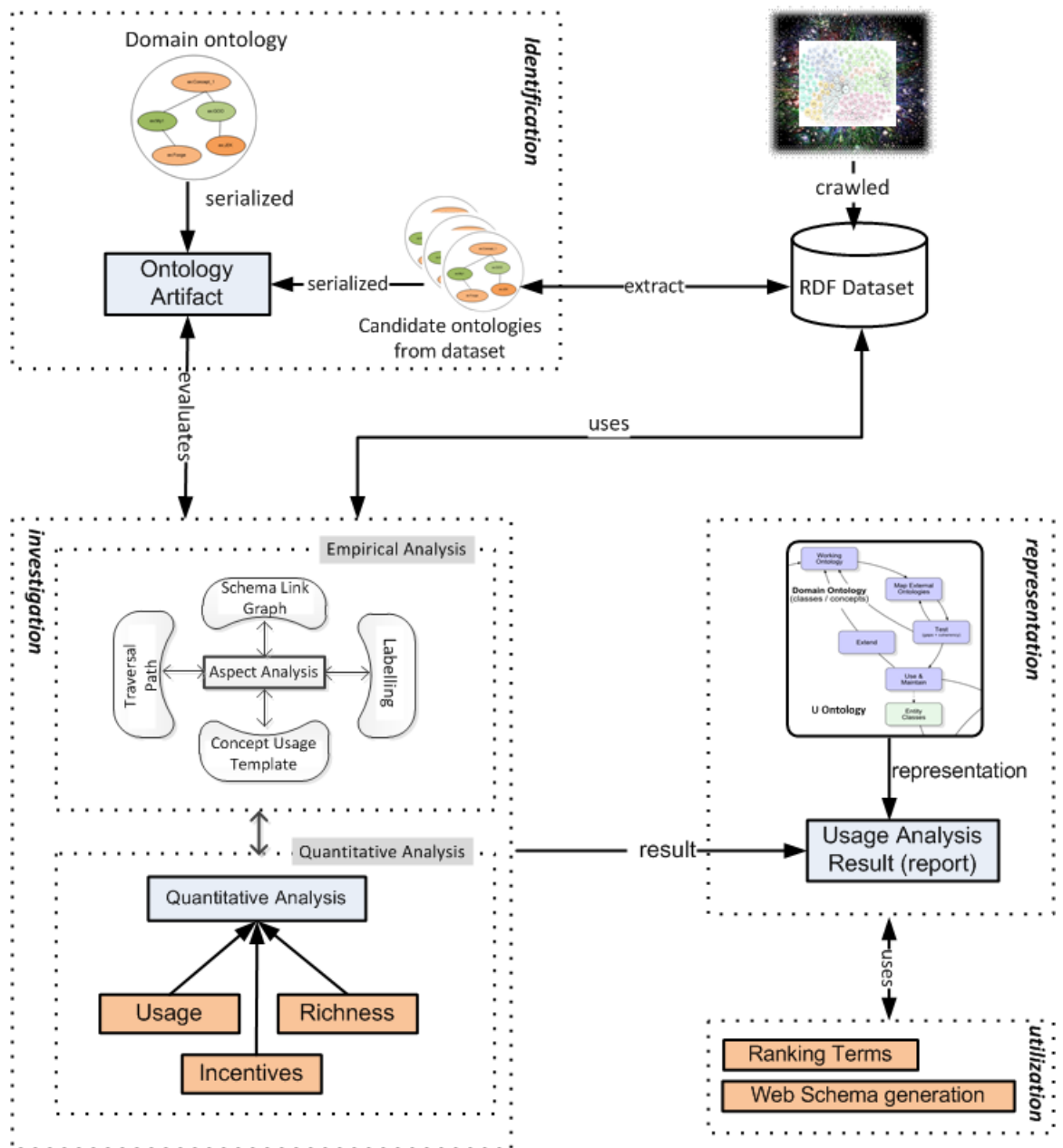


Figure 4.2: Ontology Usage Analysis Framework and its components with the process flow

dataset will not only help in identifying the ontologies, but will help to obtain the level of consensus that exists in the use of these ontologies. However, it is important to note that in such an approach, the considered dataset needs to be representative of the actual RDF data available on the Web and this aspect should be considered while interpreting the identification of ontologies and their usage in general.

To obtain such insight, **Ontology Usage Network (OUN)** is constructed to model the use of ontologies by different data publishers. The ontology usage network is based on the **Affiliation Network** model (Borgatti and Halgin, 2011) which captures the affiliation of agents to societies. In other words, it provides an intuitive way of representing participation data (membership) and studies the dual perspectives of the actors (ontologies, in our case) and the events (data publisher). Using Social Network Analysis (SNA) techniques, the Ontology Usage Network is processed and analysed to obtain the following insight:

- understand the use of different ontologies by different data publishers
- understand the ontology-to-ontology relationship present based on the co-participation of ontologies in different data sources

In order to obtain such insight, the Ontology Usage Network needs to be processed to generate the representational model capable of providing the required information. Undertaking such an analysis will assist in obtaining a better understanding of current ontology usage patterns and the similarities in usage among different data publishers.

In Chapter 5, the Ontology Usage Network Analysis Framework (OUN-AF) is presented to facilitate the identification of the ontologies from the dataset and perform the analysis to obtain the required insight. The framework implements different metrics and techniques and methods to structurally, typologically and semantically analyse the OUN and identify different ontologies and their co-participation behaviour. The methods and techniques followed for identification are dataset agnostic, thus making the approach applicable to different domain-specific datasets.

4.5.2 Investigation Phase: Investigating the Ontology Usage

To measure ontology usage, the dataset which comprises the semantic data collected from the web-of- data is considered and the instantiation and the use of properties of the conceptualized domain are measured, modelled by the domain ontology. As mentioned in Section 4.4.2, ontology usage is investigated at two levels: first empirically and then quantitatively. For these two levels of investigations, two

frameworks are developed as part of OUSAF. These two frameworks are **EMPIRICAL Usage Analysis Framework (EMP-AF)** and **QUANTITATIVE Usage Analysis Framework (QUA-AF)**. Their brief introduction is as follows:

4.5.2.1 Empirical Analysis

The use of different ontologies is empirically analysed to understand the key aspects involved in ontology usage. The EMP-AF framework implements a set of metrics to measure the use of different ontology components from different aspects to establish a better insight into the prevailing usage patterns on the Web. The different aspects considered are the use of pivot concepts, their semantic description, the use of textual description and knowledge and data patterns. The metrics developed for these aspects are as follows:

- **Schema Link Graph:** Schema Link Graph (SLG) unveils the relationships that are present between different vocabularies at instance level to semantically describe the entities being represented by the ontology concepts.
- **Concept Usage Template:** Concept Usage Template (CUT) captures the instantiation of concepts, the relationship it has, and the use of different data properties to provide factual information.
- **Labeling:** Labeling captures the use of different properties for labeling purposes. Labeling properties help in providing a textual description of the entities useful for human interpretation and user interface
- **Traversal path:** Traversal path captures the data and knowledge patterns prevalent in the dataset.

4.5.2.2 Quantitative Analysis

Based on the insight obtained through the empirical analysis, in quantitative analysis, ontologies are analysed from different dimensions in order to obtain a comprehensive insight into their usage. The QUA-AF framework implements metrics that are grouped into three categories to measure the usage of an ontology, encompassing its **usefulness**, **usage** and **commercial** advantages. Usefulness measures the richness of the ontology components and provides structural insight into how a given ontology is modelled and how the semantics are represented. While on one hand, the inclusion of such information helps in identifying the semantically rich components of an ontology, on the other hand, it also assists in drawing a comparison, if any, between the

usage and semantically rich components. Usage mainly captures usage patterns in terms of the presence of different ontological terms in describing the instance data to provide semantic metadata on the Web. Commercial advantage captures the incentive model available to the early ontology adopters and Semantic Web data publishers. It considers all the components of the ontologies such as classes, relationships, taxonomical relationships and axiomatic triples to quantify usage trends on the Web.

- **Measuring Ontology Richness:** In this category, the richness of the ontology components such as concepts, object properties (relationship) and data properties (attributes) are quantified. In the case of RDFS vocabularies, since object and data properties are not disjoint, only object properties are considered to refer to the predicates defined by the vocabulary.
- **Measuring Ontology Usage:** To analyse and quantify the use of ontologies on the Web, metrics are defined to measure the use of ontologies and their components which includes the use of different concepts, relationships, attributes and axioms.
- **Measuring Incentive:** In this category, I consider the key factors fostering the growth and adoption of vocabularies/ontologies and consider them as the driving factors for early adopters. Two of the other driving factors to consider in this research are the incentives available to structured data publishers and the support available for an ontology/vocabulary in Semantic Web applications and tools.

These sets of metrics provide a more practical view of the use of ontologies since they cover the technical aspects of the ontology (usefulness), adoption and uptake of the ontology (usage) and the incentives available for ontology users (commercial advantages), thereby covering all aspects which, if considered, can help in identifying compelling products [Simmons \(2005\)](#) which, in our case, are ontologies.

- **Combining the Analysis Results:** The quantified measures of the abovementioned aspects are then combined to obtain an overall picture of the usage of a given ontology. Combining these values further helps in ranking them to obtain the required set of ontology components, based on the user requirements.

The metrics developed for the investigation phase are discussed in Chapters 6 and 7. The developed metrics cover the different aspects of usage to obtain detailed insight

into the required quantitative measures which are useful for the exploitation of the results. Similar to identification, the methods and techniques used for investigation are ontology and dataset agnostic, therefore making the solution application to different ontologies and datasets.

4.5.3 Representation Phase: Representing Usage Analysis

The representation phase of the usage analysis methodology concerns the representation of results in such a way that it can be easily disseminated and accessed by other applications. To capture the analysis results, Ontology Usage Ontology (U Ontology) is developed. U Ontology is a meta-ontology which provides the conceptual architecture to represent the usage patterns of the domain ontology in a dataset. The usage patterns contain both the knowledge and data patterns which assist in understanding the knowledge available in the dataset and generate prototypical queries to access data. In other words, U Ontology provides machine processable information which can be used to improve the accessibility of Semantic Web data and the reuse of ontologies. The usage analysis of a particular domain ontology obtained using OUSAF is encoded using U Ontology which provides a different set of concept and relationships between concepts to allow the user to access the analysis findings programmatically. The availability of information on how ontologies are being used and what are the prominent knowledge and data patterns helps in effectively accessing the required information over the Web. Additionally, such metadata provides the meta-level information about ontologies including their usage to support application/tool development and providing pragmatic feedback to ontology evolution and change management.

Following the Semantic Web community recommendations of reusing existing ontologies wherever possible, the U Ontology is considered as an extension to the OMV (Ontology Metadata Vocabulary). OMV (Hartmann et al., 2005) attempts to provide a standard ontology metadata for describing ontologies and their entities. The metadata vocabulary for describing ontologies is modelled as an ontology and is called OMV Core² with the provision of supporting different extensions (Palma et al., 2008). The U Ontology is considered one of its extensions, implementing usage analysis of ontologies to further enrich existing application-specific ontology-related information such as mapping, ontology evaluation, and ontology changes.

In Chapter 8, the conceptual model developed to provide the representation for usage analysis is presented. A formal conceptualization model, based on RDF and

²<http://ontoware.org/projects/omv>

OWL that allows the standardized formulation of ontology usage analysis results, is adopted.

4.5.4 Utilization Phase: Utilizing Usage Analysis results

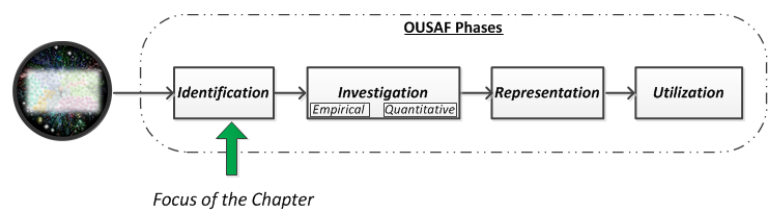
The *utilization* phase makes use of the analysis results. As mentioned in Section 1.4, usage analysis can be used by different groups of users (ontology developers, data publishers and data consumers) to access information over the Web by generating prototypical queries based on the schema-level data available in the dataset. The U Ontology which is populated during the representation phase contains usage-related information to be accessed by users to know about the usage of an ontology to retrieve information over the Web, based on usage.

The populated U Ontology provides descriptive and quantitative details on the use of different ontology components which is then useful for realizing the benefits of ontology usage analysis. Two use cases namely: (a) construction of prototypical queries; and (b) construction of Web Schema are explored in the utilization phase to demonstrate the utility of usage analysis. In the case of prototypical query generation, U Ontology provides the list of ontology components with their usage to assist in querying explicit and implicit information. Similarly, in the case of Web Schema, for a given vertical application area, highly used vocabularies are accessed to understand the structure of the domain-specific entities. Further explanation on how results from the usage analysis are obtained are mentioned in Chapters 6 and 7 and represented in Chapter 8. The formal model with instance data (ontology usage analysis results) is then used to present the ontology usage catalogue, encapsulating the usage status of a given ontology, which is discussed in Chapter 8.

4.6 Conclusion

In this chapter, proposed solution overview is presented. OUSAF Framework is discussed with its components. In order to signify and highlight the importance of ontology usage analysis, the relationship of usage analysis with other relevant overlapping areas such as Ontology Evaluation and Ontology Evolution is explored and discussed. Details on the methodological approach are presented to specify the phases involved in carrying out the empirical analysis which are: *identification*, *evaluation*, *representation* and *utilization*. In the next chapter, the model used to construct the Ontology Usage Network is discussed, as well as Social Network Analysis techniques and methods used for the identification phase of the OUSAF.

Chapter 5 - Identification Phase : Ontology Usage Network Analysis Framework (OUN-AF)



5.1 Introduction

As mentioned in the previous chapter, analysing ontology usage comprises of four phases: namely *identification*, *investigation*, *representation* and *utilization*. The identification phase, which is the focus of this chapter, is responsible for identifying different ontologies that are being used in a particular application area or in a given dataset for further analysis. As previously discussed, ontologies whose usage is to be analysed, fall into two categories:

- the domain ontology to be analyzed for usage is known
- the domain ontology to be analyzed for a particular domain needs to be identified according to application-specific requirements.

The first case is trivial as the user can access the ontology from its respective namespace URI, however, in the latter case, a mechanism is required to identify the presence of different ontologies in the required domain and to select the potential ontologies based on the users specific requirements and selection criteria. A few of

the common requirements which form the selection criteria for the identification of ontologies in this scenario are:

1. What are the widely used ontologies in the given application?
2. What ontologies are more interlinked with other ontologies to describe domain-specific entities?
3. What ontologies are used more frequently and what is their usage percentage based on the given dataset?
4. Which ontology clusters form cohesive groups?

To analyse such a set of selection criteria, to identify different ontologies, their links with other ontologies, and to identify the usage patterns prevalent in an application-specific area, detailed insight into which different data sources (data publishers) use particular ontologies is required. In order to establish a better understanding of ontology usage and to identify the ontologies, based on the abovementioned criteria, this chapter proposes the Ontology Usage Network Analysis Framework (OUN-AF) that models the use of ontologies by different data sources using an Ontology Usage Network (OUN). OUN represents the relationships between ontologies and data sources based on the actual usage data available on the Web in the form of a graph-based relationship structure. This structure is then analysed using metrics to study the structural, typological and functional characteristics of OUN by applying Social Network Analysis (SNA) ([Knoke et al., 2008](#)) techniques.

This chapter is organized as follows. Section 5.2 introduces Social Network Analysis (SNA) and the different types of relationships often represented in SNA. Section 5.3 provides the rationale for using SNA to obtain the required analysis for the ontology identification phase. It also provides an overview of the literature in which SNA has been used in the context of ontologies. In order to provide the background and introduce terms relevant to SNA, Section 5.4 presents the key terms relating to SNA and the different types of networks and properties observed in these networks. Furthermore, the necessary background knowledge on SNA is also discussed in this chapter, however, by no means should it be considered the complete background knowledge on the subject matter, therefore readers are referred to [Newman \(2010, 2003\)](#); [Wasserman and Faust \(1994\)](#) to obtain a more complete coverage on Social Network Analysis. In Section 5.5, the Affiliation Network, relevant concepts and its graphical representation are detailed. In Section 5.6, OUN-AF is proposed for the ontology identification phase of OUSAF. OUN-AF phases with a set of activities and their sequence are presented. In Section 5.7, the metrics developed to analyse the

relationship between ontologies and the data source are given. Section 5.8 gives an overview of the analysis by applying the metrics on OUN and the projected networks. In Section 5.9, the evaluation of the ontology identification phases based on OUN-AF is presented. Finally, Section 5.10 concludes the chapter.

5.2 Social Network Analysis

Social Network Analysis (SNA) is a methodical approach toward mapping and measuring the relationships between people, organizations, computers, and information resources. Historically, it belongs to the social sciences in which social relations among a set of actors were studied. However, in the past few years, the idea of networks has been extended to include other unifying themes to study social interaction in living species, digitally connected devices and natural world connections. As a result of this change, research in SNA is witnessing a substantial shift in its focus from a small network to a large scale networks that are large in size and complex in structure. In general, SNA studies the social relationships among a set of actors and these relationships take different forms, depending on the type of network under study. More importantly, SNA provides the methods to characterize the structure of social networks, the important positions in the network, the strength of relationships between different sets of nodes and the existence of sub-networks (Erétéo, 2011). In other words, SNA allows us to measure the relationship, communication, and information flow between nodes through edges and focuses on uncovering the patterns of actors' interactions in the network. Therefore, network analysis is based on the intuitive notion that these patterns are important features of the activities of the individuals who display them through their interaction (Freeman, 2003).

Social networks are made up of actors that are linked by social *relationships*. Thus, actors and relational ties (links) are the basic elements of the network. There is a wide range of social relationships which can take place between actors of the network. The interlinking between nodes denotes the flow of information reflecting their social relationships. These different types of relationships that exist have been studied in the literature (Garton et al., 1997; Erétéo, 2011) and can be grouped into the following three main categories :

- Explicit relationships
 - Interaction
 - Affiliation
-

In the next subsection, each relationship category is briefly described.

5.2.1 Explicit Relationships

The first category of explicit relationships represents the relationships between people, organizations or between people and organizations. For example, between people, the explicit relationship could be brother, sister, parent, friend, etc. One of the earliest research studies which examined explicit relationships was conducted by [Zachary \(1977\)](#). In this study, a network was formed in a university-based karate club to understand the cause of internal conflicts within the club (see Figure 5.1). The explicit relationships between cohesive groups were identified as being due to social links and eventually, the club was split into two groups to mitigate the issues within the club. In other work, [Hogg et al. \(2008\)](#) presented an empirical study of an online political forum where users engage in content creation, voting, and discussion which forms the explicit connection in the network.

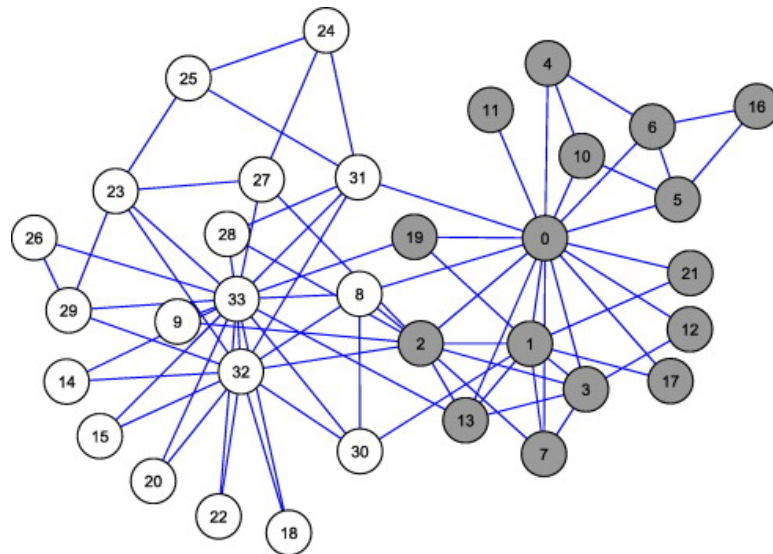


Figure 5.1: Zachary club network. The links between two nodes represents explicit relationship and nodes are split into two groups one in round white and other in round gray.

5.2.2 Interaction

The second category of relationships in a network represents the flow of information or communication between actors such as the relationship between authors, and the interaction of player belonging to one team. In the digital world, this takes the form of participation and collaboration in discussion forums and co-authoring articles in wikis. [Brass \(1985\)](#) investigated the interaction patterns of men and women in an

organization and the relationships between these patterns to study the perceptions which influence promotions to supervisory positions in the organisation. Interaction networks have been studied in biomedicine to understand the interaction between different chemical components. One such interaction-based network is presented by [Tong et al. \(2002\)](#), who analyses the protein-to-protein interaction networks (see Figure 5.2) derived from phage-display and two-hybrid analysis.

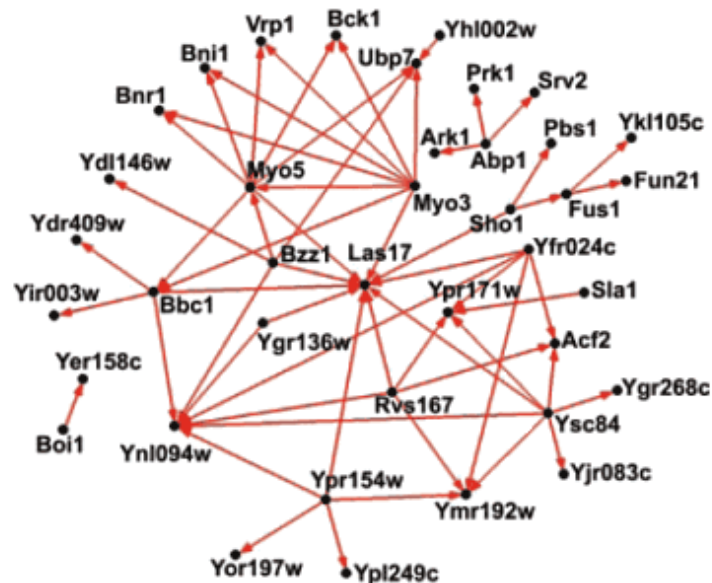


Figure 5.2: Protein to Protein interaction network.

5.2.3 Affiliation

The third type of relationship in networks is affiliations which essentially correspond to the similarity between actors in the network. Similarity emerges from the fact that actors share the same attributes which then enables the affiliation between actors. The network which represents affiliation-type relationships is termed the ‘affiliation network’. Affiliation networks have been used extensively in economic science and social networks to study the affiliations between different but related elements, based on their affiliations. For example, in economic science, affiliation networks are used to analyse how organizations and their members interact to understand the economical mechanism ([Mariolis, 1975](#)). [Wasserman and Faust \(1994\)](#) presented an affiliation network (see Figure 5.3) which was produced from the 1998 GIS dataset, consisting of 10 US computer and software firms and 54 strategic alliances between them. Affiliations networks are further discussed in Section 5.5 due to their relevance to the model adopted for identifying the ontologies.

So, using any of these relationships in a network, the underlying application can be represented as a network-based format with relationships between different elements,

thereby, allowing analysis to be carried out on it.

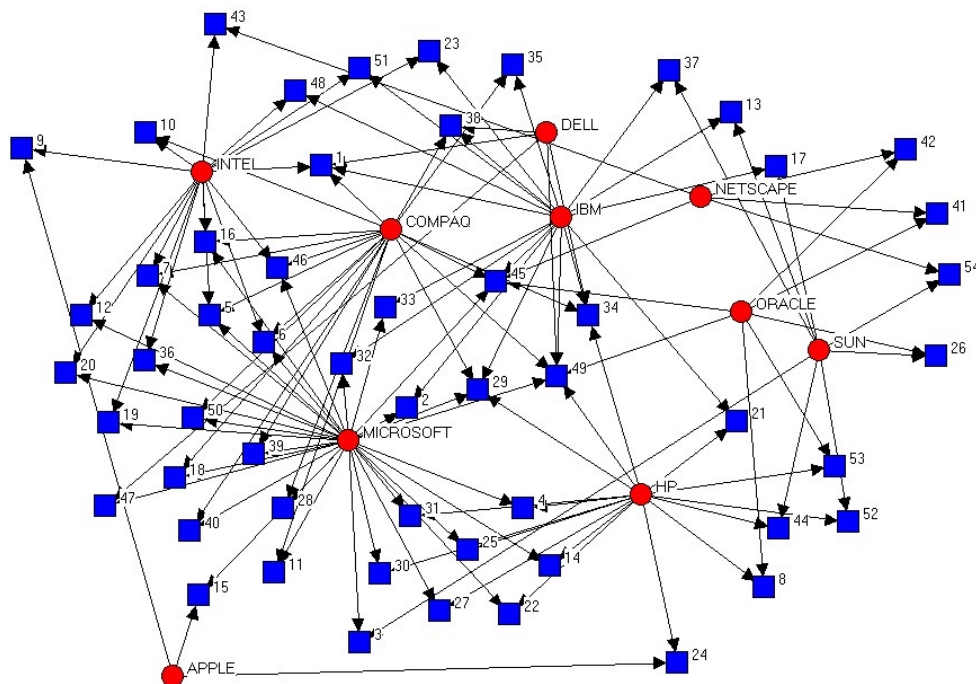


Figure 5.3: Affiliation Network (two-mode graph) consisting of 10 US computer software firms (red circles) and 54 strategic alliances (blue squares)

In the next section, the rationale for using SNA and its techniques to identify ontologies from a given dataset in the ontology identification phase will be discussed.

5.3 Rationale of using SNA in Ontology Identification

As mentioned in Section 5.1, the aim of the ontology identification stage is to identify the potential ontologies on which a detailed usage analysis are need to be performed. There may be various criteria against which the ontology from a given dataset has to be selected. Some of these are:

1. Select an ontology in a given domain which is used by the highest number of data publishers
2. What other ontologies are being co-used by the same data publishers?
3. Given the other co-used ontologies, which ontology has the central position among them? Such analysis would be beneficial to data publishers to understand

which ontologies have a central role in facilitating the use (or even adoption) of other ontologies.

4. Are there any common ontology usage patterns among data sources (data publishers) dominating the dataset?

These selection criteria are applied on a dataset that is highly complex in nature, large in volume and highly interconnected.

For example, Figure 3.2 shows the latest version of the Linked Open Data (LOD) cloud currently published on the Web. From the figure, it can be seen that various relationships exist between the different datasets. It is important to note that the LOD cloud diagram (Figure 3.2) is a high level depiction of the interlinking and encapsulates the underlying mechanisms used to create relationships between different datasets. Ontologies that are present at a lower level (not depicted in the figure) facilitate the semantic representation of the information and to have interlinking of entities distributed across different datasets. So undoubtedly, the dataset comprising data from different sources, annotated using domain ontologies, is a complex network structure. Thereby, to study such a complex network structure according to the identified selection criteria requires a representational model capable of representing multi- or bi-dimensional information components providing the base model to address the abovementioned queries.

Furthermore, specific to this case, *an **ontology** can be used by any number of users and a **user** can use any number of ontologies at the same time.* This introduces specific requirements that need to be considered while modeling the information.

Thus, the specific requirements are:

1. First there are two types of entities involved, namely ‘ontologies’ and ‘users’; and second both of these entities can have a number of connections with other types of entities. For example, a user (e.g. data publisher) can use a number of ontologies and similarly, an ontology can be utilized by several users. Therefore, the framework required to represent the inter-relation between ontologies and users in the given dataset should be able to capture and represent multi-edge systems to allow an edge span over several nodes, contrary to a normal graph where an edge connects only two nodes.
 2. These two sets of entities are disjoint as edges flow between these two sets of nodes rather than within their own set. This means an ontology cannot have a relationship (direct connection) with another ontology or a user (data publisher) cannot connect to another user in the network. (Note: Ontologies can import
-

other ontologies to reuse the term, however, here the scope of the term ‘use’ refers to the use of ontologies by the data publisher for semantic annotation.)

Keeping in mind the abovementioned complex network to be analyzed in a dataset and the specific features according to how they should be represented, a framework is needed that is able to represent the information and relationships (of distinct set of entities) within a given dataset in a format that can be analyzed further. The framework should be able to identify ontologies and their use by different data sources, thereby the flow between nodes (of different types) needs to be studied to unleash the co-membership relationships which are otherwise not visible. One possible way to represent such information in the required format is **Social Network Analysis (SNA)**. SNA provides the lens through which one studies the complex (social) networks and their components to mine the hidden relationships present in their structure. The use of ontologies by different data sources resembles several social networks in which similarities between actors are frequently a source of interaction.

In this thesis, Social Network Analysis (SNA) techniques are used to study the complex networks representing the information pertaining to the use of ontologies by different data sources.

In the next subsection, the literature in which SNA is applied to the Semantic Web in general and ontologies specifically is discussed.

5.3.1 Related work on the use of SNA in Semantic Web

One of the earliest works in which RDF data is analyzed and SNA is employed was by [Ding et al. \(2005\)](#). In this work, the authors analyzed the social and structural relationships available on the Semantic Web, focusing mainly on FOAF and DC vocabularies. The study was performed on approximately 1.5 million FOAF documents to analyze instance data available on the Web and their usefulness in understanding social structures and networks. Additionally, the use of different namespaces, concepts and properties is discussed in order to provide a perspective on different FOAF implementations. They identified the graphical patterns emerging in social networks and represented this using the FOAF vocabulary and the degree distribution of the network. As this research provides a detailed analysis of Semantic Web data by focusing on a specific vocabulary, it does not provide a framework or methodology to make it applicable to different vocabularies.

In ([Guéret et al., 2010](#)), the authors used SNA to understand the structural properties of the constructed network using the Billion Triple Challenge (BTC) 2010 corpus. Centrality measures such as betweenness and the degree distribution is used

to identify the hub nodes in the network capable of being failure points if those nodes become attached or go out of service. Their focus was on infrastructure and SNA and measured its robustness in improving the state of semantic Web data. They reported that 80% of all triples within the Linked Data cloud point either to URIs in the same namespace, blank nodes, or literals.

In (Mika, 2005), the authors studied the semantic relationships among tags, based on their co-occurrences by extending the traditional bipartite graph to a tri-partite graph. Using network analysis techniques such as a clustering coefficient and betweenness centrality, they analyzed the relationship between different classes of nodes, namely actors, concepts and instances. Since the focus of their study was to understand the emergence of community-based semantics to formalize the conceptual knowledge, it does not offer an understanding of the explicit semantics available on the Web.

Other work in the literature has analyzed a different number of ontologies using SNA. For example, Zhang (2008) covered five ontologies, Theoharis et al. (2008) analyzed 250 ontologies and Cheng and Qu (2008) used approximately 3,000 vocabularies. In all this work, ontologies were investigated to measure their structural properties, the distribution of different measures and terminological knowledge encoded in ontologies, but none includes how they are being used on the Web. This thesis attempts to incorporate ontology usage in the study model, to learn how ontologies are actually being used and the cohesive groups that are available.

In the literature, SNA has been applied to measuring the structural aspect of ontologies. In such studies, where ontologies are essentially labelled and directed graphs, different techniques are used to understand the structural properties, such as the number of edges to and from a node and the shortest distance between two nodes. However, the use of SNA and its techniques to analyse the use of ontologies and measure the relationships based on usage has only been applied marginally. Mika (2005) applied SNA to analyse the implicit semantics emerging from the use of tags, but apart from this, SNA has been mainly applied on single ontologies. However, in the identification phase of the OUSAF framework, the ontologies and their use by different data sources are represented in a way that allows the "affiliation" between ontologies and different data publishers to be measured.

In next section, the terms and concepts related to the representation and analysis of Social Network Analysis are introduced prior to a discussion on the different types of networks.

5.4 Key concepts of SNA relevant to Ontology Identification phase

Networks, which are also known as “graphs” in mathematical literature, represent the complex systems of interconnected components. In its simplest form, a network is comprised of *nodes* and *edges*. SNA provides different mathematical techniques to quantitatively analyse the network and understand the relationship patterns available in the network. In this chapter, different SNA methods and techniques are used to identify ontology usage patterns. The terms used in the discussion are described in the following subsection.

5.4.1 Key Terms and their definitions

Key terms used in this chapter and this thesis relevant to Social Network Analysis are as follows:

Network : a distinct set of actors and the connections between them

Graph : a distinct set of nodes and a set of edges
Node : basic unit of a network (or graph) which represents actors. It is also referred to as vertex in the literature.

Edges : a connection between two nodes. In social networks, edges are known as links representing relationships between two actors.

Hyperedge : an edge that connects more than two nodes

Weighted edge : an edge with an assigned value representing the importance of the edge

Labelled edge : an edge with a label attached to it to provide a description of the relationship

Hypergraph : a graph in which generalized edges (called hyperedges) may connect more than two nodes

Multigraph : The term multigraph refers to a graph in which multiple edges between nodes are permitted ([Harary, 1991](#))

Weighted graph : a graph in which each branch is given a numerical weight. A weighted graph is therefore a special type of labelled graph in which the labels are numbers (which are usually taken to be positive) ([Weisstein and Polynomials, 2004](#)).

Labelled graph : a graph with each node is labelled differently (but arbitrarily), so that all nodes are considered distinct for purposes of enumeration ([Weisstein and Polynomials, 2004](#)).

One-mode networks : one mode (1-mode) networks involve relations among a single set of similar actors.

Two-mode networks: two mode (2-mode) networks involve relations among two different set of nodes

Distance : the distance between two actors in a network (or nodes in a graph) is calculated by summing the number of distinct ties (lines) that exist along the shortest route between them.

Path : a list of nodes of a graph, each linked to the next by an edge. Formally it is defined as: a path on a graph, also called a trail, is a sequence $\{x_1, x_2, \dots, x_n\}$ such that $(x_1, x_2), (x_2, x_3), \dots, (x_{n-1}, x_n)$ are graph edges of the graph and the x_i are distinct ([Weisstein and Polynomials, 2004](#))

Direct Path : a sequence of directed edges from a source node to an end node. Formally, it is described as a sequence of vertices, v_1, v_2, \dots, v_n , in a directed graph such that there is an edge from v_i to v_{i+1} for $i = 1, 2, \dots, n - 1$.

Geodesic or short path : shortest sequence of edges between two given nodes.

Degree : the degree k_v of a node v measures the immediate adjacency of the node in the network and is computed as the number of edges incident on a given node (i.e v).

$$k_i = \sum_{v=1}^n a_{ij}, \quad 0 < k_i < n \quad (5.1)$$

where a_{ij} is the entry of the i th row and the j th column of the adjacency matrix A

k-core : The k-core of graph is a maximal subgraph in which each node has at least degree k. The core-ness of a node is k if it belongs to the k-core but not to the (k+1)-core

Density : density ρ , in general measures the connectedness in a network. Therefore, a high ρ value indicates a dense network and a low value indicates a sparse network. It is defined as:

$$\rho(G) = \frac{m(G)}{m_{max(G)}}, \quad 0 < \rho < 1 \quad (5.2)$$

where m is the number of edges in the network and $m_{max(G)}$ denotes the number of possible edges, which is $\frac{n(n-1)}{2}$ for the undirected network and $n(n-1)$ for directed ones.

Centrality : a general measure of the position of an actor within the overall structure of the social network. Centrality measures help in answering the question, “who is the most important or central actor (node) in the given network” (Newman, 2008). There are several metrics to measure centrality, the most widely used being degree centrality, betweenness centrality and closeness centrality.

Power-Law Degree distribution : The degree of a node in a network is the number of edges incident to it. Therefore, a degree distribution $p(k)$ is the probability distribution of the degrees of the nodes over the whole network. Thus, $p(k)$ represents the probability that a random node has degree k , and is defined by the fraction of the nodes in the network that have degree k (Oliveira and Gama, 2012). Real-world networks are quite different from random networks in terms of degree distribution. Random networks often show binomial degree distribution (Newman et al., 2001) because of the equal probability of an edge being present or absent in the network. However, in real-world networks (Barabási et al., 2000) discovered that the distribution of node degree is very heterogeneous and highly right skewed. This means that a large number of nodes have low degree and a small number of nodes have high degree. In the literature, this is also known as long tail as shown in Figure 5.4.

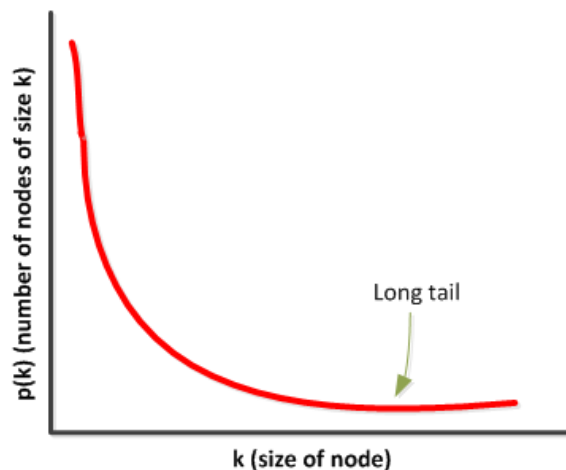


Figure 5.4: Power-law distribution

5.4.2 Types of Networks

A network, in its simplest form, is a set of nodes with edges between them. In the literature, nodes are also referred to as vertices and edges as ties and relationship links. An example of a network is shown in Figure 5.5 which contains eight nodes and eight edges. Networks are primarily composed of nodes joined by edges. There are other ways in which networks are more complex in structure and topology. Both nodes and edges can take a variety of properties which make these networks more complex than the one shown in Figure 5.5. An edge can have a direction (which could be uni- or bi-directional), label, weight and attributes, and likewise, nodes can have type, weight and attributes as well. Other types of networks with some variations are discussed as follows.

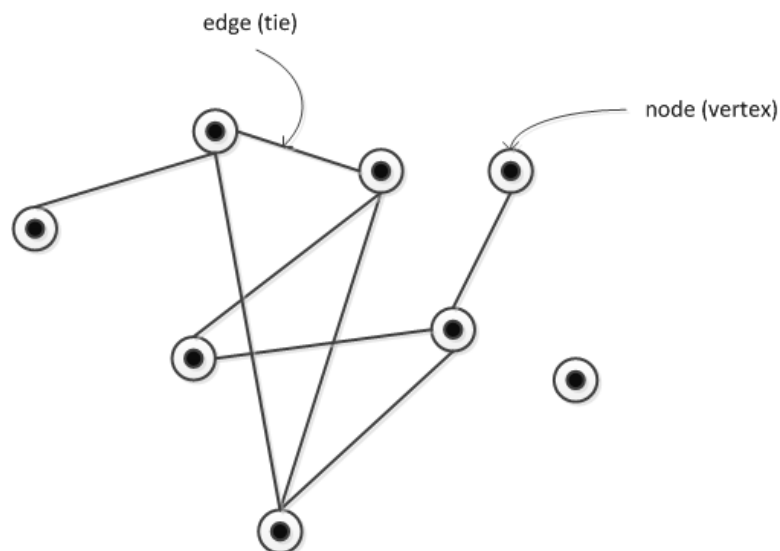


Figure 5.5: A simple example of a network (with eight nodes and edges)

5.4.2.1 Labelled and Directed Networks

A slightly more complex network is presented in Figure ???. Figure ??? (a) represents a graph with labelled nodes (i.e. A, B, C, D, E, F) and the edges are unidirected to show which two nodes are directly connected. In the context of social networks, the nodes can represent anything, such as a man, woman, boy, girl or thing such as a city, country; likewise, edges can show the kinship, friendship, professional affiliation, distance or other thing representing the relationship (tie) between nodes. In a network, either one or both nodes and edges can carry weights which makes the network a weighted graph.

5.4.2.2 Labelled, weighted and bi-directional network

The additional attribute of weight can be added to the network. A weight can be attached to a node or edge or to both. Weight attributes can represent any quantifiable measurement necessary for the interpretation of the information represented in the network. For example, Figure 5.6(b) represents a labelled, directed graph in which edges carry weight to represent the distance between cities.

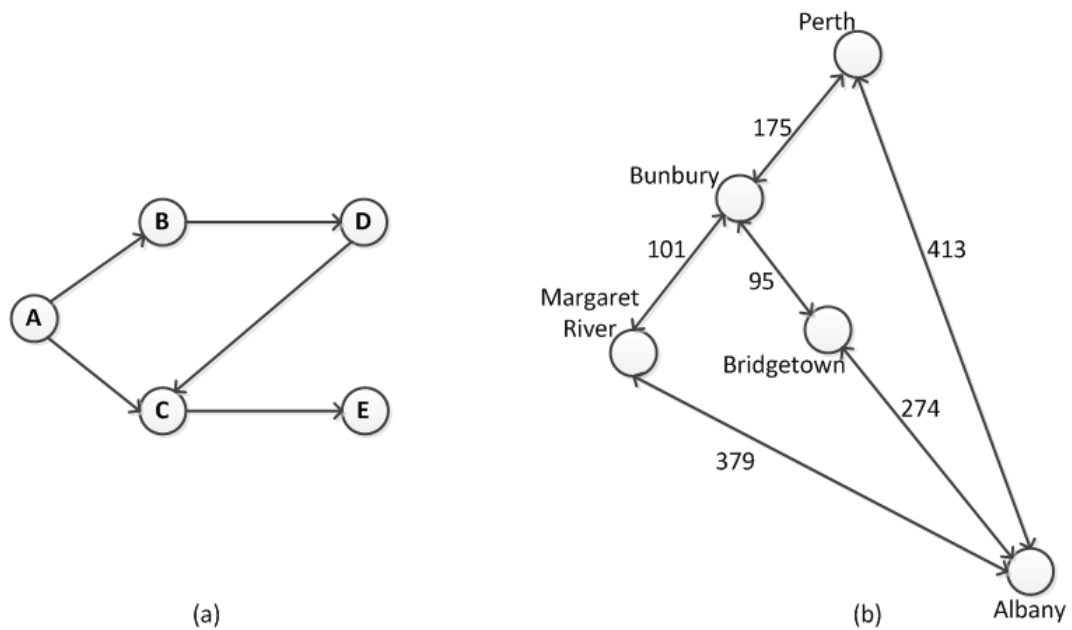


Figure 5.6: Examples of different types of networks: (a) network with labelled nodes and directed edges; (b) network with labelled nodes, weighted and bidirectional edges.

5.4.2.3 Hypergraphs

One special type of graph is called a hypergraph, in which edges join more than two nodes together. In the abovementioned graphs (Figure 5.5 and Figure 5.6), the edges connect two adjacent nodes whereas in a hypergraph, an edge (also known as a hyperedge) is incident to an unspecified number of nodes. Hyperedges are often used in social networks to indicate family ties. For example, all the individuals belonging to one family in a graph can be joined through a hyperedge which joins all the nodes representing individuals belonging to a family. In Figure 5.7, a hypergraph with three hyperedges namely 1, 2 and 3 is shown. Hyperedge 1 joins nodes B, C and D, hyperedge 2 joins E, F and G, and hyperedge 3 joins A, B and C. Hypergraphs are more expressive than regular graphs which often fall short in providing complex relational object representation (Zhou et al., 2006). Real world complex problems requiring clustering and classification of objects based on their attributes are best

represented as a hypergraph because of their expressivity and to avoid information loss.

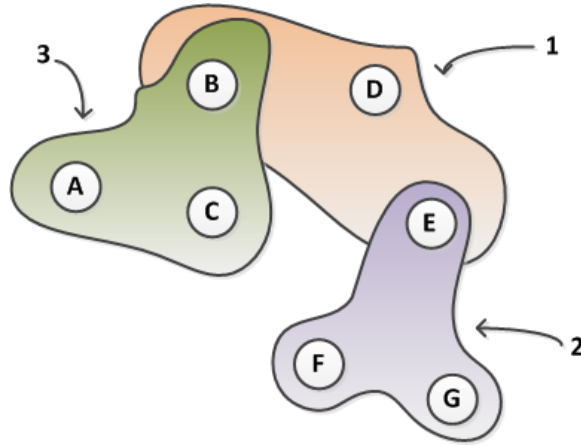


Figure 5.7: Hypergraph with three hyperedges

5.4.2.4 Bipartite (2-mode) Graph

While hypergraphs help in clustering and partitioning nodes based on their attributes, there are other special graphs which are naturally partitioned in various ways. Such graphs are called **bipartite graphs** and contain nodes (vertices) of two distinct types, with edges only between nodes of a distinct type (See Figure 5.8). **An affiliation network** is an examples of a bipartite graph in which actors and events are two types of entities (two sets of distinct type) related by the affiliation of the former in the latter ([Lattanzi and Sivakumar, 2009](#)).

The affiliation networks, their structure, key concepts, representational model and operations such as ‘projection’ are presented in the next section.

5.5 Affiliation Network

A detailed description of affiliation networks is given in this section. A wide range of social networks have been built by analysing the similarities between actors which is the source of interaction between them. This interaction, based on the similarity notion, is termed “*affiliation*” and a network representing such similarity-based relationships is called an affiliation network. The term “affiliation” is reserved for cases when the data reflects some kind of participation or membership ([Borgatti and Halgin, 2011](#)) in the social network. Affiliation networks are different in many ways from other types of networks as follows ([Wasserman and Faust, 1994](#)):

- Affiliation networks are essentially two mode (2-mode) networks

- Affiliation networks consist of subsets of actors, rather than a simple pair of actors
- The connection among members of the first mode (first set of nodes) is based on the relationship established through the second mode (second set of nodes)
- Affiliation networks provide the mechanism to study dual perspectives of the actors and the events

In an affiliation network, the importance of individual relationships within its society is studied which helps in exploring individuals' behaviour and their acceptance by the society. The modelling of relationships using affiliation helps in identifying the joint participation in social event and provides the opportunity to individuals to develop pairwise relationships with other individuals based on their participation. In the context of ontology usage, an affiliation network is used to model the use of an ontology by different data publishers. Thereby, ontologies are the actors and the data publishers form the hypothetical society (i.e. event) in which these actors participate. Before formally presenting the model and metrics to measure ontology usage, the concepts relevant to affiliation networks are discussed. Affiliation networks cannot be analysed by merely looking at the pairs of the actors or events because it is subsets of actors who participate in the events. However, it is often desirable to understand the patterns of relationships within one of the sets. This means, to know how two nodes of the same set are related to each other, based on their relationships to the other set of nodes. These kinds of relationships, which are inferred based on the relationships nodes hold with other sets of nodes is called **co-affiliation** (Borgatti and Halgin, 2011). Examples of co-affiliations are 'attendance at the same event', 'membership in the same club', and 'members of the same corporate board' in which co-affiliation among nodes of the same sets are inferred.

In order to obtain the co-affiliation in the network, the affiliation network is transformed from a two-mode network into a one-mode network (network with only one type of node). This procedure of transformation is called **projection** (Borgatti, 2009). Projection in affiliation networks is done by selecting one of the sets of nodes and linking two nodes from that set if they were connected to the same node of the other set. This means that projection allows the analysis of the network from one of two perspectives: the actor's view or the event's view (Tutoky, 2011). The procedure of projection is also referred to as the duality of the two-mode network since it allows dual perspectives (one from each mode) of the affiliation network. From the actors view, two actors are connected if they have participated in at least one event together and from the events view, two events are connected if at least one of the actors has participated in these two events.

5.5.1 Representing Affiliation Network

One of the best ways to represent networks is through a matrix. A matrix is a rectangular table in which rows and columns represent the nodes of the network and the value in the cell (where the column and row intersects) represents an edge. Similarly, affiliation networks are represented through an affiliation matrix, $A = \{a_{ij}\}$. Matrix A is a two-mode sociomatrix in which rows represent actors and columns represent events. Generally, affiliation network A is defined as: A is a bipartite graph $A = (U, V, E)$ where U (often known as actors) and V (often known as events) are disjoint set of nodes and $E \cup (UXV)$ is the set of edges. With $p = |V|$ and $q = |U|$, A is represented by an incident matrix with p lines and q columns. Formally, $A = \{a_{ij}\}$ records the affiliation of each actor with each event in an affiliation matrix such that:

$$a_{ij} = \begin{cases} 1, & \text{if actor } i \text{ is affiliated with event } j \\ 0, & \text{otherwise} \end{cases}$$

The value of 1 is put in the (i,j)th cell if i th actor (i th row) is affiliated with j th event (j th column) and an entry of 0 if i th actor is not affiliated with j th event. Table 5.1 represents the affiliation matrix of a sample author-paper affiliation network. This is a bipartite graph with two types of nodes, namely authors and papers. The edge (link) in the network shows which authors have written which papers and two authors linked to same paper represent a co-authorship relationship as depicted in Figure 5.8. For example, in Figure 5.8, author 1 has written two papers, namely 1 and 3; paper 3 has only one author, namely author 1, however for paper 1, author 2 is a co-author with author 1.

Table 5.1: Affiliation matrix of author-paper affiliation network

	paper 1	paper 2	paper 3	paper 4
Author 1	1	0	1	0
Author 2	1	1	0	1
Author 3	0	1	0	1

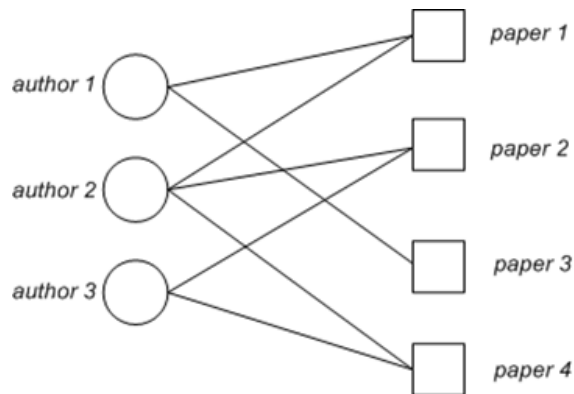


Figure 5.8: An example of an author-paper affiliation network

5.5.2 Projecting Affiliation Network

As mentioned above, it is possible to analyse a two-mode network in its original form, however there are few methods that exist for this purpose. Often, two-mode networks are transformed into a one-mode network by a procedure called projection. Projection generates a one-mode network by selecting nodes of one set (for example, authors in Figure 5.8) and linking two nodes from the set if both are connected to the same node of the other set. The projection of a two-mode network into two 1-mode networks provides the opportunity to analyse the affiliation network using methods developed for traditional unipartite or social networks. The transformed one-mode (co-affiliation) network helps to understand and analyze the ties among the members of a node set (Borgatti and Halgin, 2011). For example, Figure 5.9(a) presents the authors co-affiliation network and Figure 5.9(b) presents papers co-authorship network.

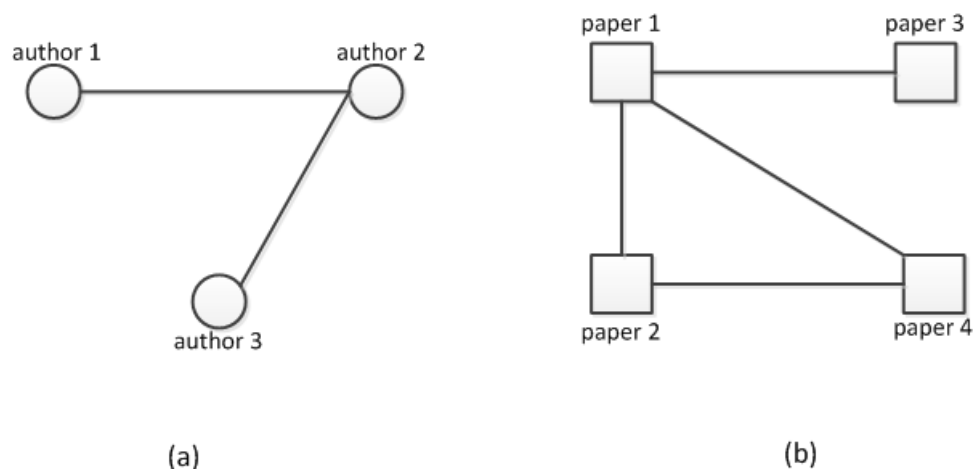


Figure 5.9: Example of projection: (a) authors co-affiliation network, (b) paper's co-authorship network

In the next section, the Ontology Usage Network Analysis Framework (OUN-AF) which is proposed for the study and analysis of the relationships between different ontologies and data sources in a dataset using SNA techniques is presented.

5.6 Ontology Usage Network Analysis Framework (OUN-AF)

As mentioned earlier, the objective of the ontology identification phase is to identify the use of different ontologies by different data publishers in a given application area to discover hidden usage patterns. Therefore, in order to mine such analysis, the Ontology Usage Network Analysis Framework (OUN-AF) is proposed, as shown in Figure 5.10. OUN-AF provides the implementation of the ontology identification phase as part of a methodological approach developed for ontology usage analysis. OUN-AF comprises three phases, namely *Input* phase, *Computation* phase and *Analysis* phase, as shown in Figure 5.10. The role of the *Input* phase is to collect and maintain the dataset containing the crawled data comprising real world Semantic Web data. The *Computation* phase provides the computational architecture to transform the input into a format that facilitates ontology identification-related activities. The *Analysis* phase analyses the computational model by using the developed metrics and interprets their results. Each phase is discussed in detail in the following subsections.

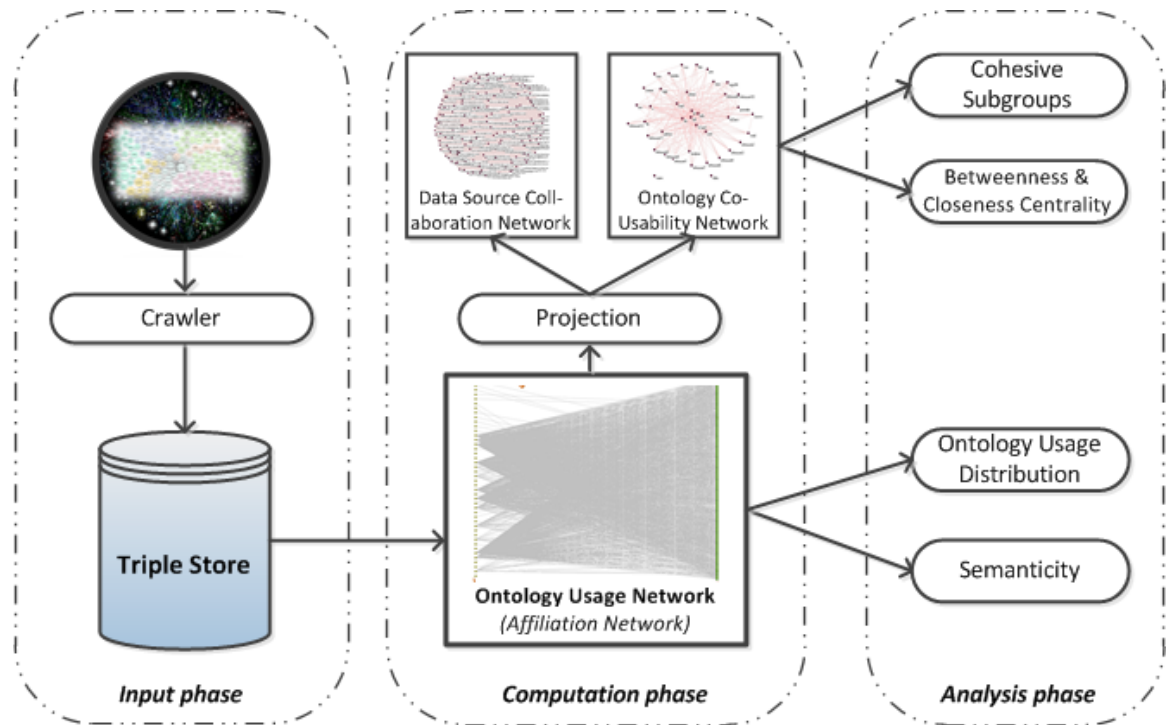


Figure 5.10: Ontology Usage Network Analysis Framework (OUN-AF) and its phases.

5.6.1 Input phase

As mentioned earlier, the input phase is responsible for managing the dataset which is then used for subsequent operations. The two key components in this phase are a crawler and an RDF triple store. The crawler crawls the Web to collect the required data to form a dataset that is then stored in the RDF triple store. In order to point the crawler to relevant and interesting data sources, the bootstrapping process first builds the seed URIs as multiple starting points. A list of seed URIs is obtained by accessing semantic search engines which return the URIs (URLs) of the data sources (web sites) with structured data. The crawler collects the Semantic Web data (RDF data) and after preprocessing it, loads it to the RDF triple store (database).

Preprocessing contains the transformation routines to convert the crawled data into the required format and append the necessary metadata such as provenance detail, timestamp, and data source details. From a data management point of view, since RDF data comprises triples (statements) which do not provide a default mechanism to group or associate certain sets of triples to a context, the Named Graph (Carroll et al., 2005) approach is used. The Named Graph approach enables the provision of contextualization by introducing an additional URI (context) to a set of related triples.

The dataset is then accessed by the components of the computation phase to query the information. SPARQL end point is used to pose SPARQL queries, which then access the triple store to evaluate the query and return the result set. The result set, in this case, is the set of RDF triples in the form of an RDF graph.

5.6.2 Computation phase

The computation phase provides the computational architecture to transform the data maintained in the RDF triple store to a model so that further analysis can be performed on the given dataset and ontologies and their usage patterns can be identified. The computational architecture comprises a model to represent the ontologies and their relationship with data publishers and a network operation in a network-based structure to analyse the ontologies, their interrelationship with other ontologies and relationship with data publishers. The OUN-AF transforms the data into two formats. The first format is a two-mode affiliation network and the second format is a one-mode network which is generated from a two-mode network using the projection procedure. The two-mode affiliation network (i.e. *Ontology Usage Network*) and the subsequent one-mode networks (i.e. *Ontology Co-Usability* and *Data-Source Collaboration* networks) are discussed in the following sub-section.

5.6.2.1 Ontology Usage Network (OUN)

OUN is an affiliation network represented as a bipartite graph providing the model to allow the creation of a relationship between two distinct sets of nodes and the analysis of the use of ontologies by different data publishers. OUN comprises *ontology* and *data-source* sets of nodes with an edge between the ontology node and data-source node if the ontology has been used by the data source. Here, data source refers to any domain name on the Web which has published RDF data by using Web ontologies. To formally define the Ontology Usage Network, first the two sets of nodes, namely *ontology set* and *data-source set* are defined and then the OUN definition is presented.

An ontology set is defined as the set O which represents the nodes of the first mode of the affiliation network. An ontology set O contains the list of ontologies used on the Web-of-Data such that there is a triple $t < s, p, o >$ anywhere in the dataset (specifically, otherwise in general, on the Web-of-Data) where $o \in O$ is the URI of either p or o .

A data-source set is defined as the set D which has the list of hostnames on

the Web-of-Data such that there exist a triple $t < s, p, o >$ in the dataset (specifically, otherwise in general, on the Web-of-Data), where s is the hostname (domain names in URL parlance) and either p or o is a member of O .

The Ontology Usage Network (OUN) is a bipartite graph, denoted as $OUN(O, D)$ that represents the affiliation network, with a set of ontologies O on one side and a set of data- sources D on the other and edge (o, d) represents the fact that o is “used” by d . A snapshot of OUN is shown in Figure 5.11

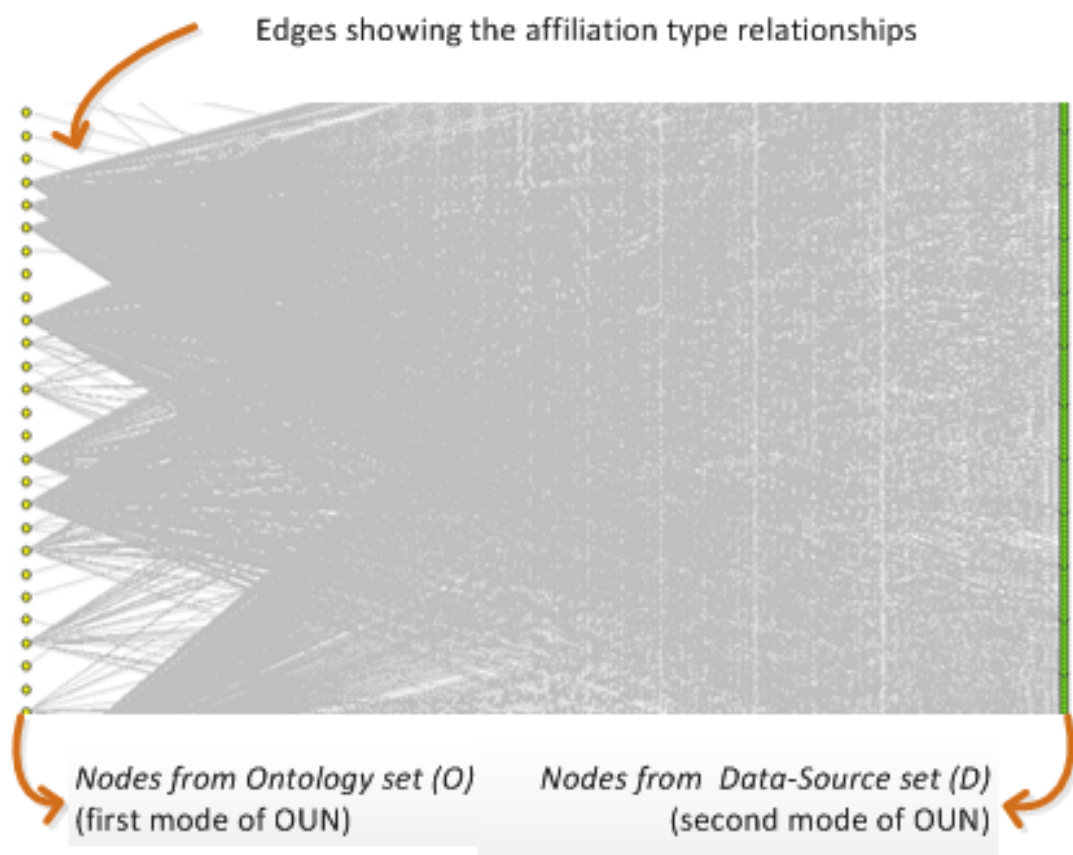


Figure 5.11: Ontology Usage Network (affiliation network with one set of nodes representing ontologies and other set of nodes representing data sources).

There are certain types of analyses which cannot be obtained directly through OUN, particularly if the requirement is to infer the connectedness present within one set of nodes based on their co-participation in the other set of nodes. For such kinds of analyses, it is necessary to study one set of nodes, hence the information represented by OUN has to be transformed from a two-mode network to a one-mode network. This transformation is achieved by using the process of projection, as discussed in the next section.

5.6.2.2 Projection

OUN is a two-mode network which enables the bipartism found in the network to be studied, however sometime it is desirable to obtain one set of nodes and study their co-membership in the network. This is achieved through the process of transforming a two-mode network into a one-mode network by using the technique called projection. In the case of OUN, projection is used to generate two one-mode graphs; one for nodes in the ontology set known as the *Ontology Co-Usability* network (See Figure 5.18), and second for the data-source set known as the *Data-Source Collaboration* network (See Figure 5.17) as shown in Figure 5.10.

The Ontology Co-Usability network is an ontology-to-ontology network, in which two nodes are connected if both of the ontologies are being used by the same data source. This means that the *Ontology Co-Usability* network represents the connectedness of an ontology with other ontologies, based on their co-membership in the data source.

The Data-Source Collaboration network is a data-source-to-data-source network in which two nodes are connected if both of them have used the same ontology to describe their data. The Data-Source Collaboration network represents the similarity of data-sources in terms of their need to semantically describe the information on the Web.

5.6.3 Analysis phase

The analysis phase is the third and last phase of OUN-AF. The objective of this phase is to mine the hidden relationships explicitly or implicitly present in the two-mode network (i.e. OUN) and one-mode networks (Ontology Co-Usability and Data-Source Collaboration). To objectively analyse the networks, the SNA techniques which are used and the metrics which are developed are explained in next two subsections.

5.6.3.1 Analysing (Two-mode) OUN

In order to analyse the OUN, different metrics are proposed. The quantitative analysis on the OUN affiliation network provides the infrastructure to measure the degree of nodes in each set of modes. In the case of OUN, the degree of the nodes representing ontologies and degree of the nodes representing data sources is obtainable. Additionally, the degree distribution, which is the probability distribution

of (node) degrees over the whole network, can be obtained to compare the network and their connections with other types of networks. In order to obtain the degree and degree distribution of the OUN, two metrics are defined to obtain these measures. The metrics are:

- *Ontology Usage Distribution*: this measures the degree of ontologies and their distribution over the network.
- *Semanticity*: this measures the degree of the data sources and their distribution over the network.

These two metrics are formally described in Sections 5.7.1 & 5.7.2, respectively.

5.6.3.2 Analysing Projected One-mode Network

As mentioned above, the projection procedure is applied which transforms the OUN into two projected one-mode networks. The resultant projected network contains nodes from their own set (e.g. ontologies) and the relationship between nodes shows their co-affiliation in the original two-mode network. These networks provide the ground to discover other interesting properties which helps further in understanding how ontologies are placed in terms of their usage and co-usage by different data sources and their typological position within the network. The two obtained networks are the Ontology Co-Usability network which is essentially an ontology-to-ontology network and the Data-Source Collaboration network which is a data-source-to-data-source network.

As the focus of the analysis phase of OUN-AF (and of OUSAF as well) is to analyze ontology usage, only the ontology-to-ontology network is considered. The ontology-to-ontology network represents the relationships between different ontologies available in a dataset. SNA provides the techniques and methods to study the strategic position of nodes in the overall network. To obtain such insight and understand the position and the groups of nodes with similar positions (which form the cluster), the following metrics are used:

- *Betweenness and Closeness Centrality*: these measures identify the nodes which have important (strategic) positions in the network such as betweenness and closeness
 - *Cohesive Subgroups*: identifies the group of nodes which share some similarities particular in terms of their relationship and position
-

These two metrics are formally described in Sections 5.7.3 & 5.7.4, respectively. It is important to note that the abovementioned metrics can be used to analyse the Data-Source Collaboration network.

In the next subsection, the set of sequential activities carried out in the OUN-AF to analyse the network is presented.

5.6.4 Sequence of OUN-AF activities

OUN-AF comprises three phases and in each phase, a different set of activities is involved. In order to provide an overview of the activities and their sequence, a summary of key activities is presented.

- In the Input phase, the dataset is built. In order to build the dataset comprising the information regarding the use of ontologies by different data publishers, data sources are identified to crawl the data.
 - In the computation phase, the crawled data is processed to extract the relevant information to build the node sets of OUN. The two node sets are *ontology set* and *data-source*.
 - To study the two-mode network, using these two set of nodes, OUN is constructed. The affiliation relationship between these two sets of nodes is established, based on the usage related data represented in the dataset.
 - Different metrics are developed to perform the required analysis. The metrics are:
 - Ontology Usage Distribution (OUD)
 - * First, the degree of each node that is a member of the *ontology set* is measured.
 - * Second, the degree distribution of the *ontology set* is measured to understand the distribution of degrees in the node set, and to understand the distribution of connections in the network. Power-law distribution is observed in the network.
 - Semanticity
 - * First, the degree of each node that is a member of the *data-source set* is measured.
-

- * Second, the degree distribution of the *data-source set* is measured to understand the distribution of degree in the node set.
- To study the one-mode network, the OUN network is transformed into two one-mode networks using the projection operation to produce the *Ontology Co-Usability* and *Data-Source Collaboration* networks .
- To measure the position and identify the key nodes in the network, the following two metrics are used on projected one-mode networks (i.e. *Ontology Co-Usability*)
 - Betweenness and closeness centrality measures identify the nodes in key strategic positions.
 - Cohesive subgroups help in identifying the clusters present in the network, based on functional or structural similarities.
- Interpret the results and identify the ontologies based on the required selection criteria

The abovementioned set of activities are depicted in Figure 5.12, based on workflow notations highlighting the key activities and their corresponding outputs. The output generated as the results of the activity or process is shown using a grey dotted line, whereas the set of activities follows normal workflow representation.

In the next section, the metrics which are used to study the OUN and the projected one-mode network as part of OUN-AF are defined.

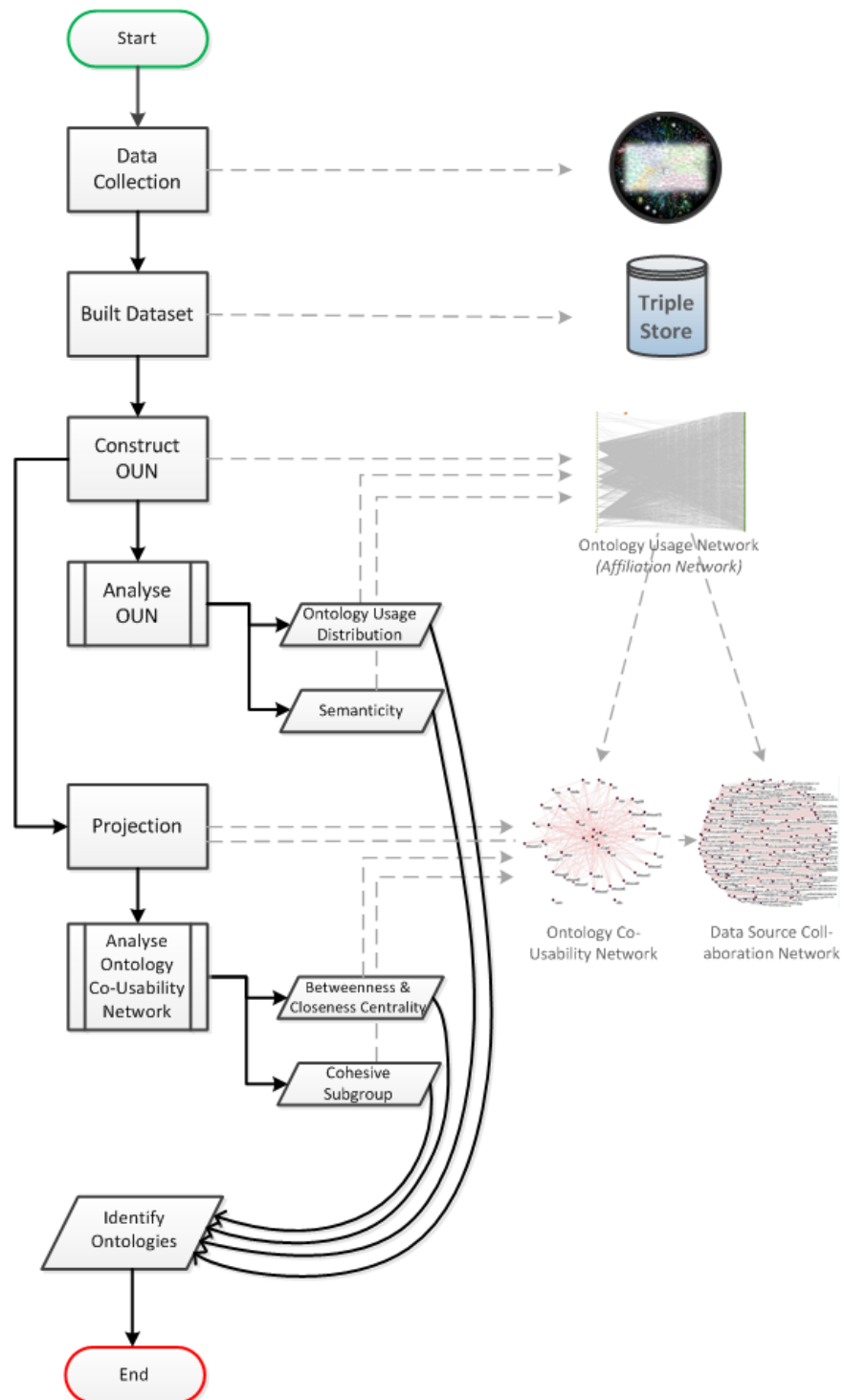


Figure 5.12: Flow of activities in OUN-AF.

5.7 Metrics for Ontology Identification

The metrics used for the identification phase are presented in the following subsection.

5.7.1 Ontology Usage Distribution (OUD)

The metric Ontology Usage Distribution, OUD_k , is used to identify which fraction of the ontologies in the network have a degree k . OUD measures the degree distribution of ontologies in an affiliation network. Recall that in an ordinary graph, the degree of a node in a network is the number of edges connected to that node. However, in the case of an affiliation network, the degree of a node is the number of ties it has with the number of nodes of the other set. The degree centrality $C_D(o_i)$ of an ontology o_i is measured as:

$$C_D(o_i) = d(o_i) = \sum_{j=1}^{n_1} A_{ij} \quad (5.4)$$

Where $i = 1, \dots, n_1$, $n_1 = |O|$, $d(o_i)$ is the degree of i_{th} ontology o , and A is the affiliation matrix representing OUN.

The normalization (which aligns the probability distribution to an adjusted value) of degree in an affiliation network is obtained by dividing the total number of nodes in the other set rather than dividing by the number of nodes in the same set. Therefore, the normalized ontology usage degree is measured as:

$$C'_D(o_i) = \frac{d(o_i)}{\text{number_of_datasources} - 1} \quad (5.5)$$

Where $\text{number_of_datasources} = |D|$ represents the total number of nodes in the other set of nodes (i.e 2nd mode of affiliation network).

5.7.2 Semanticity

Semanticity distribution $Semanticity_k$ identifies the fraction of the data sources in the network which have a degree k .

Similarly to ontology usage distribution which measures the distribution of ontologies among data sources, semanticity measures the participation of different ontologies in a given data source. In other words, semanticity measures the richness of a given data source in terms of the use of ontologies. The more ontologies are being

used by a data source, the higher semanticity value it has. Semanticity is measured by calculating the degree centrality and degree distribution on the second set of nodes in an affiliation network, which is the set representing the data sources present in the dataset. The degree centrality $C_D(ds_i)$ of a data source ds_i is measured as:

$$C_D(ds_i) = d(ds_i) = \sum_{j=1}^{n_2} A_{ij} \quad (5.6)$$

Where $i = 1, \dots, n_2$, $n_2 = |D|$, $d(ds_i)$ is the degree of i_{th} data source ds , and A is the affiliation matrix representing OUN

The normalization of Semanticity is measured as :

$$C'_D(ds_i) = \frac{d(ds_i)}{\text{number_of_ontologies} - 1} \quad (5.7)$$

where $\text{number_of_ontologies} = |O|$ represents the total number of nodes in the other set of nodes (i.e. second mode of an affiliation network).

5.7.3 Betweenness and Closeness centrality

Betweenness and Closeness centralities is used to identify the important (or key) nodes in the network. Like Ontology Co-Usability, both these centrality measures are computed on the projected one-mode network.

Betweenness Centrality is the number of shortest paths between any two nodes that passes through the given node. The betweenness centrality $C_B(v_i)$ of a node v_i is measured as :

$$C_B(v_i) = \sum_{v_s \neq v_i \neq v_t \in V} \frac{\sigma_{st}(v_i)}{\sigma_{st}} \quad (5.8)$$

where σ_{st} is the number of shortest paths between vertices v_s and v_t .

And $\sigma_{st}(v_i)$ is the number of shortest paths between v_s and v_t that pass through v_i .

Closeness centrality is a measure of the overall position of a node (actor) in the network, giving an idea about how long it takes to reach other nodes in the network from a given starting node. Closeness measures reachability, that is, how fast a given node (actor) can reach everyone in the network (Oliveira and Gama, 2012) and is

defined as:

$$c_v = \frac{n-1}{\sum_{u \in V} d(v, u)} \quad (5.9)$$

where $d(v, u)$ denotes the length of a shortest-path between v and u . The closeness centrality of v is the inverse of the average (shortest path) distance from v to any other vertex in the graph. The higher the c_v , the shorter the average distance from v to other vertices, and v is more important by this measure.

5.7.4 Cohesive Subgroups

Generally speaking, cohesive subgroups refer to the areas of the network in which actors are more closely related to each other than actors outside the group. In the most extreme case of a cohesive subgroup, each member of the group is expected to have strong connections with every other member of the group. However, this condition is very strict and normally it is relaxed by introducing the notion of cliques or n-clique. In an n-clique, it is not required that each member of the clique has a direct tie with the others, but instead that it has to be no more than distance n from each other.

Formally, a clique is the maximum number of actors who have all possible ties present between them.

In the next section, the analysis and results obtained using these metrics to identify the ontologies in a real world data set are discussed.

5.8 Analysis of Ontology Usage Network

To base the findings on empirical grounds, a dataset comprising real world instance data is built to populate the OUN and, using the metrics described in the earlier section, the analysis is performed to understand the relationships between ontologies and data publishers and the inter-relationships between the ontologies based on their co-usage by different data sources.

5.8.1 Dataset and its characteristics

A dataset comprising real world structured data which is annotated using ontologies is developed for the *identification* phase. In order to build a dataset which has a fair representation of the Semantic Web data described using domain ontologies, semantic search engines such as Sindice ([Tummarello et al., 2007](#)) and Swoogle ([Ding et al.,](#)

2004) are used to build the seed URLs. These seed URLs are then used to crawl the structured data published on the Web using ontologies. The dataset built for the identification phase comprises 22.3 millions of triples, collected from 211 data sources¹. In this dataset, 44 namespaces are used to describe entities semantically. The resulting Ontology Usage Network is depicted in Figure 5.13 and comprises 1390 edges linking 44 ontologies to 211 data sources. The complete list of ontologies and their prefixes used in the dataset collected by crawling the Web is shown in Figure 5.14.

In terms of generic OUN properties, the *density* of the network is 0.149 (Eq. (5.2)) and the *average degree* is 10.90 (Eq. (5.1)). The average degree shows that the network is neither too sparse nor too dense which is a common pattern in information networks. Details on the other properties and metrics are given in the following subsections.

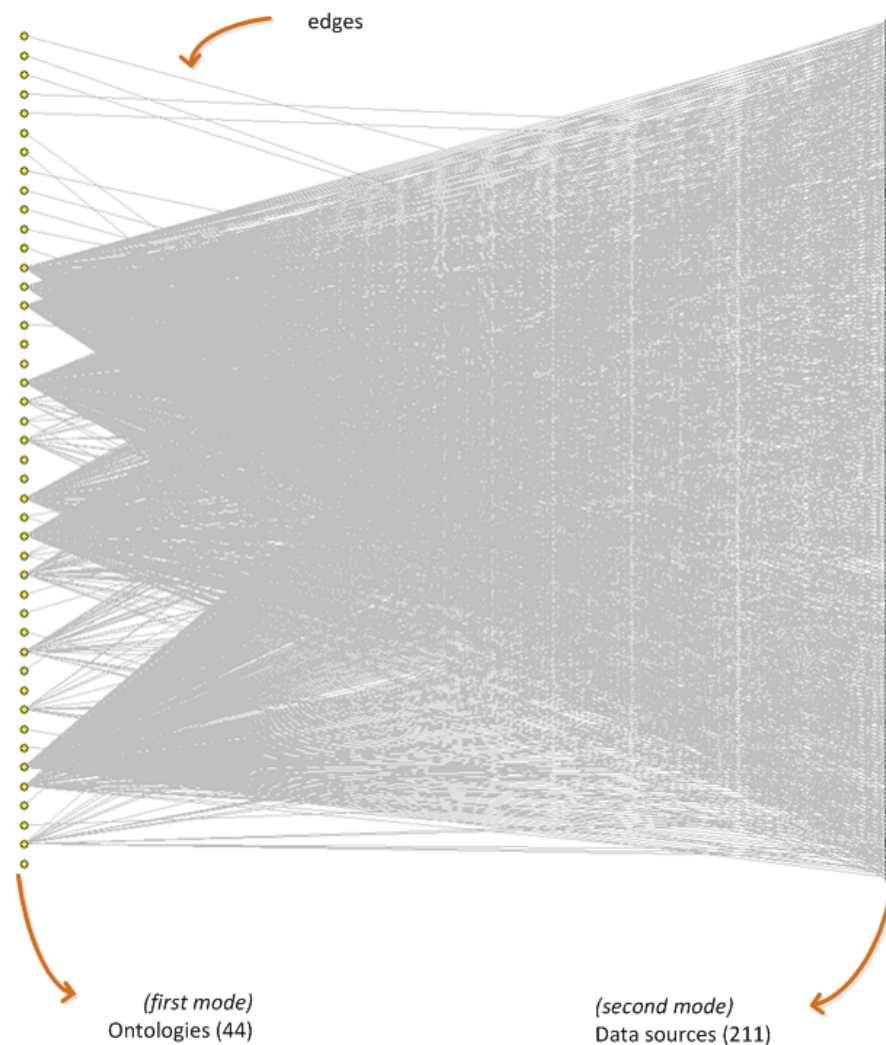


Figure 5.13: Ontology Usage Affiliation Network (bipartite graph).

¹<https://docs.google.com/spreadsheet/ccc?key=0AqjAK1TTtaSZdGpIMkVQUTRNenlrTGctR2J1bkl6WEE>

#	Prefix	Namespace	Degree
1	rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#	208
2	gr	http://purl.org/goodrelations/v1#	208
3	rdfs	http://www.w3.org/2000/01/rdf-schema#	190
4	vCard	http://www.w3.org/2001/vcard-rdf3.0# & http://www.w3.org/2006/vcard/ns#	164
5	owl	http://www.w3.org/2002/07/owl#	141
6	foaf	http://xmlns.com/foaf/0.1/#	115
7	xhtml	http://www.w3.org/1999/xhtml/vocab#	126
8	dc	http://purl.org/dc/	75
9	eClass	http://www.ebusiness-unibw.org/ontologies/edclass/5.1.4/#	38
10	v	http://rdf.data-vocabulary.org/#	35
11	ogp	http://ogp.me/ns# & http://opengraphprotocol.org/schema/	18
12	sindice	http://vocab.sindice.net/date	16
13	rev	http://purl.org/stuff/rev#	15
14	yahoo	http://search.yahoo.com/searchmonkey/commerce/Business	11
15	pro	http://www.productontology.org/id/	4
16	wgs84	http://www.w3.org/2003/01/geo/wgs84_pos#	2
17	fb	http://www.facebook.com/2008/fbmladmins	2
18	powder	http://www.w3.org/2007/05/powder-s#	1
19	inhouse1	http://herbaman.com.ar/Products.html	1
20	inhouse2	http://lokool.com/extendedgoodrelations.owl	1
21	inhouse3	http://www.kica-jugendstil.com/semanticweb.rdf	1
22	inhouse4	http://www.logicpass.com/semanticweb.owl	1
23	inhouse5	http://www.openlinksw.com/schemas/DAV	1
24	inhouse6	http://www.acigroup.co.uk/semanticweb.rdf	1
25	inhouse7	http://www.buntegeschenke.de/semanticweb.rdf	1
26	inhouse8	http://www.sv.arvut.se/sv/kvinna/shopkvinna	1
27	inhouse9	http://www.symbolontarot.nl/de-winkel-met-symbolon-artikelen.html	1
28	inhouse10	http://data.openlinksw.com/oplweb	1
29	inhouse11	http://olutools.com/shop.html	1
30	inhouse12	http://www.wifo-ravensburg.de/rdf/semanticweb.rdf	1
31	comm	http://purl.org/commerce#	1
32	coo	http://purl.org/coo/ns#	1
33	media	http://purl.org/media#	1
34	scovo	http://purl.org/NET/scovo#	1
35	vso	http://purl.org/vso/ns#	1
36	void	http://rdfs.org/ns/void#	1
37	sioc	http://rdfs.org/sioc/ns#	1
38	fibr	http://vocab.org/fibr/core#	1
39	cc	http://creativecommons.org/ns# & http://web.resource.org/cc/license	0
40	vann	http://purl.org/vocab/vann/	0
41	skos	http://www.w3.org/2004/02/skos/core#	0
42	vocab	http://www.w3.org/2003/06/sw-vocab-status/ns#	0
43	rdfa	http://www.w3.org/ns/rdfa#	0
44	g	http://www.w3.org/2003/g/data-view#	0

Figure 5.14: List of ontologies with their prefixes.

5.8.2 Analysing Ontology Usage Distribution (OUD)

Ontology Usage Distribution (OUD) refers to the use of ontologies by data sources in publishing their information. Through this, I would like to determine how the use of an ontology is distributed over the data sources in the dataset. For such distribution, the Ontology Usage Network is analysed to measure the degrees of the nodes.

Observation: Using Eq. (7.7) and (5.5), Table 5.2 shows the percentage of the ontologies being used by a number of different data sources. The relative frequency of OUD on the dataset shows that there is both extreme and average ontology usage by data sources. It also shows that 13.6% of ontologies are not used by any of the data sources and approximately half of the ontologies are exclusively used by the data sources. The second row of the Table 5.2 shows that 47.7% of ontologies (21 ontologies) are being used by a data source that has not used any other ontology. This means that there are several ontologies in the dataset which either conceptualize a very specialized domain, restricting their reusability, or are of a proprietary nature. From the third row of Table 5.2 onwards, there is an increase in the reusability factor of ontologies. This is because an increasingly large number of data sources are using them. The last row shows that 4.5% (2 ontologies) of ontologies are being used equally by 208 data sources. Through this analysis, it can be seen that there are less ontologies which are not being used at all and there are a few which have almost optimal utilization.

Figure 5.15 shows the degree distribution of ontology usage in a number of data sources. The value of degree is shown on the x-axis and the number of ontologies with that degree is shown on the y-axis. It can be seen that there are a large number of ontologies with a small degree value and only a few ontologies have a larger degree value.

Figure 5.14 shows the complete list of ontologies used in the dataset along with their degree (the number of data sources using the ontology). As previously mentioned, vast numbers of ontologies are being used by only one data source which indicates that they are either very specialized nature and/or are proprietary for exclusive use. In Figure 5.14, rows 18 to 38 show the namespaces of these ontologies which cover both very specific domains and some are proprietary. Although the license terms of ontologies assumed to be proprietary were not found, however the non-availability of their specification document makes us believe this is the case.

Table 5.2: Distribution of Ontology Usage in data sources

# of data sources	# ontologies	% ontologies
0	6	13.6
1	21	47.7
2	1	2.3
3	1	2.3
4	1	2.3
11	1	2.3
15	1	2.3
16	1	2.3
18	1	2.3
38	1	2.3
75	1	2.3
115	1	2.3
126	1	2.3
141	1	2.3
164	1	2.3
190	1	2.3
208	2	4.5

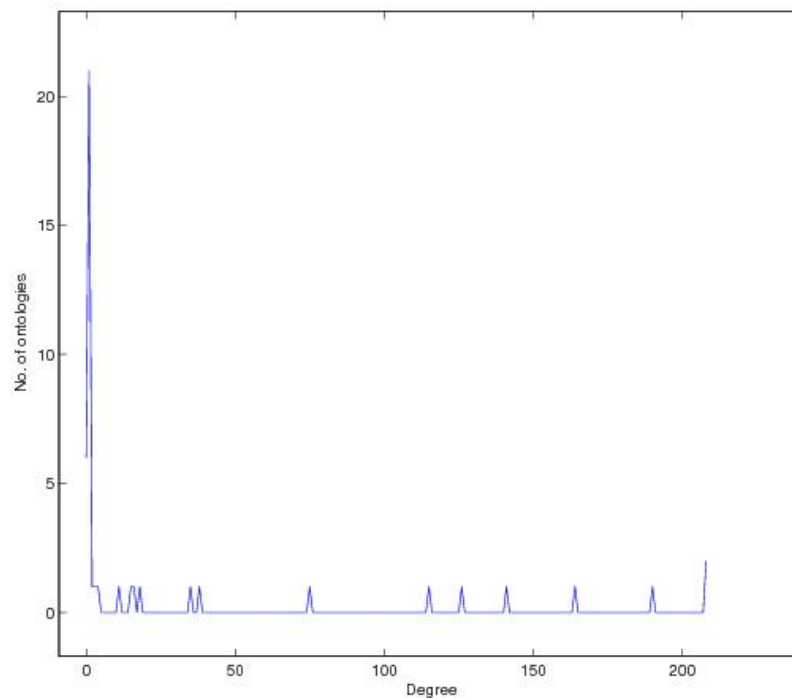
**Figure 5.15:** Degree distribution of ontology usage (data sources per ontology).

Table 5.3: Distribution of Semanticity (Ontology used per data source)

# ontologies	# of data source	% data source
2	1	0.5
3	4	1.9
4	3	1.4
5	38	18.0
6	59	28.0
7	51	24.2
8	39	18.5
9	13	6.2
10	2	0.9
14	1	0.4

5.8.3 Analysing Semanticity

Semanticity measures the richness of a data source in terms of ontology usage. In other words, by semanticity, I mean the ability of any data source (data publisher) in providing semantically rich structured data that is being annotated by one or more ontologies. The assumption is that the higher the number of ontologies being used by the dataset, the more semantically rich the data source is. Semanticity, which is essentially the number of ontologies per data source, is obtained by measuring the degree of the nodes of the data source in the Ontology Usage Affiliation network.

Observation: Using Eq. (5.6) and (5.7), in the OUN, it is observed that on average, 6.6 ontologies per data source are used in the dataset which, in my view, is an encouragingly high semanticity value, particularly bearing in mind that there are several ontologies with very low ontology usage degree values such as 0 and 1, as described in the previous section on Ontology Usage Distribution section. After determining the average semanticity of the data sources, their degree distribution i.e. Eq. (5.7) is observed. Table 5.3 shows the relative frequency of ontologies being used by a number of data sources. The degree distribution of ontology usage per data source is different from ontology usage distribution.

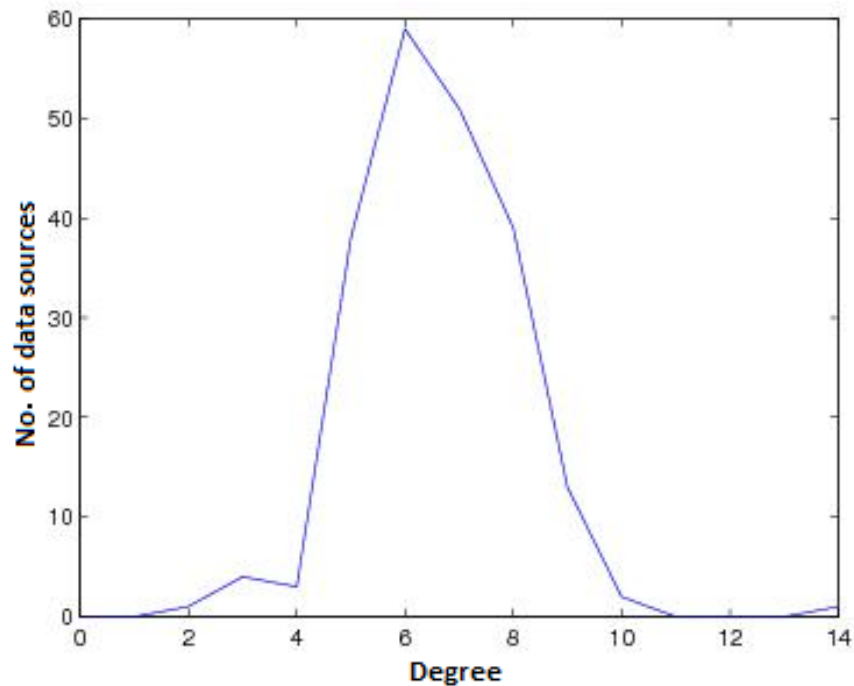


Figure 5.16: Degree distribution of semanticity (Ontologies per Data source).

At the lowest, in the network, two ontologies² are used by one data source, which shows the lowest semanticity value and 14 is the maximum semanticity value which is also used by one data source. When the data sources' degree distribution is plotted, it follows the Gaussian distribution, as shown in Figure 5.16. Gaussian distribution (Weisstein, 2005), which is essentially a bell shaped curve, is normally concentrated in the centre and decreases on either side. This signifies that degree has less tendency to produce extreme values compared to power law distribution. It is believed that Gaussian distribution (also known as normal distribution), which circumvents the exponential growth in degree distribution, is quite helpful in designing the algorithms that need to consume data on the Web from a scalability point of view.

Now, let's look at each set of nodes separately to analyse their characteristics and better understand the relationships emerging within the nodes of the same set. To do this, using the projection process discussed in Section 5.5.2, two networks, namely the Ontology Co-usability (Figure 5.18) and Data-Source Collaboration network (Figure 5.17) from the Ontology Usage Affiliation network are generated.

²www.oetl.it & www.openlinksw.com

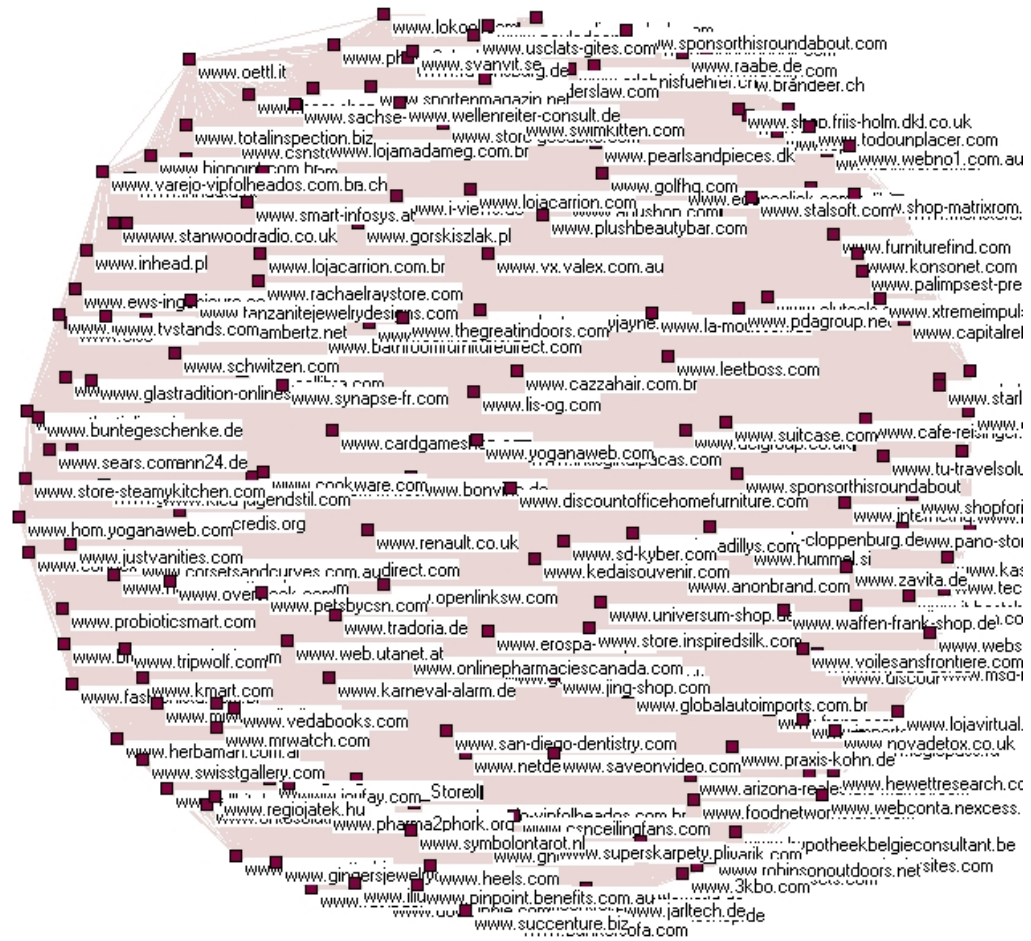


Figure 5.17: Data source collaboration network.

In the following section, we focus only on the Ontology Co-usability network and analyze its properties in detail. A similar analysis can be done with the Data-Source Collaboration network but the details are not presented here because they are not directly relevant to this study.

5.8.4 Formation of Ontology Co-Usability

Ontology co-usability is an undirected graph extracted by projecting the Ontology Usage Affiliation network on an ontology set of node to form an ontology-to-ontology network. In an ontology co-usability graph, ontologies are linked to other ontologies if they are being used by the same data source. Collaboration networks such as the projected Ontology Co-usability network are of coarser representation compared to the affiliation network; however, they are still more informative since many collaboration

patterns are available through these graphs (Franceschet, 2011) such as components and cohesive subgroups (Fershtman, 1997).

Observation: In the dataset which is being analysed, the Ontology Co-usability network comprises 44 vertices and there are 305 edges between these vertices (Figure 5.18). It includes 38 loops which indicate the affiliation of nodes in the network, thus, this tells us that 6 nodes (which means 6 ontologies, that is, nodes without any edges in Figure 5.18) are not being used by any data source at all. For general network properties, the density and average degree (Eqs. (5.2) and (5.1) respectively) of the Ontology Co-usability network is 0.295 and 13.86, respectively. It is interesting to see that though we tend to lose some information through the projection process (two-mode to one-mode), the extracted network is denser (Ontology Usage Affiliation density is 0.149) and has a high occurrence of cliques. Also, the average degree of ontology co-usability is 13.86 which is higher than the original bipartite graph i.e. 10.90. This shows that a large number of ontologies are mutually (collaboratively) used by different data sources having data which is common or related in nature and being semantically similar to each other. This highlights the fact that we can generate a minimum set of vocabularies (URIs) of the interlinked ontologies representing the schema requirements of publishers, facilitating querying and inferencing information efficiently on the Web.

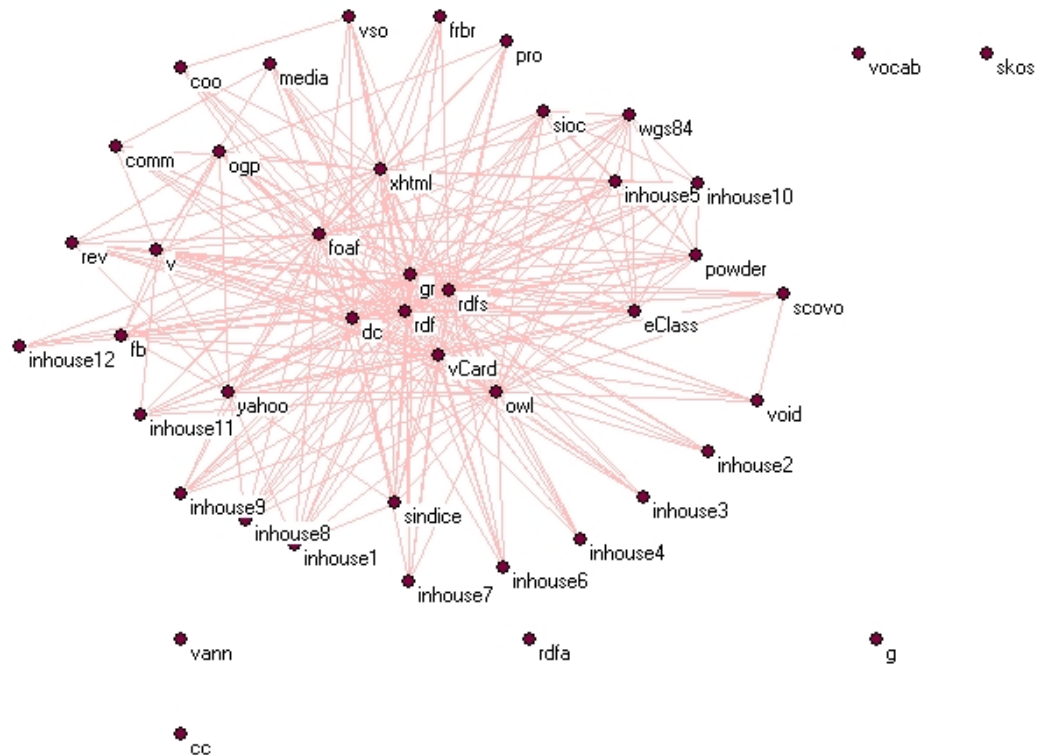


Figure 5.18: Ontology Co-usability network.

In the next section, I will look at the centrality of the vertices of the Ontology Co-usability network to understand which ontologies are more central in terms of betweenness and closeness measures.

5.8.5 Analysing Betweenness and Closeness

As mentioned earlier, betweenness centrality measures the number of shortest paths going through a certain node. This is based on the notion that a node which lies between the shortest paths of many nodes has a central position in the network. Nodes with high betweenness centrality are important for other nodes to reach (communication gateway) since they fall on the geodesic paths between other pairs of nodes.

However, on the other hand, the closeness centrality of a node in the network is the inverse of the average shortest path distance from the node to any other node in the network (Newman, 2008). The larger the closeness centrality of a node, the shorter the average distance from the node to any other node, and thus, this is viewed as the nodes efficiency in spreading information to all other nodes.

Observation: In the context of the Ontology Co-usage network, the interpretation of betweenness and closeness measures are different from other collaboration networks such as co-authorship collaboration networks [Newman \(2004\)](#). In betweenness, which is measured using Eq. (5.8), the nodes of larger values are considered to be the hub of the network, controlling the communication flow (or becoming the major facilitator) between nodes with a geodesic path passing through these hub. In the Ontology Co-usage network, ontologies are linked based on the fact that these are being co-used by the data sources, therefore it is believed that the ontologies with maximum betweenness centrality act as a semantic gateway³ and become a major motivational factor for the usage of other ontologies.

Likewise, in closeness centrality which is measured using Eq. (5.9), the larger the value, the shorter the average distance from the node to any other node, and thus the node (with a larger value) is positioned in the best location to spread information quickly ([Okamoto et al., 2008](#)). This centrality measure in the ontology co-usage graph enables the establishment of correspondence between ontologies which have concepts related to each other, supplementing each others conceptual model to form an exploded domain. The utilization of ontology indexing based on closeness centrality is very similar to the features discussed in ([David and Euzenat, 2008](#)) in supporting the application specific use of ontologies such as:

- i) the ontologies closer to each other in their usage are better candidates for vocabulary alignment,
- ii) ontologies closer to each other have more entities which correspond to entities of other ontologies, and
- iii) closely related ontologies tend to facilitate query answering on the Semantic Web.

The betweenness and closeness centrality of ontology co-usage nodes is shown in Figures 5.19 and 5.20, respectively. The node size in Figures 5.19 & 5.20 reflects the centrality value.

³Semantic Gateway can be considered as the Drug Gateway effect, in which a certain drug becomes the driving force (or reason) for the utilization of other drugs.

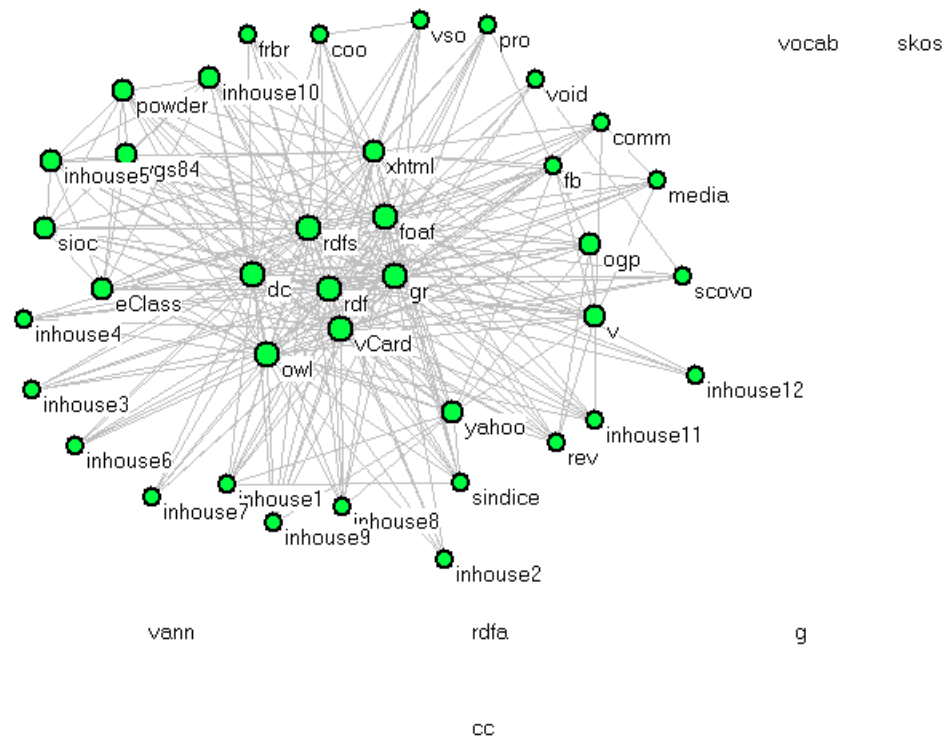


Figure 5.20: Closeness centrality of Ontology Co-Usage network.

5.8.6 Analysing Cohesive Subgroups

A connected component in an undirected graph is a sub-graph in which any two nodes are connected to each other by a path, traversable through intermediate nodes. In a collaboration graph such as the Ontology Co-Usage network, a connected component is a maximal set of ontologies that are mutually reachable (and connected) through a chain of (co-usage) links. The connected components reveal the state of connectedness of the ontologies in the Semantic Web in general and specifically in our dataset (Guéret et al., 2010). It is believed that to promote the reusability of knowledge and allow several conceptual models to interplay on the Web, widely connected components forming a cohesive subgroup of ontologies is a desirable property.

Observation: A cohesive sub-group analysis to identify connected components of the Ontology Co-Usage network shows that the network is widely connected. The connected component is 86.36% (See Figure 5.21, in which only six are not connected in the network (this means 0-core), while others are connected with varying k-core values) making it a giant network since it encompasses the majority of the nodes. This means that 86.36% of the ontologies are reachable to each other by following the links (domain names URIs) of the data sources included in the dataset (or on the Web

to generalize it). Note that the size of the cohesive sub-group, in terms of percentage, closely matches the findings of (Broder et al., 2000) for the classical Web which was 91%.

Within the giant connected component, to know the sub-component based on the equal distribution of the concentration of links around a set of nodes, k-core is computed. k-core is the maximum sub-graph in which each node has at least degree k within the sub-graph. Figure 5.21 stacks the k-core components, based on ascending k values from highest to lowest. From Figure 5.21, it is easy to see which ontologies are highly linked, based on ontology usage patterns invariance across data sources.

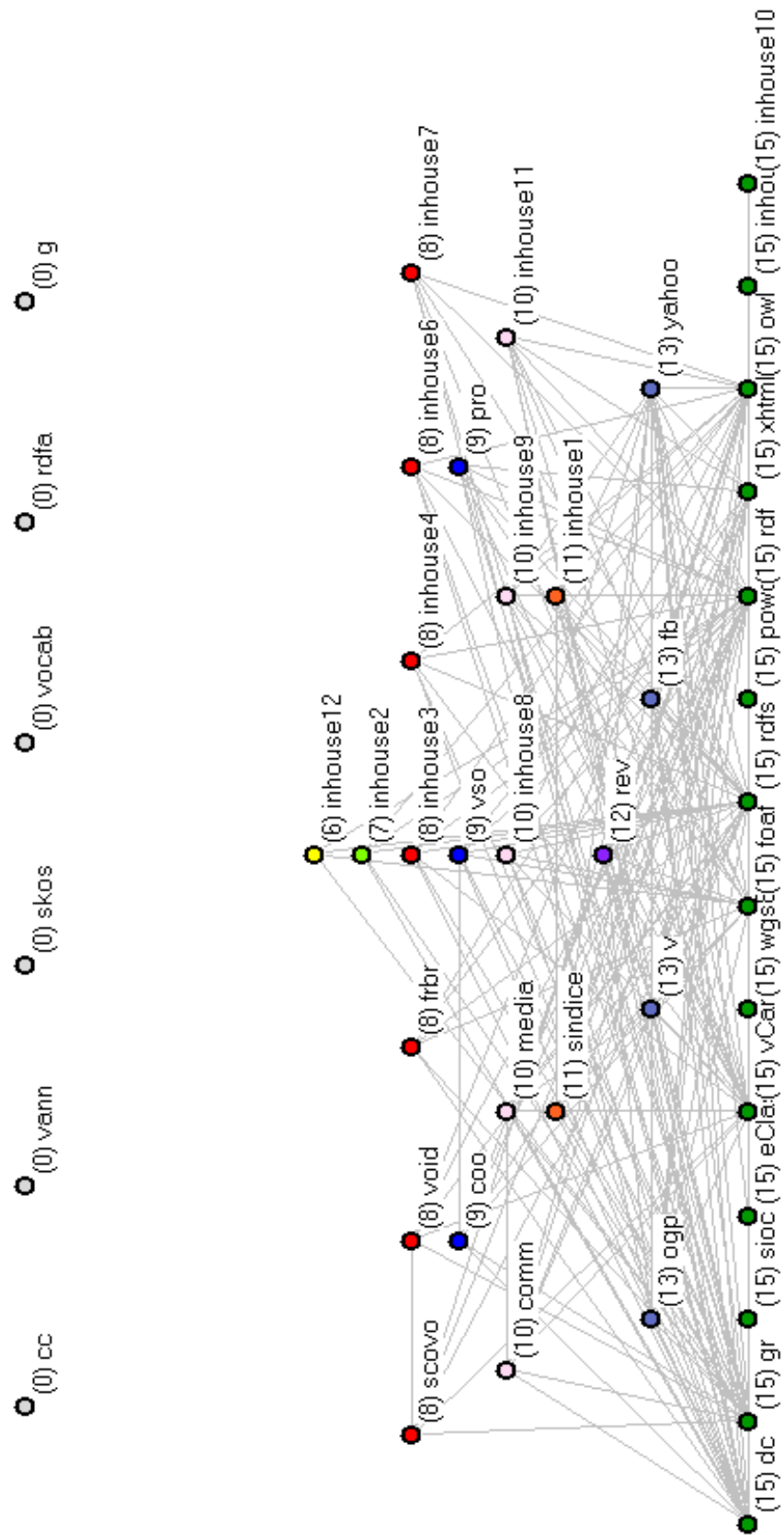


Figure 5.21: Stacking of k-cores of Ontology Co-Usage network.

5.9 Ontology Identification Evaluation

The aim of the ontology identification phase is to identify the ontologies which can be further analysed to understand their usage patterns and trends in detail. The OUN-AF framework provides a model and analysis that is capable of addressing the selection criteria requirements, mentioned in Section 5.3. In the following sub section, how the OUN-AF can assist in identifying different ontologies of interest, according to the selection criteria in two different scenarios, is demonstrated.

5.9.1 Scenario 1: Ontologies and Data Publishers

Let us consider two different scenarios:

Case 1 : From the given dataset, analyse how many data publishers (ontology users) are using a given ontology to describe their domain-specific entities.

Analysis : A more generalized description of this requirement is to understand the distribution of ontologies over the data -sources included in the dataset. To obtain answers for such queries, the *Ontology Usage Distribution* (OUD) metric (Section 5.7.1) is used to measure the degree and its distribution over the OUN. Figure 5.14 lists all the ontologies and their degree values. The degree value of each ontology tells the number of unique data publishers (data sources) are using them. By examining the list, it can be seen that in the dataset (which focuses on the e-Commerce application area) *gr*, *vCard* and *foaf* are being used by 208, 190 and 115 data sources, respectively.

Additionally, the degree distribution plot, as shown in Figure 5.15, helps in understanding how, generally, ontologies are being adopted. It is also observed that the degree distribution, as shown in Figure 5.15, closely follows the power law distribution, albeit not exactly, which, in fact, is the distribution model observed in several information networks, particularly internet (or Web) networks ([Albert and Barabási, 2002](#)). Networks with power-law distribution are also sometimes referred to as scale-free networks ([Barabási et al., 2000](#)) because they tend to be scale-free. Looking at the distribution and following the patterns found in such scale-free networks, it can be safely assumed that the use of ontologies by different data sources follows ‘preferential attachment’ ([Barabási and Albert, 1999](#)).

Case 2 : What other ontologies are being used by the same data publisher to understand the level of semanticity present?

Analysis: In order to understand what other vocabularies are being co-used by a data publisher to semantically describe the entities representing application area, *Semanticity* metrics (Section 5.7.2) are used. From the obtained results based on the dataset used, it can be clearly seen that the top 10 vocabularies which are being co-used by data sources with the highest semanticity value in the eCommerce domain (including standard vocabularies) are: *rdf*, *gr*, *rdfs*, *vCard*, *owl*, *foaf*, *xhtml*, *dc*, *eCl@ss* and *v*.

It is observed that the distribution follows the Gaussian distribution as shown in Figure 5.16 which means that a few of the data sources have higher semanticity than the others which makes it easy to identify the data sources which are publishing semantically rich structured data.

5.9.2 Scenario 2: Ontologies Co-usability

It is always desirable and somewhat interesting to know how things are being interlinked and co-used on the Web. Ontology co-usability, which is produced through the projection procedure over the affiliation network, helps in identifying which ontologies are being co-used with other ontologies and the frequency. This means that the link between two nodes (ontologies) in the one-mode network, produced through projection, tells that these two ontologies are co-jointly being used to describe information by a data source. From Figure 5.18, it is clear that except for six ontologies, all are being co-used which is a positive trend and helps in realizing the Semantic Web vision where entities are not only semantically described but also interlinked with other entities to form the *Web-of-Data* which is processable by computers. Based on the analysis, ontologies which are largely co-used with other ontologies are *gr*, *foaf*, *dc*, and *vCard*. Here, W3C-based vocabularies are not discussed because firstly, these are meta-languages and secondly, they do not represent any domain or application area.

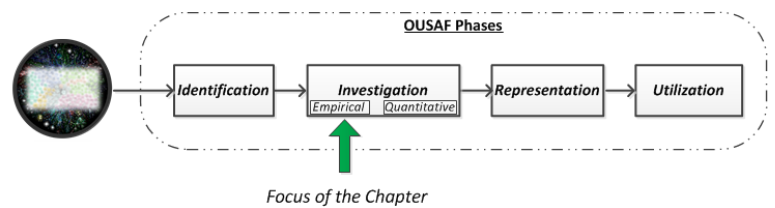
5.10 Conclusion

In this chapter, the OUN-AF framework that assists in the identification phase of the OUSAF was presented. The OUN-AF with its components (dataset, OUN and metrics) helped in obtaining a detailed insight into how different data sources are using particular ontologies and how these are being co-used. The analysis will also assist in understanding how different ontologies are interlinked and their usage

patterns in the dataset. Such insight, based on real instance data obtained by crawling the Web, provides substantial evidence as to how different but related domain-specific ontologies are being co-used by data publishers to provide semantically rich structured data on the Web. The identification of prominent domain ontologies presently prevalent helps data publishers and application developers to consider these to archive the better experience.

In the next chapter, the *investigation phase* of the OUSAF is presented. The output of this phase i.e. identification of ontologies, provides the input to perform a detailed analysis on the "usage" of a given domain ontology.

Chapter 6 - Investigation Phase: Empirical Analysis of Domain Ontology Usage (EMP-AF)



6.1 Introduction

In order to make effective and efficient use of an ontology, it is important to understand how a given ontology is being used by the users and its adoption level. So, the next phase of the OUSAF framework, after the identification phase, is the investigation phase which is responsible for analysing the use of domain ontology(ies) on the Web. There are different types of users of an ontology such as Ontology Owner, Data Publisher and Application Developer and each type of user requires different kinds of insight or information pertaining to the ontology usage as briefly described below:

Ontology Owner : Ontology owner would be interested in knowing the following details:

- What is the adoption level of my ontology?
 - Who is using it?
-

- Which specific components of the ontology are being used?

The answers to these questions will help an ontology owner to evaluate the performance of his/her ontology usage. The availability of such information provides a pragmatic feedback loop to the ontology evolution process, as shown in Figure 1.5. Therefore, having such information is essential for ontologies to remain useful on the Semantic Web.

Data Publishers: Data publishers would be interested in knowing the following details, either for a given ontology or about their application area:

- What exactly is being used by other data publishers from a given ontology?
- Which concepts of a given ontology are being used more and which concepts are being linked using which relationships?
- How is an (domain-specific) entity being attributively described?

The answers to these questions will help data publishers understand what and how ontologies are being used in their respective application areas. The availability of such information is necessary for data publishers to realize the benefits that he/she will achieve by reusing existing ontologies. As mentioned in Section 1.4.2, by adopting (or reusing) used ontologies a positive network effect which means increasing the overall perceived utility of ontologies is achieved. Furthermore, the increased use of an ontology by the community helps it to become the defacto structure (or schema) to represent the respective application area (or domain) (Ashraf et al., 2011).

Application Developer : In order to effectively and efficiently consume Semantic Web data (published on the Web), application developers need to know:

- What terminological knowledge of an ontology is available on the Web to use?
- Which concepts of a given ontology are being used more and how are these concepts being interlinked (using which relationships)?
- What are the common data and knowledge patterns available?
- How are entities being annotated or textually described?

The answers to these questions are important to the application developer because they provide a snapshot of the prevailing schema of the structured data published on the Web, allowing developers to program routines accordingly for the efficient and

effective retrieval and consumption of semantically rich data. Knowing how entities are being described helps developers query specific information about entities and develop the operation and interfaces accordingly.

To obtain such an erudite insight into the use of ontologies from different perspectives for the different groups of users, a framework to analyse domain ontology usage is needed. The proposed framework needs to be based on *real world instance data* to provide a practical insight from different perspectives and should *cover different aspects* to fulfil the needs of a wide range of users. There are two different ways to perform such an analysis that is capable of providing the required information and insight regarding ontologies, which is explained in the next section.

6.2 Different ways of Analysing Domain Ontologies

As mentioned in earlier chapters, the use of ontologies to semantically describe the data on the Web has recently picked up pace to take advantage of the benefits offered by Semantic Web technologies. However, being in the early stage of adoption, there is limited understanding of how ontologies are actually being received by the end users. For example, a data publisher makes use of a certain portion of the ontology and its components, based on his/her requirements, which could be different to the components used by other data publishers, depending on their requirements. But the need to understand such usage patterns by different data publishers and other users is important, as explained in the previous section. In order to comprehensively understand how different users are using ontologies and what is the prominent and prevalent structure emerging through their usage, a *neutral observational approach* is required that provides an impartial empirical perspective on their usage. While such impartial insight is necessary to understand the use of ontologies, in order to translate these neutral observations into actionable knowledge, a *quantitative measures approach* to ontology usage is required to determine the usage of domain ontologies. These two different but interlaced approaches to analysing ontology usage (see Figure 6.1) provide a multi-view of the ontology usage landscape. These two approaches are briefly described in the next subsections.

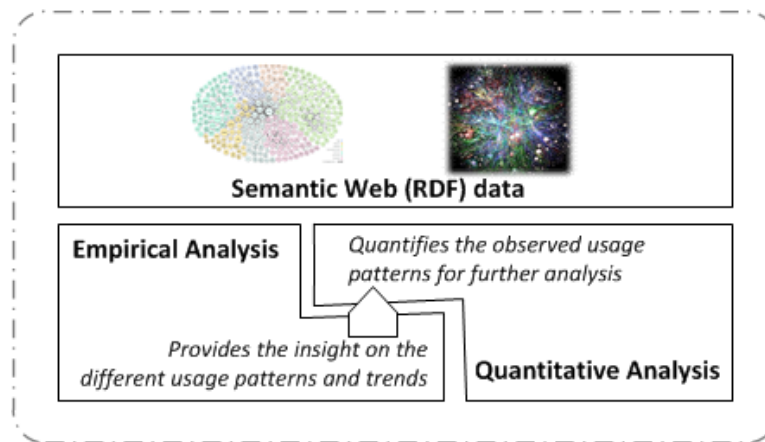


Figure 6.1: Two different ways to analyse ontology usage.

6.2.1 Empirical Analysis of Domain Ontology Usage

Empirical analysis is aimed at obtaining a neutral observation on the Semantic Web data to identify the prominent and prevalent structures emerging from the present use of domain ontologies. The need for empirical analysis at this stage of ontology adoption is rightly noted by [Herman \(2011\)](#) (W3C Semantic Web Activity Lead):

"[...] we are at the point when we can measure what we got, and we can therefore come up with empirical data that will help us to concentrate on what is essential [...]"

The empirical analysis of ontology usage on Semantic Web data, which essentially comprises schema-level and instance-level data, needs to analyse and extract patterns of ontology usage. While observing the schema-level data, which takes the form of terminological statements (T-Box) and instance-level data which takes the form of assertional statement (A-Box), it is important to decide on the *aspects* which need to be observed. In this case, aspects refers to the different viewpoints from which Semantic Web data has to be analysed. Each aspect offers a unique set of requirements necessitating different approaches and techniques to explore them. Terminological knowledge, which is encoded in the RDF statements by making use of the URI references defined by the domain ontologies, is considered during the empirical investigation to analyse the key aspects of ontology usage. The important aspects that are relevant to terminological knowledge analysis and helpful in addressing the requirements of different users (described in Section 6.1) are as follows:

1. **Understand how different vocabularies are interlinked at the instance level:** At the schema-level, this involves how different terminological statements originating from different ontological namespaces are being used to describe

domain-specific entities. On the Semantic Web, the RDF data model and ontologies allow linking decentralized entities across different sources and domains. An understanding of how entities are linked at the schema level across various ontologies helps in extracting schema patterns and analyzing entity linkage that exists within the dataset (Nikolov et al., 2010). For ontology owners as well for application developers, it is useful to know the relationships present at the schema level to understand the users approach toward semantically describing the domain entity as well as to prepare routines to query them accordingly.

2. **Understand how a concept is instantiated and described:** How are the pivotal concepts which represent the core elements of the domain used to describe the entities? In order to establish a thorough understanding of the use of pivotal concepts, it is important to know its instantiation, what other concepts contribute to its semantic description, what relationships it maintains with other concepts and what attributes are used to provide factual knowledge.
 3. **Understand the availability of textual description for human readability:** In order to allow an (semantic) application developer to consume the information distributed across remote systems and develop interfaces for human interpretation, knowledge regarding the use of textual description is important. Information about the presence of annotation and labelling properties enables application developers to develop data-driven interfaces which are quite different from the classical form-based interfaces (Davies et al., 2010). Ell et al. (2011) listed a few of the benefits of labels which include displaying human readable information instead of displaying URIs, using labels for indexing (Ashraf et al. (2011) also highlighted similar benefits) and support for keyword and question-based searches over the web of data.
 4. **Understand the data and knowledge patterns prevalent in the dataset:** Whether querying an anonymous dataset (triple store) whose schema is not known (unlike in traditional databases (RDBMS) where schema is known) or posing a federated query over the Semantic Web, it is very helpful and handy to have some idea in advance about the nature of the data expected from the data source. For example, a prototypical query based on common patterns invariantly appearing across several data sources helps in generating a relaxed (generalized) query to start exploring the dataset. Therefore, it is helpful to have some understanding about the knowledge and data patterns available in the dataset to generate prototypical queries.
-

To empirically understand the use of domain ontologies in relation to these aspects, the **EMPirical Analysis Framework (EMP-AF)** is proposed.

6.2.2 Quantitative Analysis of Domain Ontology Usage

While the above empirical analysis provides an overview of ontology usage from a neutral perspective to understand the use of domain ontologies, in order to take these impartial observations into actionable knowledge, one needs to quantify the observation. In other words, empirical analysis identifies the *key factors* involved in proliferating and driving ontology adoptions, but to utilize the key factors so that they can be used in various scenarios such as ranking, indexing and querying the information, they need to be quantified. These key factors lead to the development of more focused metrics to measure ontology usage by considering the conceptualised model represented through the ontology, the use of the conceptual model and the motivational factors involved in ontology adoption.

To undertake such quantified analysis of ontology usage on the Web, the **QUAntitative Analysis Framework (QUA-AF)** is proposed.

In this chapter, the EMP-AF is discussed and the QUA-AF is discussed in the next chapter. The remaining sections of this chapter are organized as follows. Section 6.3 presents the EMP-AF framework and its two phases, namely the data collection and aspect analysis phase. Section 6.4 defines the metrics used to empirically analyse the use of domain ontologies. To explain the working of the EMP-AF framework, in Section 6.5 a case study is described which will be then used as an example to analyse domain ontology usage. Section 6.6 discusses the implementation of the data collection phase and details the dataset characteristics of the case study. Section 6.7 provides details on the results obtained by analysing domain ontology usage, based on the metrics developed as part of EMP-AF. Section 6.8 presents a discussion on the analysis of the EMP-AF framework by considering the requirements of different types of users, as discussed in Section 6.1. Finally, Section 6.9 concludes the chapter.

6.3 EMPirical Analysis Framework (EMP-AF)

In order to empirically analyse the usage of domain ontologies, the **EMPirical Analysis Framework (EMP-AF)** framework is proposed. The framework comprises two phases, namely the *data collection phase* and the *aspects analysis phase*, as shown in Figure 6.2. The data collection phase is responsible for collecting the

real world instance data necessary for empirical analysis, whereas the aspect analysis phase is responsible for analysing the use of domain ontologies from different aspects to obtain the insight required by different users, as mentioned in Section 6.1. In the next subsections, the objectives and working details of each phase are presented.

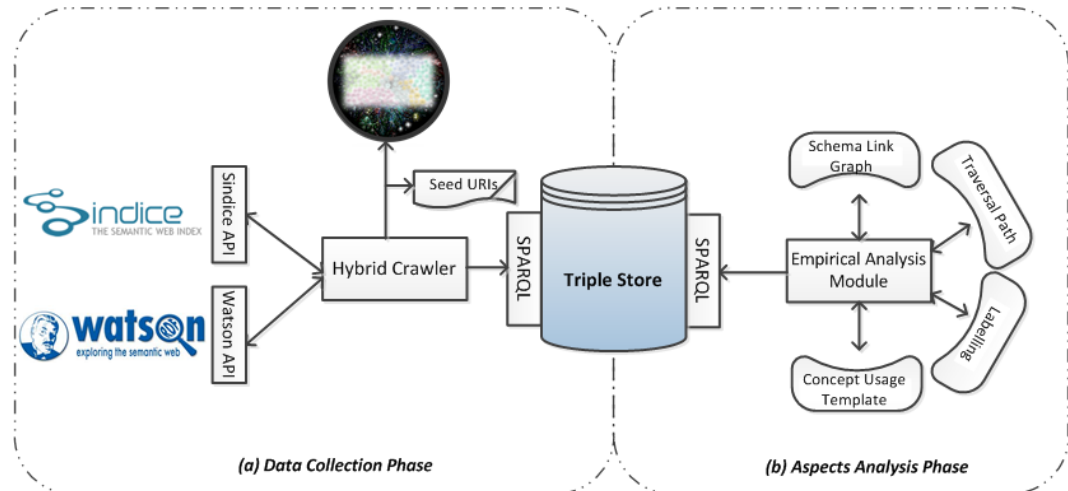


Figure 6.2: Empirical Analysis Framework (EMP-AF).

6.3.1 Data Collection phase

As mentioned earlier, in order to obtain erudite insight into the use of ontologies and their components in a real world setting, it is of paramount importance that data are collected from the data sources that are using domain ontologies to describe their data. The identification phase of the OUSAF framework (Chapter 5) provides the candidate ontologies which are being used by data publishers in a given application area. Identifying these ontologies helps to find potential data sources which use these ontologies which can be included in the data collection phase of the EMP-AF.

The data collection process (see Figure 6.2(a)) crawls the Web to collect the Semantic Web data published by the different data publishers. This means that the crawler that is responsible for collecting the data needs to be aware of the different ways by which structured data is published on the Web and the different serialization formats being used on the Web to publish Semantic Web data. Aside from the different infrastructure requirements, the crawling process should be able to deal with network issues which may arise during the crawling process.

In order to address the abovementioned requirements and gather real world instance data described using ontologies, a hybrid crawler is developed as part of the EMP-AF framework. The details of hybrid crawler and the collected dataset are described in Section 6.6.

6.3.2 Aspects Analysis phase

The aspect analysis phase (see Figure 6.2(b)) of the EMP-AF framework focuses on the execution of empirical analysis. This phase comprises the Empirical Analysis Module which implements four metrics to investigate the dataset from different aspects and a data access component to evaluate SPARQL queries. The four metrics implemented as part of the Empirical Analysis Module are introduced below.

1. In order to understand how different vocabularies are interlinked at the instance level, the first metric to consider is the **Schema Link Graph (SLG)** which reveals the relationship between different vocabularies at the instance level, based on the use of terminological statements (of ontologies) in the dataset. Hence, SLG addresses the first requirement of empirical analysis by helping ontology owners and application developers understand the semantic relationships present on the Web in the given application area and to use these for further processes.
 2. In order to understand how a concept is instantiated and described to obtain a detailed usage analysis, the **Concept Usage Template (CUT)** is proposed. It captures the instantiation of concepts, the relationships it has and the data properties used to describe it. It also captures the different vocabularies being co-used with this concept. This detailed multi-perspective insight provided by CUT helps all types of ontology users glean relevant information.
 3. In order to understand the availability of textual description for human readability, the **labelling** aspect is proposed. It captures the use of properties for labelling purposes. Labelling benefits application developers by helping them better understand the available textual descriptions, as mentioned in the third requirement of empirical analysis. As good practice, data sources make use of labelling properties which are either part of the standard vocabularies or popular in the community, therefore, while formulating the labelling properties, one needs to consider all these different usage patterns.
 4. In order to understand the data and knowledge patterns prevalent in the dataset, the **Traversal path** structure is constructed to capture the prevalent knowledge patterns in the domain ontology usage and understand the invariant patterns available to assist in accessing information. Traversal paths extract the knowledge and data patterns available in the dataset to facilitate the generation of prototypical queries, as mentioned in the fourth requirement of empirical analysis. Traveling the graph, especially an RDF graph which is a multi-edge
-

and directed graph, is a computationally expensive operation therefore, in order to find the occurrence of different patterns, one needs to consider a some preprocessing stage to reduce the overall computation time.

These metrics help in addressing the requirements of different users (discussed in the introduction section) and the aspects highlighted in Section 6.2.1. Section 6.4 formally describes these metrics.

In the next subsection, the set of sequential activities carried out as part of EMP-AF framework is presented.

6.3.3 Sequence of EMP-AF activities

The EMP-AF comprises two phases, namely the data collection and aspect analysis phases. Each phase involves a certain number of activities to carry out the required functionality. In order to provide an overview of the flow of the activities and their sequence, a summary is presented as depicted in Figure 6.3.

- In the data collection phase, a dataset relevant to an application area (domain focused) is collected.
 - The hybrid crawler is implemented to crawl the relevant Semantic Web data.
 - The crawled data is populated into the triple store.
 - The data is analysed using the metrics defined in the aspect analysis phase.
 - In order to reduce the computation cost of the resource intensive operation, preprocessing is done.
 - Using SLG, the relationships present in the dataset are analysed.
 - Using CUT, the use of pivot concepts are analysed.
 - The labelling present in the dataset is observed.
 - The knowledge patterns are observed by constructing the traversal paths
 - The results are analysed to infer the use of domain ontologies on the Web.
-

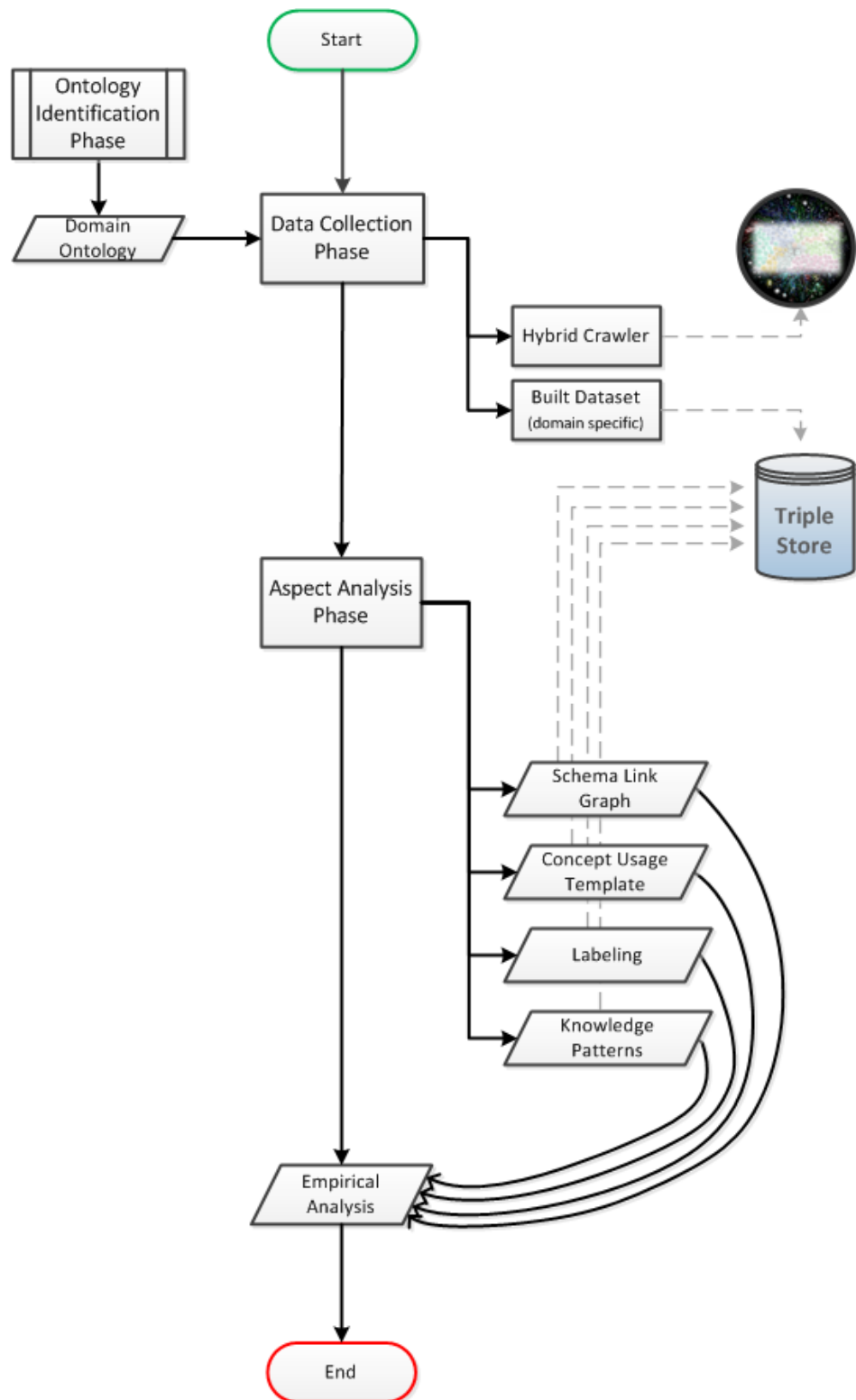


Figure 6.3: Flow of activities in EMP-AF.

6.4 Metrics for EMP-AF

In this section, the metrics used for empirical analysis as part of the EMP-AF framework are presented. Additionally, to explain the analysis obtained from each metric, a sample RDF Graph (Figure 6.4(a) for SLG and Figure 6.5 for other metrics) is used which provides an overview of the computation process and results obtained from each metric.

Before proceeding with the discussion on the metrics, the preliminaries necessary for the metrics are defined.

6.4.1 Preliminaries

The generic preliminaries which are applicable to the whole thesis are described in Section 4.2, however, the terms and definitions specific to the discussion in this chapter are presented as follows.

RDF Triple (triple) A *triplet* $:= (s, p, o) \in (U \cup B) \times U \times (U \cup B \cup L)$ is called an RDF triple, where s is called subject, p predicate, and o object.

Class I refer to a class as an *RDFTerm* which appears in either

- o of a triple t where p is `rdf:type`; or
- s of a triple t where p is `rdf:type` and o is `rdfs:Class` or `owl:Class`

Property I refer to a property as an *RDFTerm* which appears in either

- p of a triple t ; or
- s of a triple t where p is `rdf:type` and o is `rdf:Property`

Instance of a Concept (C) A triple $t = (s, p, o)$ or set of triples in the dataset is an instance of a triple pattern $t_c = (s_c, p_c, o_c)$ if there exist

- s_c is URI Reference
- p_c is `rdf:type`
- o_c is the *class* (Concept) of domain ontology

In the next sub-section, the metrics used in the aspect analysis phase to empirically analyse ontology usage are presented. Additionally, the analysis obtained from each metric is explained with the help of an RDF graph.

6.4.2 Schema Link Graph (SLG)

The Schema Link Graph (SLG) is an undirected graph consisting of a finite set of vertices V and a set of edges E , representing a link between the two vertices. SLG is used to study the relationship between different ontologies in describing entities. Formally, the Schema Link Graph is defined as follows:

Schema Link Graph (SLG) : *The Schema Link Graph (SLG) is a tuple (V, E) , where n is a node ($n \in V$) such that n is the ontology namespace used in the dataset. By ‘used’, it means the presence of a triple where n appears as an object (for instantiation with *rdf:type*), or in a predicate to describe the object. E is the edge set and $e \in E$ is an edge of graph V linking two nodes n_1 and n_2 such that either there is a triple which entails that n_1 is the namespace of the subject and n_2 is the entailed namespace of the object or there is an m sequence of triples connected through a blank node such that n_1 is the entailed namespace of the subject of the first triple and n_2 is the namespace of the object in the m -th triple where $m > 1$.*

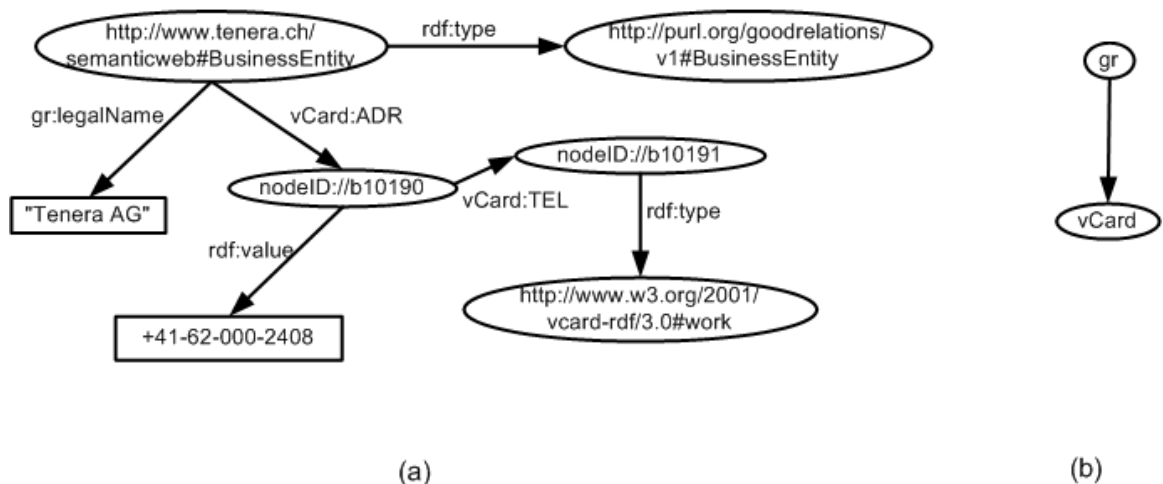


Figure 6.4: (a) Sample RDF graph from the dataset with blank nodes and (b) the corresponding Schema Link Graph.

Example : For example, Figure 6.4(a) shows an RDF graph snippet extracted from the RDF graph representing the semantic data published by `http://www.tenera.ch`. Here, the sequence of triples (subject, predicate, object) is connected to semantically describe the entities (resources) using ontologies. In the sample RDF graph, there are schema level triples creating individuals of type class defined in the domain ontology and the instance level triples describe the entities. For the construction of SLG, triples with the *rdf:type* predicates are retrieved and if there is a relationship joining the resources instantiated using different namespaces

(ontologies), a link between these two ontologies is created. As shown in Figure 6.4(a) <http://www.tenera.ch/semanticweb#BusinessEntity>, a resource is defined by the GRO concept and linked using `cVard:ADR` with a resource of `vCard` vocabulary. Therefore, in the resultant SLG, there are two nodes *gr* and *vCard* with an edge. Figure 6.4 (b) shows that there is a URI in the RDF graph of type *gr* directly or indirectly (through blank nodes) connected with the URI of type *v*.

6.4.3 Concept Usage Template (CUT)

The Concept Usage Template (CUT) captures how a concept is used in the dataset and what properties (both domain ontology predicates and other predicates) are used to describe the entities instantiated by the concept. The template attempts to capture the ubiquitous patterns available and arranges them to facilitate the processing of information for specific purposes, such as searching, browsing, querying and reasoning.

The template captures six aspects of the concept usage. It looks at the RDF graphs available in the dataset and analyses the concepts instantiation, the use of different vocabularies in describing entities, the presence of different relationships, the use of different data properties, the use of other concepts to describe the same entity and the presence of other constructs to provide additional and supplementary information about the entities represented by the concept.

CUT comprises the following metrics:

6.4.3.1 Concept Instantiation

This refers to the number of instances instantiated by the class representing the concept. This gives us the number of entities available in the dataset and reflects the dominance of the entity in the dataset when compared with templates of other concepts. In most Web ontologies, subsumption axioms are used to provide the taxonomical relationship between concepts and with inference, provisioning the concept instantiation may fluctuate depending on where the concept falls in the taxonomy hierarchy. Since most triple stores implement RDFS entailment rules, it is safe to consider the `rdfs91` rule while measuring the instantiation of a top level concept in the taxonomic hierarchy. The *concept instantiation (CI)* of a concept *C* is given as follows:

¹IF(<v subClassOf w> and <u type v>) THEN < u type w>

$$CI(C) = |\text{triples}|$$

where,

$$\begin{cases} s = RDFS\text{Term} \\ p = rdfs:type \\ o = \text{class defined by ontology} \end{cases} \quad (6.1)$$

In the case of subsumption axioms (Gomez-Perez and Corcho, 2002), the $CI(C)$ can include the instances instantiated by the sub-concepts (subclasses) of o such that :

$$o = entail_{rdfs}(C) \quad (6.2)$$

where $entail_{rdfs}(C)$ is a function which implements the *RDFS9* rule:

IF (uuu rdfs:subClassOf xxx AND vvv rdfs:type uuu) THEN (vvv rdfs:type xxx)

$CI(C)$ returns the numeric value representing the number of entities defined by the concept and its sub-concepts.

6.4.3.2 Vocabs

Vocabs provides the list of ontologies (other than the domain ontology) used to describe the entity. Ontologies are represented here with their namespace prefixes and include both the predicates ontology prefix and the concepts prefix to which it is linked. Vocabs help in understanding the different ontologies which are co-used in describing different aspects of the entity. Formally, *Vocabs* is defined as:

Definition: *Vocabs* is a set of namespaces (empty possible) of the vocabularies used in a *triple* such that o is the domain ontology concept and p is the URI reference of the ontology other than the domain ontology used to describe the s .

$$Vocabs = \{vocab_1, vocab_2 \dots vocab_n\} \quad (6.3)$$

such that $vocab_i$ is the namespace of the p 's URI reference.

6.4.3.3 Object Property Usage

This provides a list of relationships available to describe the entity by relating it to other sets of entities and resources. This includes the properties defined by the domain

ontology as well the properties of the ontologies listed in *Vocabs*. Object property usage allows an understanding of the available information pertaining to the entity and its richness by exploring the entities linked to it through these properties.

$$\begin{aligned} ObjectPro(C) &= \{pre_1, pre_2 \dots pre_n\} \\ \text{Such that } pre_i &= Property \end{aligned} \tag{6.4}$$

The *ObjectPro(C)* set contains the *URI references* representing the object properties defined by the ontologies belonging to *Vocabs*.

6.4.3.4 Attribute Usage

This provides the textual information about the entity. This may include the RDF label properties and the data type properties of the domain ontology and non-domain ontologies. A textual description linked with entity instance is useful information for data processing and the user interface.

$$Attri(C) = \{att_1, att_2 \dots att_n\} \tag{6.5}$$

such that $att_i \in (L_p \cup L_t)$

The *Attri(C)* set contains the *URI references* representing the datatype properties defined by the ontologies belonging to *Vocabs*.

6.4.3.5 Class Usage

Class usage records the list of other concepts of which the entity is a member. This allows more to be learned about the entity as different concepts when used to instantiate the same entity and define the broader view reflecting the reality being represented by the entity. It is believed that class usage provides the conceptual overlap which exists between related but different concepts formalized by different ontologies and can be exploited to generate semantic mapping between related terms.

$$\begin{aligned} \text{ClassUsage (C) is set of classes such that there exist a triple in} \\ \text{the dataset where } p = \text{rdf:type and } o \text{ is } \mathbf{class} \text{ and } o \neq C \end{aligned} \tag{6.6}$$

6.4.3.6 Interlinking

Interlinking provides a list of linking properties used to create links across different datasets. An example of such links is link base and equivalence link (Dodds and Davis, 2010). Here, the main focus is on equivalent links which helps to specify the different URIs which refer to the same entity or resource. Semantic Web languages provide built-in support for creating equivalent links between different component of the ontology and data. The resources and entities are linked through the `owl:sameAs` relation which tells applications that these two resources (subject URI and object URI) are describing the same entity and their data can be merged to obtain an exploded view of the entity. So, interlinking is obtained by identifying the use of any interlinking property for a given entity.

Example : The abovementioned analysis methods and metrics are explained by using a sample RDF graph. Figure 6.5 shows the sample RDF graph of a fictitious “Example.com” data source. The RDF data describes a company which is in the car sales business. The triples in the RDF graph represent information regarding the business entity, its shop/office location (address), and the offers and products included in the deal. For the sake of brevity and readability, relevant triples in turtle syntax are listed and will be used in this section for discussion and explanation. Lines 1 to 7 of the sample RDF code contains the prefixes which are used in the triples to access the vocabulary (or terms) defined by their respective namespaces to describe the resources (entities). Lines 8 to 37 of the sample RDF code describe the different resources linked through relationships in order to semantically describe the entities.

```

1  @base <http://www.example.com/websource#>
2  @prefic gr:<http://purl.org/goodrelations/v1#>
3  @prefix dc:<http://purl.org/dc/terms/>
4  @prefix vso:<http://purl.org/vso/ns#>
5  @prefix coo:<http://purl.org/coo/ns#>
6  @prefix foaf:<http://xmlns.com/foaf/0.1/>
7  @prefix vCard:<http://www.w3.org/2001/vcard-rdf/3.0#>
8  :cardealer
9      rdf:type gr:BusinessEntity ;
10     dc:title "business entity and car data";
11     dc:date "2009-09-12";
12     gr:legalName "The Example Company";
13     owl:sameAs <http://www.acme.com/example>;
14     foaf:homepage <http://www.example.com>;
15     rdfs:seeAlso <http://www.example.com/about.pdf>;
16     gr:offers ex:Offering_1.
17 ex:Offering_1
18     rdf:type gr:Offering;
19     gr:includes ex:product_1;
20     rdfs:comment "Eco Car on sale"@en;
21     gr:availableAtOrFrom ex:location;
22     gr:category "Used Car".
23 ex:product_1
24     rdf:type gr:ProductOrServiceModel;
25     rdf:type vso:Automobile;
26     rdf:type coo:Derivative;
27     gr:name "The Blue Car";
28     gr:category "Automobile";
29     gr:color "Red".
30 ex:location
31     rdf:type gr:LocationOfSalesOrServiceProvisioning;
32     vCard:ADR [
33         vCard:Street "2253 Jackson Ave.";
34         vCard:Pcode "00553";
35         vCard:City "New York";
36         vCard:Country "US".
37     ].

```

Figure 6.5: Sample RDF code for discussion

The CUT of the entity (i.e. `ex:cardealer`) of type `gr:BusinessEntity` is shown in Table 6.1. In the sample RDF graph, `ex:cardealer` is the business entity which is an instance of the type `gr:BusinessEntity` class defined in the GoodRelations ontology. The value of the concept instantiation using Eq. (6.1) is $CI(C) = 1$, as there is only one instance of the type `gr:BusinessEntity`. *Vocabs* is the set of prefixes used to describe the entity and in this example, Eq. (6.3) returns $Vocabs = \{gr, dc, foaf\}$. Note that in *Vocabs*, the prefixes of W3C-based standard languages

Table 6.1: Sample RDF Graph: CUT of gr:BusinessEntity (ex:cardealer)

Entity	gr:BusinessEntity
Instantiation	1
Vocabs	gr, dc, foaf
Object properties	gr:offering
Attributes Usage	de:title, de:date, foaf:homepage
Class Usage	
Interlinking	rdfs:seeAlso

such as RDF, RDFS and OWL are not considered to be the focus is more on the domain ontologies. In the case of object property usage, Eq. (6.4) returns $ObjectPro(C) = \{gr : offering\}$ and for attribute usage, Eq. (6.5) returns $Attr_i(C) = \{dc : title, dc : date, foaf : homepage\}$. The class usage of the product entity (i.e. ex:product_1, line 23) of type gr:ProductOrServiceModel returns the set of classes of which the entity is also member, i.e $ClassUsage(C) = \{vso : Automobile, coo : Derivative\}$ (not shown in Table 6.1 which covers the CUT of gr:BusinessType. Additionally, link base and equivalence links are provided to allow users to access additional relevant information (rdfs:seeAlso; line 15) and explode the information about the entity by merging the description published on two different locations (owl:sameAs; line 13), i.e $Interlink(C) = \{rdfs : seeAlso, owl : sameAs\}$.

6.4.4 Labelling

Labels refers to the textual information provided with the entity description to allow a better understanding of the entities before these entities are processed by Semantic Web applications. The emphasis is to analyze how labelling properties are used with entity description, which is helpful for information retrieval and presentation.

While analyzing the entity, I look at the use of different label properties in the data and discuss their usefulness in scenarios such as finding hidden information from the label text, using language tags to facilitate the internationalization of semantic applications and developing the user interface for information which is syntactically published for machine consumption.

6.4.4.1 Formal Labels

RDFS specification provides two properties; rdfs:label and rdfs:comment to provide human-readable information about the resources. The former is normally used to

provide a human-friendly version of the resource name which is an opaque URI otherwise, and the latter is used to present a human readable description of the resource. These two label properties are referred to as **formal label (fl)** while analyzing the presence of label properties in the dataset in general and in the entity description specifically. Such online documentation on resources is very useful and often domain ontologies define more specific labelling properties.

The following metric is defined to measure the use of *fl* for each pivotal entity. $Entity_{fl}$ measures the ratio of entities with at least one formal label to all pivot entities in the dataset.

If C is the concept of the domain ontology (class) then:

$$fl = \{rdfs:label, rdfs:comment\} \quad (6.7)$$

$$Entity_{fl}(C) = \text{number of instances}(C) \text{ with fl} / \text{total number of instances}(C)$$

6.4.4.2 Domain Labels

There are two common practices for defining domain ontology label properties: first, by describing label properties as the subproperty of `rdfs:label` using the subproperty axiom (subsumption), and second, by having a datatype property with `rdfs:Literal` as its range. In some cases, the label properties are defined by specifying literal datatype and in such cases, `xsd:string` datatype is used. Here, these domain-ontology-defined label properties are referred to as **domain labels (dl)**. [Ell et al. \(2011\)](#) proposed label-related metrics to measure the completeness, the efficient accessibility of label properties and the unambiguity of the labels in the knowledge base. These metrics help in quantifying the presence of labels in a dataset, however to understand their usefulness in the real setting for information retrieval and presentation purposes, one needs to analyze label properties for each pivotal entity and discuss their usefulness.

Likewise, $Entity_{dl}$ computes the ratio of entities with at least one domain ontology label to all pivot entities in dataset. The sum of these two measures tells us how rich a particular concept (pivot entity) is in terms of labels. If C is the concept of the domain ontology (class) then:

$$dl = \{i \mid i \text{ is the label property defined in domainontology}\} \quad (6.8)$$

$$Entity_{dl}(C) = \text{number of instances}(C) \text{ with dl} / \text{total number of instances}(C)$$

Example : To use an example to explain what labels are available and how they are used in the knowledge base by using metrics, namely $Entity_{fl}$ and $Entity_{dl}$,

let us refer back to the sample RDF graph (see Figure 6.5). The focus is on `gr:BusinessEntity` as the pivot concept, the label metrics for the entity of type `gr:BusinessEntity` is measured using Eq. (6.7) and (6.8), respectively.

$$Entity_{fl} = 0/1 = 0$$

$$Entity_{dl} = 1/1 = 1$$

The label attributes used for the description of the `ex:cardealer` entity are available/listed from lines 9 to 16 of the sample code. For $Entity_{fl}$, only RDFS-based label properties (i.e `rdfs:label` and `rdfs:comments`) are considered and none of them is used in this particular example. There is only one instance of entity (individual) of type `gr:BusinessEntity` therefore $Entity_{fl}$ equals zero. Likewise, for $Entity_{dl}$, `gr:legalName` predicate usage which is a domain ontology label property (the complete list of domain ontology labels are discussed in Section 6.7.1) is present, therefore the value of $Entity_{dl}$ is 1.

6.4.5 Knowledge Patterns (Traversal Path)

A traversal path determines the sequence in which properties are used to access the description of related entities within a given context. A traversal path starts with the instance of the entity class in focus and follows the available sequence of instance-property-instance triples to record all the paths in the dataset. The following metrics pertaining to traversal paths are defined.

6.4.5.1 Unique paths

Unique paths computes the number of unique paths leading from the entity (out links). One entity can have zero or many paths of varying lengths, depending on the RDF graph in the dataset. A complete set of unique paths helps in understanding the data patterns available which can further assist in querying the dataset.

6.4.5.2 Average Path Length

Average path length helps in understanding the entity description depth available in the dataset

6.4.5.3 Max path length

Max path length helps in understanding the maximum possible description depth available in the knowledge base.

6.4.5.4 Path steps

Path steps helps in identifying the triples found in the traversal paths.

In traversal paths, unique paths available in the RDF graph (or dataset) and the maximum and average traversal path lengths are computed. The traversal path procedure constructs the list of all available paths in the dataset and this list of paths is then used to compute the maximum and average path length. Additionally, the path steps of each path are generated and their frequency in the path list is computed to reflect the occurrences of each path step in the paths list. As mentioned earlier, the computation of these metrics on a large graph becomes computationally expensive therefore preprocessing is done on the dataset to make the computation process practical.

Example : In the example code, there are two unique paths in the RDF graph, one of length 3 and the other of length 2 (see Figure 6.6). The length is computed by counting the number of predicates (relationship) available in a path. The path steps and their strength value are shown in Figure 6.7. It can be seen that the first path step has a strength of 2 as this appears in two paths and the remaining one only has a strength value of 1 as this appears once in both paths. Paths and path steps provide a snapshot of the knowledge in the form of triple patterns that indicate the invariance of instance data or entity description across the data sources that are contextually relevant (domain specific).

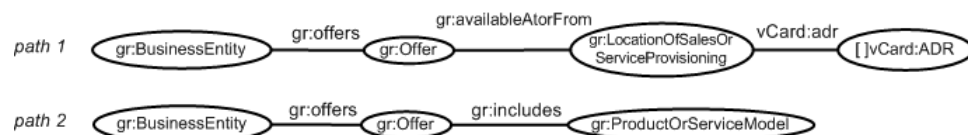


Figure 6.6: Traversal paths

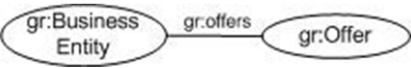
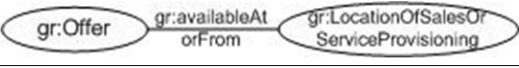


Path step	Strength
	2
	1
	1
	1

Figure 6.7: Path steps and their strength

In the next section, a case study is presented to introduce the domain ontology on which the analysis will be performed.

6.5 Case Study: Empirically Analysing Domain Ontology Usage

One of the domain ontologies identified by the identification phase of the OUSAF framework is the GoodRelations Ontology (GRO). GRO, its schema and key concepts of the ontology are described to introduce the conceptual model represented by the ontology. This ontology will be used in the subsequent section to empirically analyse the use of domain ontologies on the Web.

6.5.1 GoodRelations as a domain ontology

GoodRelations (Hepp, 2008) is one of the first Web ontologies of its kind of, developed and introduced in 2008, to conceptualize the eCommerce domain on the Web. From the outset, GRO has allowed businesses to describe their company (Business Entity), offers and product-related data, based on the RDF data model, over the Web which can be accessed and processed by different Semantic Web applications and search engines. It has recently seen an increase in popularity (See Figure 6.9) and adoption (See Figure 6.8) by the Semantic Web community, particularly after being recognized by the main search engines such as Google (www.google.com), Yahoo (www.yahoo.com) and Bing (www.bing.com). GRO has been successful in selling the idea and value of explicit semantics to these search engines, which have, for a long time, been processing unstructured data to extract fuzzy semantics algorithmically from documents.

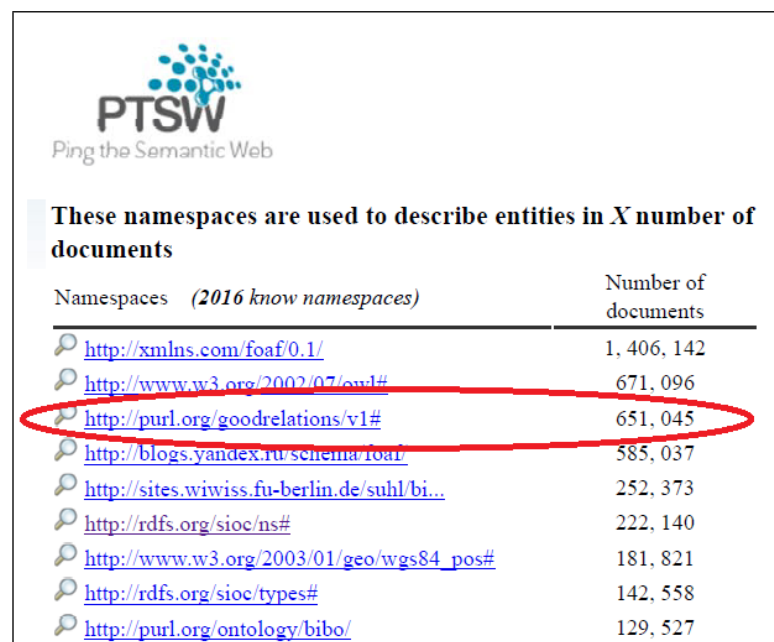
Prominent Users of GoodRelations

GoodRelations is being used by 10,000+ small and large shops world-wide. On this page, we list very prominent users.

Current Users

- **Google** officially recommends GoodRelations for sending structured information for Google Rich Snippets to Google (since 11/2010).
- **Yahoo** officially recommends GoodRelations for sending structured information for their SearchMonkey feature (since 10/2008).
- **Best Buy** is using GoodRelations as fundamental part of their digital marketing strategy and publishes full catalog, store, and special offer with GoodRelations on their production Web sites.
- **O'Reilly** is using GoodRelations for Semantic SEO of all of their book titles.
- **Volkswagen UK** is using GoodRelations for exposing car feature and car component information at massive scale.
- **Renault UK** is using GoodRelations for Semantic SEO for their merchandise shop.
- **OpenLink Software** is using GoodRelations as the fundamental vocabulary for E-Commerce technology based on Virtuoso and other products.
- **Peek & Cloppenburg** is using GoodRelations for publishing information on all European stores plus the brands available in each one of them.
- **CSN Stores** is using GoodRelations for Semantic SEO of all of their 2,000,000 item pages and substores.
- **Arzneimittel.de**, one of Germany's leading mail order pharmacies, is using GoodRelations in RDFa on all of their ca. 250,000 item pages.

Figure 6.8: GoodRelations Ontology Adopters (<http://wiki.goodrelations-vocabulary.org/References>; Accessed 15 Sept, 2012)



PTSW
Ping the Semantic Web

These namespaces are used to describe entities in *X* number of documents

Namespaces (2016 known namespaces)	Number of documents
http://xmlns.com/foaf/0.1/	1, 406, 142
http://www.w3.org/2002/07/owl#	671, 096
http://purl.org/goodrelations/v1#	651, 045
http://blogs.yandex.ru/schema/foaf/	585, 037
http://sites.wiwiss.fu-berlin.de/suhl/bi...	252, 373
http://rdfs.org/sioc/ns#	222, 140
http://www.w3.org/2003/01/geo/wgs84_pos#	181, 821
http://rdfs.org/sioc/types#	142, 558
http://purl.org/ontology/bibo/	129, 527

Figure 6.9: GoodRelations Popularity reported by PingTheSemanticWeb.com (<http://pingthesemanticweb.com/stats/namespaces.php>; Accessed 12 Sept, 2012)

6.5.2 Conceptual Schema and Pivot Concepts

GRO is a kind of live ontology which is evolving with time to capture the changes and improve its conceptual representation of the domain model. The latest version

of the GRO ontology comprises 31 concepts (classes), 50 object properties, 44 data properties and 48 named individuals. Keeping backward compatibility intact, the ontology model is updated frequently to add some new object and data properties, based on the experience and feedback gained through real world implementations. From a high level view, the GR model² is based on three main concepts, each focusing on a separate aspect of the eCommerce domain. These three main concepts are *Business Entity*, *Offering* and *Product or Service* and each is discussed in detail in the following sections. GRO is available at <http://purl.org/goodrelations/v1> and **gr** is the prefix used in this chapter and elsewhere to refer to the vocabulary namespace defined by GRO.

6.5.2.1 Business Entity

The `gr:BusinessEntity` concept represents a business organization (or any individual) which intends to offer or seek products on the Web. The main purpose of this concept is to provide the necessary attributes needed to describe any business, such as the name of the company, address, location, vertical industry in which it operates and any other identifier which makes it uniquely identifiable on the Web. None of the above mentioned properties are mandatory to describe the business entity (company or individual) using GRO, however the more information that is available, the easier it will be to find and consume information with high precision. For large organizations that have multiple outlets or shop locations, GRO provides concepts (`gr:Location` and deprecated `gr:LocationOfSalesOrServiceProvisioning`) to describe shops or service centres through which products or services are provided. Each shop location has its own operation hours which are described using the opening hour specification (`gr:OpeningHoursSpecification`)

6.5.2.2 Offering

`gr:Offering` is the pivotal concept in the GRO. This concept allows the description of a particular offering a business entity is likely to make or seek on the Web. In the latest version, there are 15 data type properties (all optional) available to describe offer details such as availability, validity, name and description of the offering. Recently, name and description have also been added to make it easy to give any name and description to allow users to know more about the offer itself. Offering can include one or more products with a price specification describable in any possible currency.

²<http://www.heppnetz.de/ontologies/goodrelations/goodrelations-UML.png>; (last accessed 25 Sept. 2012)

It is possible to attach supplementary details such as warranty promises, customers who are eligible for the offer, shipment options and charges and acceptable methods of payment.

6.5.2.3 Product or Service

The third main concept is Product or Service (`gr:ProductOrService`). As mentioned earlier, an offering can contain one or more products (or services) and is usually described using one of the three possible subclasses of this main (abstract) class. GROs main focus is to cover the conceptual model of offering rather than being a product ontology. However, `gr:ProductOrService` and its sub-concepts can be used to describe a product and its qualitative and quantitative properties to describe lightweight product ontology.

A description of the implementation of the data collection phase is presented in the next section.

6.6 Data Collection: Hybrid crawler and Dataset

In order to have a clear understanding of the RDF data and the use of ontologies to provide a shared inference and structure on the Web, a dataset comprising domain-specific data extracted from the Web is built to conduct an investigation on empirical grounding. This thesis is particularly interested in data sources which use the domain ontology using core concepts to provide schema level metadata. In the next sub-sections, first, the approach adopted in identifying the potential data sources and the minimum selection criteria used is discussed. Then, the dataset collection approach, including hybrid crawling and the selection of seed URIs followed by the dataset characteristics is described.

6.6.1 Hybrid Crawler

One of the potential sources for the required data is the LOD cloud³ which (as I write) hosts 295 datasets containing approximately 32 billion triples in total. This appears to be a very fertile source of data for our study, however, as reported in (Hitzler and van Harmelen, 2010) and (Bizer et al., 2009), the datasets in the LOD cloud are publishing more data and merely using ontologies, hence, neglecting if not failing in providing

³<http://www4.wiwiw.fu-berlin.de/locloud/state/> (Last accessed on 27 Sept. 2012)

schema level meta-information deemed necessary for information apportioning over the Web. The published LOD statistics also mention that 64.75% of the datasets have made use of non-W3C base-vocabularies (RDF, RDF Schema and OWL) which are called here as open ontologies/vocabularies. Of these open ontologies, 78.31% of datasets use mutually and/or exclusively DC (Dublin Core) (31.19%), FOAF (27.46%) and Simple Knowledge Organization System (SKOS) (19.66%) ontologies to provide schema level information. *Noticeably, only 4 (1.36%) out of 295 are reported to have used GRO and on other hand, PingTheSemanticWeb.com ranks GRO as the third most used ontology after FOAF and OWL* (see Figure 6.9). These numeric facts highlight the scarceness in the use and availability of ontological knowledge in the LOD dataset. Therefore, a dataset was built to collect the RDF data currently published using the domain ontology.

To collate domain-focused data, the minimum criteria employed for the selection of potential data sources is to identify the data publishers which have at least described the key concepts using the domain ontology. In our case, Business Entity and Offering are the primary identification drivers. A list of seed URIs for crawling using Sindice API⁴ and the Watson⁵ semantic search engine (see Figure 6.10) was built. For crawling, an initial attempt was made to use the available semantic crawlers such as LDSpider (Isele et al., 2010) but since most of the eCommerce-related RDF data is embedded within HTML pages using RDFa and due to a lack of interlinking between different resources even within the same hostname, the existing crawler could not be used effectively. Therefore, using LDspider API, a hybrid crawler which crawls in a similar way to traditional Web crawlers by following hyperlinks and extracting only the RDF triples available in Web documents was implemented. Using REST-based Web services, namely Any23⁶ and RDFa Distiller⁷, the extracted RDFa snippets from web documents were then transformed into an RDF/XML document to have one RDF graph for each Web document.

⁴<http://sindice.com/developers/api> (last accessed 21 Sept 2012)

⁵<http://kmi-web05.open.ac.uk/WatsonWUI/> (last accessed 21 Sept 2012)

⁶<http://incubator.apache.org/projects/any23.html> (last accessed 21 Sept 2012)

⁷<http://www.w3.org/2007/08/pyRdfa/> (last accessed 21 Sept 2012)

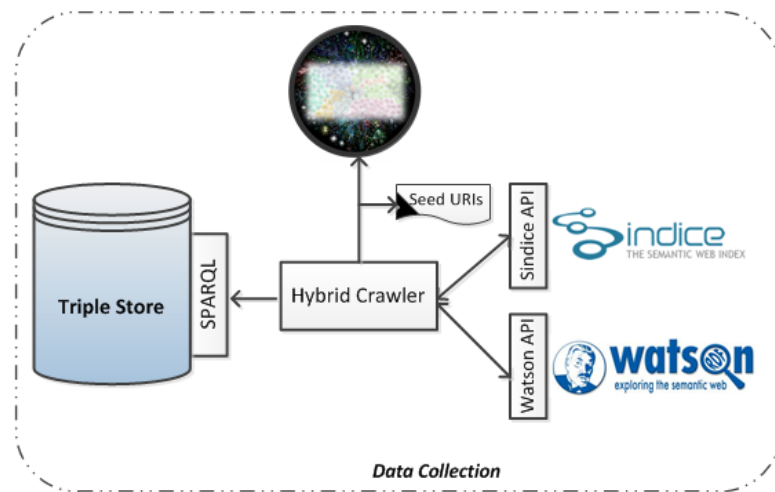


Figure 6.10: Schemata diagram of Hybrid Crawler.

RDF Graphs were then loaded into the OpenLinks Virtuoso⁸ triple store to create the dataset for further analysis known as **GRDS**. From an RDF data management perspective, named graphs (Carroll et al., 2005) are used to group all the triples from one data source (hostname) under a unique named graph International Resource Identifier⁹ (IRI), allowing the dataset to be queried vertically (one data source) and horizontally (across data sources).

6.6.2 Dataset characteristic

The empirical analysis is performed on the GRDS dataset which is built using the hybrid crawler discussed earlier. The GRDS dataset comprises 22.3 million triples (loaded into the open source version of the Virtuoso triple store) collected from 211 different data sources (pay-level domains). The complete list of data sources included in GRDS are shown in Figure 6.11.

⁸<http://virtuoso.openlinksw.com/> (last accessed 21 Sept 2012)

⁹<http://www.w3.org/International/O-URL-and-ident.html> (last accessed 21 Sept 2012)

6.6.3 Data providers landscape

By observing the structured eCommerce data landscape (while building the GRDS), I categorize data publishers into three groups, based on their publishing approach, usage pattern and data volume.

6.6.3.1 Large Size Retailers

This group includes large online eRetailers and retailers who are traditionally premises-based and have only recently entered the eRetailing business. Such data sources provide more detailed (rich) offerings and product descriptions which is useful for entity consolidation and interlinking with other datasets. Such companies include Volkswagen.com.uk, BestBuy.com, Overstock.com, O'Reilly.com, and Suitcase.com, to name a few.

6.6.3.2 Web shops

A large number of semantic eCommerce adopters are small to medium Web shops which offer their products and services mainly through Web channels. Most of these Web shops use Web content management packages¹⁰ such as Magento¹¹, Oxid-eSales¹², WP 4 eCommerce¹³, osCommerce¹⁴ and Joomla Virtuemart¹⁵ to add RDFa data in offer-related Web pages. This approach of embedding Semantic Web data in existing Web pages works well for small and medium Web shops since no special infrastructure arrangement is required in most cases as the semantic metadata (data describing products and offers) is embedded within existing Web documents, hence offering several benefits to both producers and consumers.

6.6.3.3 Data Service providers (Data spaces)

To leverage the benefits offered by semantic eCommerce data, businesses are offering data services that are built on consolidated semantic repositories. Moreover, the providers use APIs to access and transform proprietary data into RDF before making them available through their repositories. For example, Linked Open Commerce

¹⁰Complete list of their references are available at http://www.ebusiness-unibw.org/wiki/GoodRelations#Shop_Software

¹¹www.magentocommerce.com (last accessed 21 July 2012)

¹²www.oxid-esales.com/ (last accessed 19 Mar 2012)

¹³wordpress.org/extend/plugins/wp-e-commerce/ (last accessed 15 July 2012)

¹⁴www.oscommerce.com/ (last accessed 8 July 2012)

¹⁵virtuemart.net/ (last accessed 1 Sept, 2012)

(LOC)¹⁶ contains Amazon.com data although Amazon.com has not yet published RDF, RDFa, transformed using OpenLink Virtuoso Sponger¹⁷.

6.6.4 Use of different Namespace Analysis in GRDS

The availability of different ontologies in the dataset and their usage intensity can be seen by querying the dataset and identifying the data sources using those ontologies. A different approach is adopted in reporting namespaces. Instead of counting the number of triples matching specified criteria, the percentage of the data sources that match the criteria available is reported. This approach provides more unbiased usage analysis, as it disregards the size of the implementer and looks at the number of data sources using it. For example, a large implementer such as BestBuy.com uses a term (e.g. `gr:contains`) to describe its two hundred thousand products and happens to be the only data source using this term in the dataset, hence this will count as only one instance of usage in the dataset. Table 6.2 lists the vocabularies present in the captured dataset along with the percentage of data sources using them.

In total, there are 48 namespaces found in the dataset, 22 being listed in Table 6.2 and the others are excluded from the list. It is found that 12 in-house ontologies with no formal description available, 4 with erroneous URIs and 7 namespaces representing W3C's formal specification such as RDF, RDFS, OWL, etc. The complete list of vocabularies found in the GRDS dataset is presented in (Ashraf, 2011). The first four vocabularies in GRDS (see Table 6.2), next to *gr*, namely *vCard*, *foaf*, *Yahoo* and *dc* are, on average, used by 53% of the data sources to describe the commonly used entities.

¹⁶<http://www.linkedopencommerce.com> (last accessed 8 Oct., 2012)

¹⁷<http://docs.openlinksw.com/virtuoso/virtuososponger.html> (last accessed 5 July 2012)

Table 6.2: List of vocabularies and their percentage in GRDS

Prefix	Namespace	%Data sources
Gr	http://purl.org/goodrelations/v1#	97.16
vCard	http://www.w3.org/2006/vcard/ns#	79.15
foaf	http://xmlns.com/foaf/0.1/	54.98
yahoo	http://search.yahoo.com/searchmonkey/commerce/	41.71
Dc	http://purl.org/dc/terms/	36.49
eCl@ss	http://www.ebusiness-unibw.org/ontologies/eclass/5.1.4/#	18.01
V	http://rdf.data-vocabulary.org	16.59
Og	http://opengraphprotocol.org/schema/	9.00
rev	http://purl.org/stuff/rev#	7.11
pto	http://www.productontology.org/id/	1.90
geo	http://www.w3.org/2003/01/geo/wgs84_pos#	0.95
Cc	http://creativecommons.org/ns#	0.95
frbr	http://vocab.org/frbr/core#	0.47
void	http://rdfs.org/ns/void#	0.47
sioc	http://rdfs.org/sioc/ns#	0.47
vso	http://purl.org/vso/ns#	0.47
coo	http://purl.org/coo/ns#	0.47
scovo	http://purl.org/NET/scovo#	0.47
comm	http://purl.org/commerce#	0.47
media	http://purl.org/media#	0.47

6.7 Empirical Analysis of Domain Ontology Usage

In this section, domain ontology usage is empirically analysed based on the GRDS dataset and the metrics defined in Section 6.4 as part of EMP-AF framework.

The computation of certain metrics defined for the empirical analysis required preprocessing to overcome the computational challenge. Before proceeding with the analysis, in the following section, the preprocessing performed as part of the Empirical Analysis Module is discussed.

6.7.1 Preprocessing

In order to compute the metric values and gather the results of simple measures (computationally less expensive) such as concept instantiations, the presence of certain triple patterns and the use of different properties with a given pivot concept are obtained by posing SPARQL queries to the dataset. However, for computationally complex operations such as traversal path, querying the dataset using the triple stores SPARQL endpoint does not offer a practical solution. Any query with more than three triple patterns in chain with filter clause(s) fails to return the result set in a reasonable time. As a work around, the dataset is exported into N-Triples (a line-delimited syntax

for RDF graphs) format using Jena API (McBride, 2002) and nxparser API¹⁸ is used to extract the paths fanning out from the pivot entity. The list of paths is then used to compute the maximum and average path length. Additionally, the path steps of each path are generated and their frequency in the path list is updated to reflect the occurrences of each path step in the path's list.

To understand the use of label properties by the data publishers, two metrics are used, namely *Entity_{fl}*, *Entity_{dl}* to measure the use of formal label properties and domain-ontology-specific label properties, respectively. Aside from RDFS, several ontologies have defined their own labelling properties which are often used together to provide the same contextual information but using different predicates. Publishers do this to provide support for different vocabularies to make it easy for consumers, however, sometimes it becomes an issue to decide which one to use while querying the data, from the consumers point of view. A few labelling properties which are formally defined as sub-properties (using `rdfs:subPropertyOf`) of `rdfs:label`, make it easy for the application to include all the labels available for an entity, if lightweight reasoning is supported. To make our analysis of labels more empirically grounded, the definition of *Entity_{fl}* was relaxed to also include all the labelling properties which are sub-properties of `rdfs:label` and this includes: `foaf:name`, `skos:prefLabel`, `sioc:name` and `skos:prefLabel`. Another exception/extension has been made to include `dc:title`, even though it is not defined as a sub-property of `rdfs:label`, but since it is one of the largely used (Ell et al., 2011) properties in LOD, it is included under *Entity_{fl}*. After relaxing the conditions, the following is the set of label properties as part of the formal labels:

$$\textit{FormalLabels} = \{ \text{foaf:name}, \text{skos:prefLabel}, \text{sioc:name}, \text{dc:title} \}$$

In order to compute *Entity_{dl}* for a given pivot concept, a set of label attributes defined by the domain ontology is needed where the pivot concept is the `rdfs:domain` of the label property. For the three pivot concepts used in this analysis, the following is the set of domain labels.

$$\textit{DomainLabels}_{\textit{gr:BusinessEntity}} = \{ \text{gr:legalName} \}$$

$$\textit{DomainLabels}_{\textit{gr:Offering}} = \{ \text{gr:condition}, \text{gr:category} \}$$

$$\textit{DomainLabels}_{\textit{gr:ProductOrService}} = \{ \text{gr:category}, \text{gr:color}, \text{gr:condition}, \text{gr:datatypeProductOrServiceProperty} \}$$

¹⁸<http://code.google.com/p/nxparser> (last accessed 14 Aug 2012)

Based on the preprocessing discussed above, the SLG and the usage of each pivot concept using the CUT metrics is analysed.

6.7.2 Analysing the Schema Link Graph (SLG)

Using the Schema Link Graph model, a graph representing all the ontologies available in the dataset was obtained where the links reflect the co-usability of different ontologies. Figure 6.12 shows the links between entities defined across various ontologies. The node size represents the degree of an ontology which means the number of other ontologies linked with the ontology in further describing the entities available in the dataset. For example, the *foaf* node has a degree value of 7 which means that the *foaf* resources are further linked with *dc*, *frbr*, *vso*, *vCard*, *pto*, *gr* and *v* resources. In the Schema Link Graph, the average node degree is 4.12 with a standard deviation 3.61 which shows that the degree distribution ostensibly follows the Power Law distribution (Clauset et al., 2009). However, the average degree distribution in the Schema Link Graph is encouraging as it reflects a good co-usability factor which exists in the dataset. After analysing the use of different vocabularies and the linking of entities over different vocabularies, in the next section, the domain ontology usage is examined in a more detailed fashion to understand the data and knowledge patterns available in the dataset.

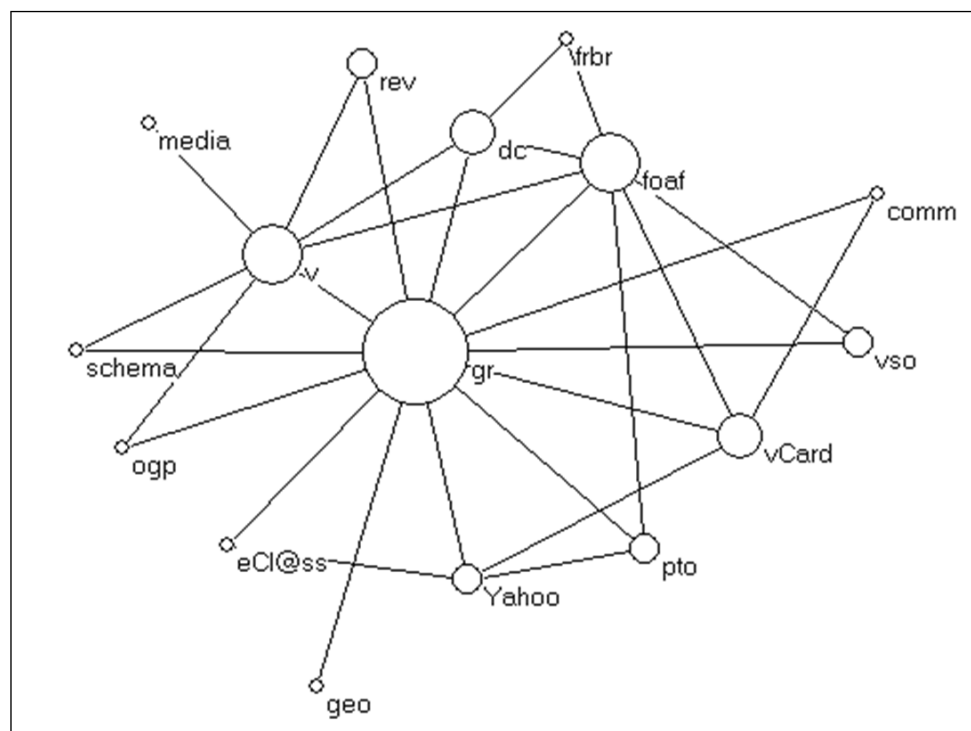


Figure 6.12: Schema Link Graph (SLG) in GRDS

6.7.3 Analysing the Concept Usage Template (CUT) and Labelling

In order to carry out the empirical analysis of the domain ontology, it is important to identify the pivot concepts which represent the core entity in the domain conceptualized by the domain ontology. While there are some advanced approaches (Zhang et al., 2006) available which can be employed to automatically find the key concepts of the domain ontology, the `gr:BusinessEntity`, `gr:Offering` and `gr:ProductOrService` pivotal concepts, introduced in Section 6.5.2 were used.

6.7.3.1 `gr:BusinessEntity` Analysis

In GRO, `gr:BusinessEntity` represents a business organization (or any individual) which intends to offer or seek products on the web. First, the RDF usage based on the CUT is examined and then the available paths and labels provided with the entities of this concept are discussed. Table 6.3 provides the analysis results for the `gr:BusinessEntity` concept. In our dataset, $CI(gr:BusinessEntity)$ i.e. Eq. (6.1) is 789440 entities in total and of these, 54,542 are of the type `gr:BusinessEntity` concept. This means that 6.9% of the entities are of this type in the GRDS. From the *Vocab* (Eq. (6.3) set, the co-usage of different vocabularies in the entity description can be seen. The list of object properties provides the approximation of the relationship entity has and provides substantial evidence about the discoverable related entities in the knowledge base. By looking at the object properties, one can easily see that this pivot business entity is described with its location address and contact-related details. In addition to this relationship, attribute usage provides all the attributes used to provide textual information about the entity. In RDF data, it is presumed that all the resources are identified with URIs which when dereference returns the human readable information about the resource. Interestingly, in attribute usage ($Attri(C)$) found the use of several attributes which are from `schema.org`¹⁹ and are not valid URIs. This also indicates the adoption and use of non-semantic schema in RDF data which is believed to be a good sign as far as the burgeoning of structured data on the Web is concerned, though the semantic aspect is being ignored²⁰.

¹⁹<http://schema.org> (last accessed 1 Oct., 2012)

²⁰On a side note, there has been a community effort in mapping `schema.org` terms with their semantic version published at <http://schema.rdfs.org/mappings.html>

Table 6.3: CUT of gr:BusinessEntity

Entity	gr:BusinessEntity
Instantiation	54,542
Vocabs	vCard, gr, foaf, yahoo, v and schema
Object properties	vCard:adr, vCard:email, vCard:url, yahoo:image, gr:offers, gr:hasPOS, foaf:logo, foaf:homepage, foaf:maker,, foaf:page, gr:hasOpeningHourSpecification, foaf:depiction
Attributes Usage	vCard:fn, vCard:tel, vCard:email, vCard:organization-name, vCard:fax, vCard:adr, vCard:Tel, gr:hasISICv4, gr:legalName, v:name, v:pricerange, v:category, foaf:maker, yahoo:seatingOptions, yahoo:cuisine, yahoo:features, yahoo:smoking, yahoo:serviceOptions, yahoo:mealOptions, yahoo:priceRange, yahoo:hoursOfOperation, schema:postalCode, schema:addressLocality, schema:streetAddress, schema:telephone
Class Usage	vCard:VCard, cVard:org, yahoo:Business, yahoo:Restaurant, gr:BusinessEntityType, comm.:Business, v:Organization
Interlinking	rdfs:seeAlso, owl:sameAs

Class Usage (*ClassUsage(C)*), which lists the other classes of which entity is a member, returns 7 other classes. This tells us that one or more entities of the gr:BusinessEntity class in this dataset also has membership with seven other classes. This membership relationship information, in fact, provides the intrinsic overlapping in the conceptualization of different concepts which have several aspects in common, but not essentially the same interpretation in cross domains. To promote information interoperability on the Web, the identification of related but different concepts in the knowledge base facilitates alignment between different concepts in the ontology mapping process. I believe that related concepts often maintain an elusive relationship, requiring more diverse mapping predicates to capture the natural linkages between disparate concepts instead of using a mapping predicate with strong semantics i.e owl:equivalentClass (Bergman, 2011).

In interlinking, the information related to the linking of similar but disparate entities is captured. This includes the link base and the equivalence links indicating that different URIs are, in fact, referring to the same resource or entity. We found the use of two interlinking properties for the entities in the dataset, namely owl:sameAs and rdfs:seeAlso. The rdfs:seeAlso provides very little information about the resource it links to but is a standard Semantic Web method of linking hypertext to

provide reference to additional resources or documents. The last component of CUT measures the use of label properties in the dataset. As mentioned earlier, $Entity_{fl}$ and $Entity_{dl}$ metrics are used to measure the use of formal label predicates and the domain ontology-specific label properties, respectively. Focusing on this pivot concept, 32% of the entities have used the label properties with the following values for these two metrics:

$$Entity_{fl} = 1,703 \text{ (9\% of entities have used formal labels)}$$
$$Entity_{dl} = 17,146 \text{ (91\% of entities have used domain labels)}$$

One of the most obvious and surprising findings is the dominance of the domain label predicates over the formal labels. Contrary to the previous findings in (Elliott et al., 2011; Manaf et al., 2010) and the general presumption that formal labels are more frequently used, in our experiment, the dominance of domain ontology-specific label properties can be seen. This also signifies that information (data) publishers prefer to provide specialized label properties to help consumers access less ambiguous contextual information, useful for querying and interface presentation.

6.7.3.2 gr:Offering Analysis

gr:Offering is the concept which enables business entities to publish their offers on the Web, either for selling or buying products. Table 6.4 presents the CUT for the gr:Offering pivot concept. In RDF usage, an interesting finding is the use of different but related vocabularies to semantically describe offering-related information. Three vocabularies which supplement offering information, namely *media*, *rev* and *comm* are included, however, two names which are included in the gr:BusinessEntity concept, *vCard* and schema vocabularies have been excluded. In both Object Property (Eq. (6.4)) and Attribute Usage (Eq. (6.5)), similar to the previous concept, the use of different predicates from different vocabularies used to provide the offering description can be seen. Another interesting finding is the use of product vocabularies to describe the products being offered; therefore, the use of different concepts defined in product ontology as part of the Class Usage can also be seen. Since the list was long, this chapter only provides the number of concepts used from the pro-vocabulary. The use of interlinking predicates is the same as the previous pivot concept and one can easily assume that these two predicates are consistent across all key concept and entities.

Next, the use of label properties by the entities of the gr:Offering type are analysed. Of 61330 entities, 11% used labelling properties with the following distribution:

$Entity_{fl} = 4,171$ (62% of entities used formal labels) $Entity_{dl} = 2,610$ (38% of entities used domain labels)

Table 6.4: CUT of gr:Offering

Entity	gr:Offering
Instantiation	61,330
Vocabs	gr, foaf, v, comm, media, rev, yahoo
Object properties	gr:availableAtOrFrom, gr:hasBusinessFunction, gr:eligibleCustomerTypes, gr:acceptedPaymentMethods, gr:availableDeliveryMethods, gr:includesObject, gr:hasPriceSpecification, gr:hasWarrantyPromise, gr:includes, gr:hasManufacturer, gr:hasInventoryLevel, gr:hasBrand, foaf:page, foaf:depiction, foaf:thumbnail, yahoo:media/image, yahoo:product/specification, yahoo:product/manufacture, v:url, v:photo, v:hasReview, media:depiction, media:sample, media:contains, rev:hasReview
Attributes Usage	gr:validFrom, gr:validThrough, gr:eligibleRegions, gr:hasStockKeepingUnit, gr:availabilityStarts, gr:hasEAN_UCC-13, gr:description, gr:name, gr:condition, gr:hasMPN, gr:BusinessEntity, gr:hasCurrency, rdfs:title, rdfs:comments, dc:description, dc:title, dc:contributor, dc:date, dc:type, dc:duration, dc:position, v:name, v:description, v:price, v:category, v:brand, ogp:image, ogp:type, ogp:site_name, ogp:title, ogp:url
Class Usage	v:Product, media: Album, media:Recording note: I have found around 26 product types defined by http://www.productontology.org/ .
Interlinking	rdfs:seeAlso, owl:sameAs

6.7.3.3 gr:ProductOrService Analysis

In GRO, a lightweight description of the products being offered are described through gr:ProductOrService and three of its sub-classes.

Table 6.5 shows the usage summary for the gr:ProductOrService concept. In total, there are roughly 38,000 entities defined as ‘type of product’. Since in GRO, product-related concepts are arranged in a taxonomical hierarchy to allow users to specify the exact nature of the product being offered, the subsumption axiom is used to include all the instances belonging to the super concept. Vocabulary usage for product and offering is almost identical and entities of both concepts use the same vocabularies to describe the instances. One important improvement to Class Usage, compared

with our previous study (Ashraf et al., 2011) is that, now most new eCommerce data publishers use product ontologies to describe their products. For example, in our dataset, more than approximately 100 concepts of *pto* are used to specify the type of products being offered. In interlinking, the usage of `rdfs:seeAlso` predicate is seen, however, there is no usage instance of the `owl:sameAs` predicate. Possible reasons for the (temporary) nonexistence of this predicate in product instances is first, product ontologies have recently begun to emerge but these ontologies do not offer rich product descriptions such as covering the qualitative and quantitative properties of products, and secondly, `owl:sameAs` interlinking is algorithmically complex and less effective and it is preferred to be done through social engagement²¹. Pertaining to the use of label properties with product instances, the label metric values are as follows:

$Entity_{fl} = 30,379$ (99.05% of entities are using formal labels) $Entity_{dl} = 360$ (0.95% of entities are using domain labels)

In the product pivotal concept, 30,739 entities have labels attached to the instances which mean that 80% of the entities offer textual descriptions to provide human readable descriptions of the product. Of these 80%, only 0.95% of the entities provide domain label properties and 99.05% formal labels, which is quite a different trend compared to the above two pivot concepts. As mentioned earlier, GRO provides only high level concepts to identify the product but recommends using product ontologies such as *eCl@ss* and *pto* to provide a semantic description of the products, therefore, there is little or negligible use of domain ontology-specific labels.

²¹In a keynote speech at ISWC2011 (<http://www.cs.vu.nl/~frankh/spool/ISWC2011Keynote/>; last accessed 17 Sept., 2012), Frank van Harmelen mentioned the role of social engagement being more effective than an algorithmic approach in interlinking entities.

Table 6.5: CUT of gr:ProductOrService

Entity	gr:ProductOrService
Instantiation	37,996
Vocabs	gr, foaf, yahoo, v, vso, eCl@ss, pto
Object properties	gr:hasMakeAndModel, gr:hasInventoryLevel, gr:hasManufacturer, gr:description, gr:depth, gr:height, gr:weight, gr:width, vso:mileageFromOdometer, gr:hasBusinessFunction, gr:hasMakeOrModel, gr:hasBrand, gr:hasPriceSpecification, foaf:depiction, foaf:thumbnail, foaf:page, foaf:logo, rev:hasReview, v:hasReview, vso:bodyStyle, vso:engineDisplacement, vso:gearsTotal, vso:previousOwners, gr:name, vso:transmission, vso:fuelType, vso:feature (<i>note: there are several in-house developed ontologies to describe product attributes</i>)
Attributes Usage	gr:description, gr:hasStockKeepingUnit, gr:hasEAN_UCC-13, gr:name, gr:hasMPN, gr:condition, gr:category, vso:modelDate, vso:VIN, vso:color, vso:engineName, vso:rentalUsage
Class Usage	eCl@ss, v:Product, yahoo:Product, vso:Automobile (<i>note: http://www.productontology.org has hundreds of classes which are used in dataset for describing high level product type / category</i>)
Interlinking	rdfs:seeAlso

6.7.4 Analysing Knowledge Patterns (Traversal Path)

Referring to Section 6.4.5, traversal path metrics are defined to understand the available knowledge patterns in the dataset by constructing traversal paths and computing the strength of the path steps in those paths. The number of traversal paths in the dataset, originating from each pivot concept, is presented in Table 6.6.

Table 6.6: Traversal path of all three pivot concepts

	gr:BusinessEntity	gr:Offering	gr:ProductOrService
Number of unique paths	12,245	14,871	2,453
Maximum path length	6	4	3
Average path length	3.12	2.78	2.13

Table 6.6 shows the number of unique paths which exist for each pivot concept. To recap, in traversal paths, all the unique paths originating (fanning out) from the given pivot concept are calculated. This provides the data and schema level patterns available in the knowledge base. Since gr:BusinessEntity is considered a kind of root

(not in the literal sense) concept, therefore it can be seen that it has the largest maximum traversal path length and similarly `gr:ProductOrService` being the later concept in the ontological model, has the lowest maximum length. Interestingly, there is not much significant deviation in the average path length which indicates that even though `gr:BusinessEntity` has the maximum path length on average, all the pivot concepts have a close average path length. Such insight into data and schema patterns and the depth in triple chaining patterns helps in planning data management including storage, querying and reasoning. To further understand the triple patterns available in traversal paths, the following table lists the dominant path steps extracted from the paths with their frequency.

Table 6.7: Path Steps frequency in Traversal Path

Path step	Frequency
<code>gr:Offering gr:hasBusinessFunction gr:BusinessFunction</code>	51928
<code>gr:Offering gr:hasPriceSpecification gr:PriceSpecification</code>	34659
<code>gr:Offering gr:includesObject gr:TypeAndQuantityNode</code>	29038
<code>gr:Offering gr:availableAtOrFrom gr:Location</code>	24914
<code>gr:Offering gr:hasManufacturer gr:BusinessEntity</code>	19430
<code>gr:Offering gr:eligibleCustomerTypes gr:BusinessEntityType</code>	15906
<code>gr:SomeItems gr:hasMakeAndModel gr:ProductOrServiceModel</code>	7168
<code>gr:Offering gr:availableDeliveryMethods gr:DeliveryMethod</code>	5462
<code>gr:Offering gr:hasWarrantyPromise gr:WarrantyPromise</code>	4090
<code>gr:BusinessEntity gr:offers gr:Offering</code>	2398
<code>gr:BusinessEntity vCard:adr vCard:Address</code>	2385
<code>gr:OpeningHoursSpecification gr:hasOpeningHoursDayOfWeek gr:DayOfWeek</code>	1953
<code>gr:Offering gr:includes gr:ProductOrService</code>	1814
<code>gr:Location gr:hasOpeningHoursSpecification gr:OpeningHoursSpecification</code>	1025
<code>gr:BusinessEntity gr:hasPOS gr:Location</code>	598
<code>gr:Offering media:contains v:Product</code>	514
<code>gr:BusinessFunction gr:hasBrand gr:Brand</code>	265
<code>gr:Offering media:contains media:Recording</code>	218
<code>gr:BusinessEntity vCard:url owl:Ontology</code>	182
<code>gr:WarrantyPromise gr:hasWarrantyScope gr:WarrantyScope</code>	19
<code>gr:DayOfWeek gr:hasNext gr:DayOfWeek</code>	7
<code>gr:DayOfWeek gr:hasPrevious gr:DayOfWeek</code>	7
<code>gr:Offering rev:hasReview rev:Review</code>	4

Table 6.7 lists the dominant path steps with the frequency found in traversal paths. This provides a snapshot of the terminological knowledge and the schema level triples available in the dataset. This and the traversal path information, which provides the summary of the knowledge base, helps in generating the SPARQL query template

to access domain-related knowledge from any dataset. However, note that while this provides a complete set of terminologies used in the dataset, not necessarily all entities use these terms, therefore certain terms need to be optional in the automatic query generation process. To support more effecting automatic query generation, based on the summary above, one can consider attaching frequency to each term to have some distribution estimation.

In next section, the empirical analysis obtained using the EMP-AF framework is evaluated using a few of the requirements discussed in the introduction section.

6.8 Empirical Analysis Evaluation

There are different types of users, each of whom may have their own requirements pertaining to the required understanding on the use of ontologies. The aim of empirical analysis is to analyse and obtain a detailed insight into the use of domain ontologies on the Web. In the aspect analysis phase, key aspects which can provide broader visibility into the adoption, uptake and usage of domain ontologies are considered to define the metrics for investigation. The following subsection will analyse how these results help to address a few of the question raised in the introduction section, using the results obtained by employing the developed metrics.

6.8.1 Scenario 1: Application developers need to know how a given ontology is being used.

For (Semantic Web) application developers, it is important to know the nature, structure and volume of data available to them for the application. By using the EMP-AF framework, there are several sub-requirements which can be identified to provide precise information to the developers. These precise requirements are described in the following sub-cases.

6.8.1.1 Case 1 : What terminological knowledge is available for application consumption?

Terminological knowledge which refers to the use of terms (vocabularies) defined by ontologies are important as it provides a representation and description of the entities involved in the given domain. Application developers using this information can prepare generic queries to access the data or prepare the interface based on the available (ontological) conceptual elements. The Concept Usage Template (CUT)

which captures all the terminological knowledge attached to the concept, provides a unified source of information to the developer (as well as to other types of ontology users) to prepare the data access layer, accordingly. For example, Table 6.3 shows how *gr:BusinessEntity* concept is (generally) being used and provides specific details on how many instances of this concept are present (i.e 54,542), what other entities it is connected to and what relationships it uses. As shown in Table 6.3, *vCard:adr*, *vCard:email*, *vCard:url*, *yahoo:image*, *gr:offers*, *gr:hasPOS*, *foaf:logo*, *foaf:homepage*, *foaf:maker*, *foaf:page*, *gr:hasOpeningHourSpecification*, and *foaf:depiction* relationships (object properties) are used to provide relevant details for the instances of the concept.

6.8.1.2 Case 2 : What common data and knowledge patterns are available?

From a data management and processing point of view, it is important to know the different types of patterns being followed in the dataset (or usage in general). Information regarding the patterns helps not only in generating prototypical queries but also assists in strategizing the index for efficient information retrieval and storage. Traversal paths and their frequency identify the presence of different knowledge patterns and their frequency in the dataset. For example, in Table 6.7, it can be seen that the knowledge patterns which dominate the whole dataset (indicating that the majority of data publishers have published this piece of knowledge) is (*gr:Offering* → *gr:hasBusinessFunction* → *gr:BusinessFunction*) and this pattern has 51,928 occurrence in the dataset, whereas on the other extreme side, (*gr:Offering* → *rev:hasReview* → *rev:Review*) patterns have the least occurrence which is 4.

6.8.1.3 Case 3 : How are entities being annotated or textually described?

Information regarding the use of different properties to provide textual description to entities is very helpful for developers (as well as to other users) in different ways. For example, knowing which textual or annotative property is being used helps developers design the user interface where the “human readable” description of the entities is displayed rather than showing the URI which is opaque in describing what an entity is, as this is not reader friendly. Additionally, the information regarding which labelling properties, either of standard vocabularies (such as RDF, RDFS, OWL) or of other vocabularies including domain ontologies (such as DC, FOAF, GR), are being used helps in developing an interface that provides/displays information that is machine accessible but also human readable. In the case of the *gr: BusinessEntity* concept, almost 91% of data publishers have used the **domain labels** to provide a

textual description of the entity and the labelling property used for this concept is `gr:legalName` which provides a human readable name of the business entity.

6.8.2 Scenario 2: Data publishers need to know what is being used to semantically describe domain-specific information.

As mentioned in Section 6.1, it is recommended that data publishers, wherever possible, reuse ontologies instead of developing new terms or ontologies, the reason being that the more an ontology is reused, the more value in terms of perceived utility it has. Therefore, for data publishers, it is desirable to know how a given entity is being described and what ontologies are being used. By using EMP-AF, two such requirements are analysed and presented to the data publishers to provide them with the required insight.

6.8.2.1 Case 1 : How is a company (or business) being described and what attributes are being used?

For any business, it is very important to provide a semantic description of their business to make their products or services discoverable by agents/clients. The best approach is to understand how presently, such information is being published by others and what is the dominant structure prevailing on the Web. The dominant structure provides the template which can then be used for publishing Semantic Web data on the Web. EMP-AF provides CUT to capture such a prevalent structure and assists the data publisher in their publishing process. Table 6.3 provides the prevalent semantic description of `gr:BusinessEntity` (which conceptualizes the concept of a company/business) and can be used by data publishers to describe his/her company.

Attribute usage (the fifth row of Table 6.3) provides a list of datatype properties being used by others, helping data publishers know what attributes and which terms are being used to describe a company. Specific to the case study considered in this chapter, a few of the attributes (for a complete list see Table 6.3) used are : *gr:legalName*, *vCard:fax*, *vCard:adr*, *vCard:Tel*, *schema:postalCode*, *schema:addressLocality*, and *schema:streetAddress*

6.8.2.2 Case 2 : What other entities are a company (entity) linked to?

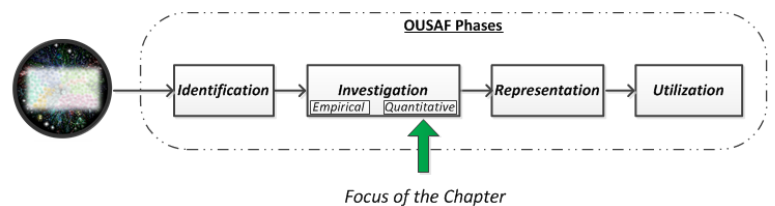
For data publishers, it is important to know how a given entity is being linked with other entities and what relationships are being used. The availability of such information helps data publishers specifically and others generally to know in what dimensions an entity is being described and interlinked with other ontologies. For example, is the company only being described to provide address-related information or does it also describe the company's product-and-service-related information? The CUT metric of EMP-AF provides different sub-metrics to obtain the specific details of concept usage. One of the sub-metrics is *ObjectPro* which captures the relationships the pivot concept holds with other types of resources (entities). Table 6.3 (fourth row) provides a list of relationships (object properties) the business/company type entity holds with other entities. A few of the relationships being used (see Table 6.3 for complete list) are: `gr:offers`, `gr:hasPOS`, `foaf:homepage`, `foaf:maker`, `foaf:page`, `gr:hasOpeningHourSpecification`. It can be seen that other data publishers have provided information pertaining to the branches a company has, offers relating to its products/services and the address of the homepage of the company.

6.9 Conclusion

In this chapter, the EMP-AF framework is presented to perform empirical analysis on the use of domain ontologies on the Web. The developed metrics are used on the dataset to analyse how the domain ontology (GoodRelations, in this case) is being used and how its key concepts are described. The obtained insight helps in addressing the needs and requirements of different types of ontology users in order to make effective and efficient use of the available Semantic Web data.

In the next chapter, which also implements the investigation phase of the OUSAF framework, the use of domain ontologies are quantitatively analysed. The quantitative analysis provides the quantitative measures to help in further realizing the benefits of Ontology Usage Analysis.

Chapter 7 - Investigation Phase: Quantitative Analysis of Domain Ontology Usage (QUA-AF)



7.1 Introduction

In the previous chapter, the EMP-AF was proposed to perform an empirical analysis of domain ontology usage. The empirical analysis, through its observed factors such as the relationship between different ontologies based on an entity's semantic description, ontology component usage, contextual description provision and availability of knowledge pattern, helps to understand the uptake and adoption of domain ontologies on the Web. In other words, it gives a comprehensive analysis of the "usage" aspect of a domain ontology and its components.

While the insights obtained through EMP-AF highlight the key aspects of usage, to fully realize the perceived benefits of Ontology Usage Analysis (OUA), as mentioned in Chapter 4, and in order to undertake a quantitative analysis of OUA, in addition to considering the *usage* dimension, two other dimensions, the "technology" and "business" dimensions, are also important to consider as they have a direct relationship with ontology adoption and usage.

The **technology dimension** captures the technical aspects of the ontology and its components, such as the richness of its structural representation that assists in

the usage of its components by different users. It symbolizes the conceptual model which includes the structural characteristics of ontologies and the formal model which includes the formalization of the conceptualized model. In other words, it considers the design, structural and functional aspects of ontologies to capture its characteristics in the OUA.

The **business dimension** embodies the impetus or commercial advantage (be it monetary or technology) being received directly or indirectly by the end users through the use of ontologies. In other words, it quantifies the incentives available to either the user of the ontologies or to the ontology itself because of its recognition, popularity and dominance. These two dimensions along with the **usage dimension** that provides an insight into the use of domain ontologies in real world settings are important to consider in order to have a comprehensive multi-dimensional insight to ontology usage and their adoption in the real world. Considering these three dimensions together also closely aligns with the "usage model" presented by [Simmons \(2005\)](#) which states that any compelling product is found at the intersection of "business", "usage", and "technology" dimensions, as shown in Figure 7.1. In the context of OUA, ontologies being the engineering artifact are considered as "product" and their usefulness is measured through the three dimensions of "business" being the actual (quantified) value received through the use of ontologies, "usage" being the use of the product in the real world and "technology" being the formal model behind the development of ontologies. In order to analyse domain ontology usage quantitatively, ontologies need to be analysed from the **technology**, **usage**, and **business** dimension. Each dimension has a different aspect of ontology usage analysis to cover which is described as follows:

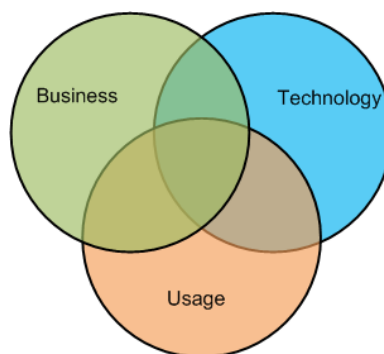


Figure 7.1: Usage Model of [Simmons \(2005\)](#)

1. **Measure the characteristics of an ontology and its components that assist in its usage (technology dimension)** : In order to comprehensively understand how ontologies are being used and what exactly is being used, it is important to understand the characteristics of the conceptual model, its

structure and components. In particular, it is important to measure how different concepts and relationships are defined in an ontology and their semantic description in the ontological model. In other words, the technology dimension measures the richness of ontology components which provides structural insight into how a given ontology is modelled and how the semantics are represented.

2. **Measure the use of an ontology and its components (usage dimension):** This dimension measures the use of ontological components such as concepts, relationships and attributes. This measure helps in understanding how ontologies are being used in real world settings.
3. **Measure the driving factors behind ontology adoption (business dimension):** In order to gain a comprehensive insight into the use of ontologies and their components, it is important to identify and incorporate the driving factors behind the adoption of the ontologies. This dimension measures the benefits that are realised by the users as a result of using an ontology.

To quantify these measures, a mechanism is required to compute and evaluate each dimension in order to undertake a comprehensive analysis of ontology usage. Therefore, in order to quantitatively analyse the use of domain ontologies considering the abovementioned requirements and dimensions, in this chapter, the **QUAntitative Analysis Framework (QUA-AF)** is proposed. The rest of the chapter is organized as follows. Section 7.2 presents the QUA-AF framework and its three phases: the data collection phase, the computation phase and the application phase. It also describes the sequence of the set of activities carried out in the QUA-AF framework. Section 7.3 presents the metrics defined for each dimension to quantitatively analyse domain ontology usage. In Section 7.4, a case study focusing on the domain of eCommerce is presented which will be used in the rest of the chapter to explain the working of the QUA-AF framework and the interpretations of the results obtained from it. In Sections 7.5 and 7.6, GoodRelations and FOAF ontologies (from the case study presented in Section 7.4) are quantitatively analysed using the QUA-AF framework. The evaluation of the framework on the analyzed domain ontologies is discussed in Section 7.7 and Section 7.8 concludes the chapter.

7.2 QUAntitative Analysis Framework (QUA-AF)

The proposed **QUAntitative Analysis Framework (QUA-AF)** comprises three phases: the *data collection phase*, the *computation phase*, and the *application phase*, as shown

in Figure 7.2. The data collection phase is responsible for collecting the required data in the required format from the different sources in order to perform the required analysis of each dimension. In the computation phase, different sets of metrics are defined to analyse the ontology usage in each dimension. In the application phase, the obtained results are then converted into actionable information.

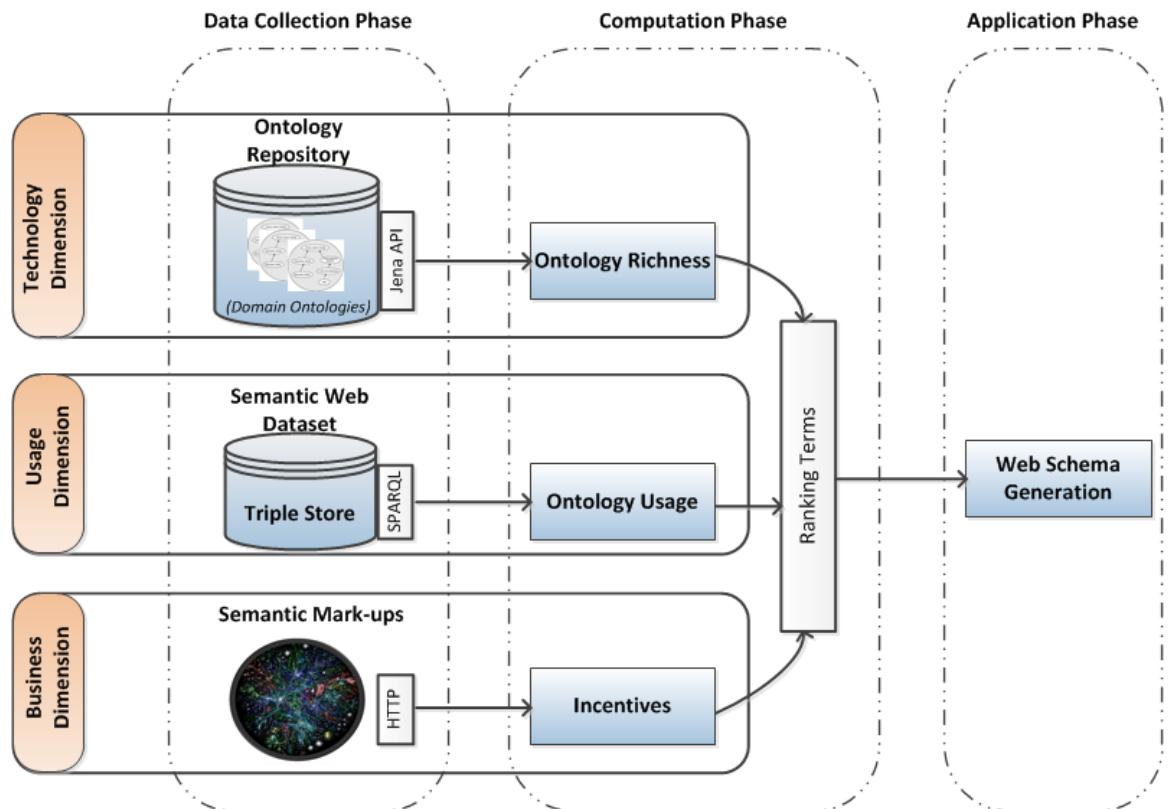


Figure 7.2: Quantitative Analysis Framework for Ontology Usage Analysis

In the next subsection, the objective, the technical aspects and the working of each phase is presented in detail.

7.2.1 Data Collection phase

As mentioned earlier, the data collection phase is responsible for collecting the data required by the QUA-AF framework. Each dimension which needs to be considered in quantitative analysis requires a different type of information to measure the aspects involved in it. This is achieved by having different types of repositories to provide the dimension-specific data for computation purposes as follows:

7.2.1.1 Ontology Repository

The ontology repository collects the data to perform the analysis related to the *technology dimension*. Since the technology dimension captures and quantifies the design and structural characteristics of ontologies which assist in its adoption, the ontology repository hosts (stores) the authoritative representation of domain ontologies. The authoritative representation of an ontology includes ontology documentation, ontology formal conceptualization and metadata about the ontology. Generally, the main sources of such information are ontology libraries (Ding and Fensel, 2001) which maintain the databases of different ontologies.

However, the use of existing ontology libraries for the QUA-AF framework raises two issues. First, existing ontology libraries (like OntoServer (Volz, 2001), ONION (Gangemi et al., 1999a), and Cupboard (d'Aquin and Lewen, 2009)), even though they are complete systems, are computationally expensive considering the need in hand as they require various pre-processing operations such as bootstrapping, meta-data entry, etc., therefore considering them becomes a complex and configuration-extensive choice, leading to an increase in complexity. Second, most online ontology libraries are application-specific (such as OBO Foundry (Smith et al., 2007) and contain biological and biomedical domain-specific ontologies) and are therefore limited in offering ontologies from diverse domains which make them less applicable for our case. Therefore, in order to avoid such drawbacks, for the QUA-AF framework, a local repository of different domain ontologies is maintained.

7.2.1.2 Semantic Web (RDF) data

Semantic Web Dataset collects the data for analyses related to the *usage dimension*. In order to measure the use of an ontology and its components in a real world setting, RDF data is crawled from the Web, comprising published structured data described using semantic markups. The required dataset which comprises real instance data annotated using domain ontologies is crawled and maintained in triple stores to obtain the Semantic Web (RDF) data published on the Web.

7.2.1.3 Semantic Markup Repository

The Semantic Markup repository collects the data for analyses related to the *business dimension*. In order to identify the impetus which encourages data publishers (users) to publish semantically annotated structured data on the Web, a repository is needed to maintain the list of semantic markups supported by different search engines which assist them in the identification and classification of information. The Semantic

Markup repository needs to list all the terms being used which are recognized or supported by search engines either while crawling the data or being used as canonical terms to describe entities. This data is then used to measure the incentives of different vocabularies.

7.2.2 Computation Phase

The computation phase (See Figure 7.2) of the QUA-AF framework focuses on performing quantitative analyses of domain ontology usage by computing different measurements for each dimension. This phase comprises three modules, each focusing on a dimension as described in the following subsections.

7.2.2.1 Ontology Richness Module

The ontology richness module determines the analysis related to the technology dimension of ontologies. In this module, the richness of ontology components such as concepts and relationships are measured and quantified to represent the technology dimension. This module accesses the ontology's authoritative documentation stored in the ontology repository to measure its typological and structural characteristics. For the computation of such information (conceptual model richness), Jena API (Carrol and McBride, 2001) is used to access the ontologies and construct the graph model to measure different properties. Metrics are defined to measure the **concept richness**, **relationship richness**, and **attribute richness**. The metrics defined for the ontology richness module are described in detail in Section 7.3.1.

7.2.2.2 Ontology Usage Module

The ontology usage module determines the analysis related to the usage dimension of ontologies. It measures how a domain ontology and its components are being used in a real world setting. While measuring the use of different ontology components, it needs to consider the axioms available in the ontology to entail the implied usage of the terms defined in the ontologies. For the computation of usage, Semantic Web data comprising real world data published on the Web, annotated using domain ontologies is used. Using Semantic Web data, this module defines metrics to measure **concept usage**, **relationship usage**, and **attribute usage**. The metrics defined for the ontology usage module are described in detail in Section 7.3.2.

7.2.2.3 Incentive Module

The incentive module determines the analysis related to the business dimension. This module captures the commercial advantages available to Semantic Web data publishers. It attempts to recognize the use of different semantic markups by the data consuming application (search engines, for example) and their (name/string) matching with the terminological knowledge available in the ontologies to consider this as the motivational factor behind their adoption. It evaluates the available support for different ontologies by different search engines (or other applications such as RDF triple store, semantic reasoner) and give weightage to those terms accordingly. In order to evaluate the support available in different search engines, manual effort is required to prepare the list of terms being supported by the engine.

As mentioned earlier, the business dimension refers to the commercial incentives or advantages being received by the users through the use of ontologies. But, as mentioned in Chapter 1, as this is still the early stage of Semantic Web technology usage and adoption, it is hard to quantify the exact commercial benefits due to the lack of any study or statistics in this regard. However, I consider it a key factor in fostering the growth and adoption of vocabularies and view it as one of the “driving factors” for early adoption in our study. Two of the other driving factors are the incentives available to structured data publishers and the support available for an ontology/vocabulary in Semantic Web applications and tools.

Using the Semantic markup list, this module defines the **incentive** metric to measure the available commercial incentive for domain ontologies.

The metric defined for the Incentive module is formally described in Section 7.3.3.

7.2.2.4 Ranking different measures

Once the analysis of each module is completed, the results are combined to obtain a consolidated value of ontology usage. Each dimension in QUA-AF contains different metrics and involves different aspects of the ontology, therefore in order to obtain a unified observation of usage, the analysis output of each dimension is weighted according to its preference to generate a consolidated value. The final usage values are then ranked to obtain an ordering list, based on the users requirements.

The ranking approach used in the QUA-AF framework is formally described in Section 7.3.4

7.2.3 Application Phase

The application phase of the QUA-AF framework implements a use case to represent the obtained result. The use case scenario highlights the need for a consolidated Web Schema representing the information for a particular application area. The Web Schema that is generated is based on ontology usage analysis to capture the prevalent and prominent data usage patterns which can be then used by other data publishers. Therefore, based on the identified requirements of the use case scenario, the QUA-AF framework constructs the Web Schema and captures the terminological knowledge representing the information specific to given application area.

In the next subsection, the set of sequential activities carried out by the QUA-AF framework is presented.

7.2.4 Sequence of QUA-AF activities

As mentioned in Section 7.2, the QUA-AF framework comprises three phases: the data collection phase, the computation phase, and the application phase. Each phase involves a certain number of activities to carry out the required functionality and operation. The set of activities and their sequence followed in the QUA-AF framework is depicted in Figure 7.3 and is described below.

- In the data collection phase, for each dimension, the following activities are performed.
 - To measure the technology dimension of ontologies, an ontology repository is built to store the domain ontologies along with their authoritative documentation.
 - To measure the usage dimension of ontologies, a Semantic Web dataset is built to store the RDF data published on the Web. The dataset is also refreshed with new crawled data.
 - To measure the business dimension of ontologies, a list of Semantic Markups supported by different search engines is maintained.
 - In the computation phase, the following activities are performed to measure the aspects of each dimension.
 - To measure ontology usage from the technical dimension, the ontology richness module is defined. The module contains the following metrics:
-

-
- * Concept richness to measure the structural and typological characteristics of concepts.
 - * Relationship richness to measure the structural and typological characteristics of object properties (relationships).
 - * Attribute richness to measure the structural characteristic of datatype properties (attributes).
- To measure ontology usage from the usage dimension, the Ontology Usage module is defined. The module contains the following metrics:
- * Concept usage metric to measure the use of the concept.
 - * Relationship usage metric to measure the use of relationships.
 - * Attribute usage metric to measure the use of datatype properties (attribute) .
- To measure ontology usage from the business dimension, the incentive module is defined. The incentive module defines the incentive metric to measure the commercial incentives available to the user as a result of using the ontology.
- Measures obtained in each module are consolidated using a weight factor to rank the ontologies and their components.
- In the application phase, the obtained quantified analysis is used to construct the Web Schema.
-

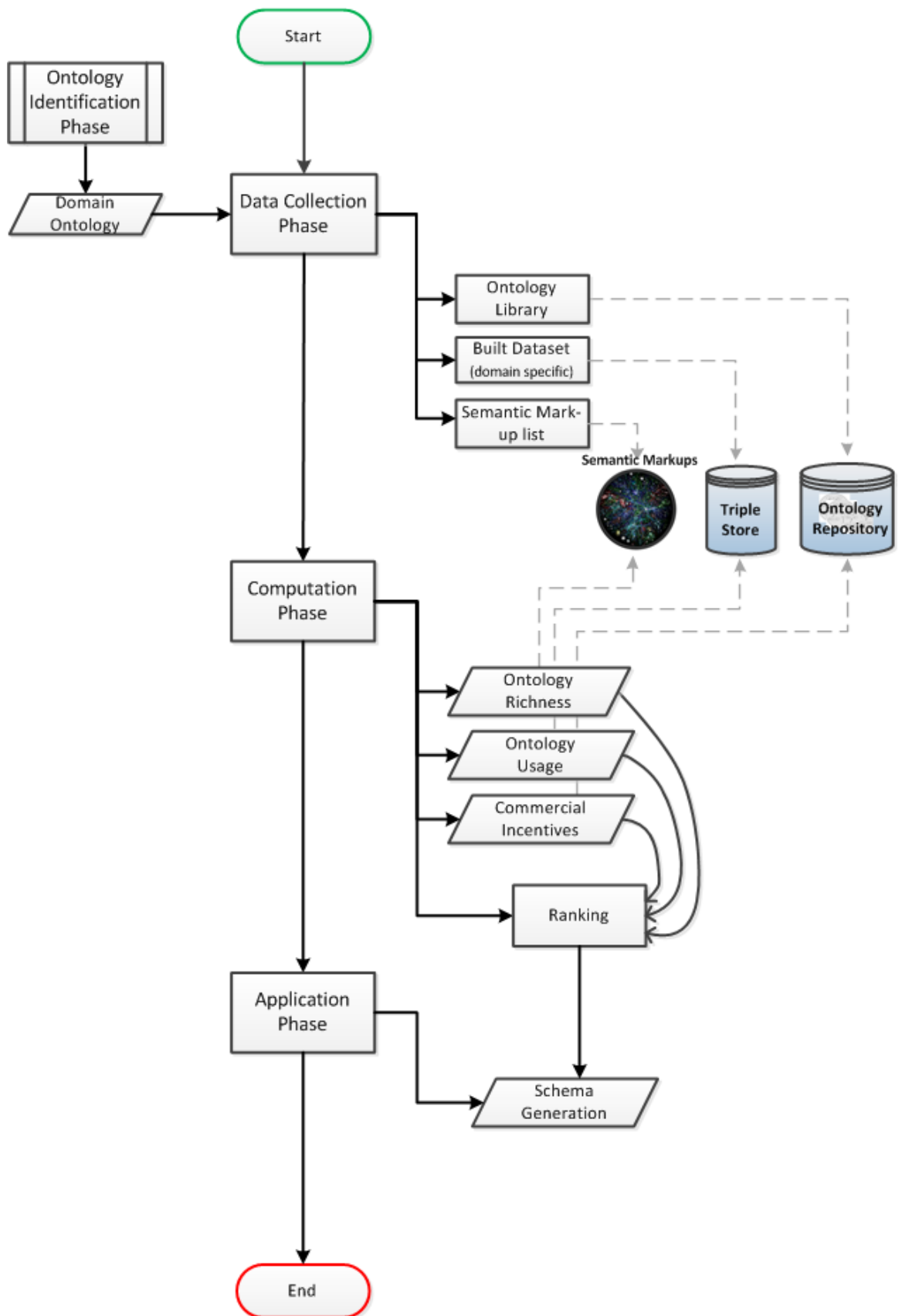


Figure 7.3: Flow of activities in QUA-AF

7.3 Metrics for quantifying dimensions for OUA in QUA-AF

In this section, the metrics to measure each dimension required for the quantitative analysis of domain ontology usage are defined. In order to quantify ontology usage from the three dimensions, a set of metrics is defined for ontology richness, ontology usage and incentive measurements. The metrics defined to measure the ontology from different dimensions are explained using a sample ontology and its instantiation, as depicted in Figure 7.4. The following namespaces are used in the example code to explain the metrics:

- **so** is the namespace for the sample ontology. Sample ontology components are referred to using the **so** namespace, such as `so:Student`
- **ex** is the prefix used to refer to the namespace for instance data such as `ex:jam_ashraf`
- For W3C-based vocabularies, standard namespaces are used such as **rdf**, **rdfs**, and **owl**.

7.3.1 Measuring Ontology Richness

Measuring the richness of ontological terms quantifies the importance of the terms within the ontological model. Ontological terms comprise ontology components such as concepts, object properties (relationships) and data properties (attributes). In the case of RDFS vocabularies, since object and data properties are not disjoint, this thesis only considers the object property to refer to the predicates defined by the vocabulary. The richness of an ontology is measured by the metrics *concept richness (CR)*, *relationship value (RV)*, and *attribute value (AV)*.

7.3.1.1 Concept Richness (CR)

Concept richness (CR) defines the structural richness of a concept. When considering a specific concept in an ontology, one needs to consider the relationship it has with other concepts and the number of attributes available to describe its instances. This includes the typed binary relationship (non-hierarchical) with other concepts and data properties providing attribute values for the data description of the concept. Formally, the concept richness of a particular concept $CR(C)$ of a given domain ontology is

calculated by adding the number of non-hierarchical relationships and attributes that it has.

$$CR(C) = |P_C| + |A_C| \quad (7.1)$$

where

P_C is the number of object, properties (relationship) that Concept C has, and

A_C is the number of datatype properties (attributes) that Concept C has.

$CR(C)$ of a concept reflects its possible contribution in providing a formal structure to represent the specific view of the domain, conceptualized by the concept. In other words, the higher the concept richness value of a concept, the richer the concept is in terms of its description. P_C returns the number of object properties that concept C has, while A_C returns the number of data properties of concept C . The value of $CR(C)$ is a positive integer including zero.

To explain with an example, consider the sample ontology shown in Figure 7.4. Let C be the *so:Student* concept (i.e $CR(so:Student)$) and compute the Concept Richness $CR(C)$. The values for P_C and A_C are as follows:

$P_{so:Student} = 3$ because the student concept has three object properties: *so:play*, *so:livesAt* and *so:StudiesIn*.

$A_{so:Student} = 4$ because the student concept has four attributes defined for it: *so:lastName*, *so:firstName*, *so:gender* and *so:DOB*.

Therefore,

$$CR(so:Student) = P_{so:Student} + A_{so:Student} = 3 + 4 = 7$$

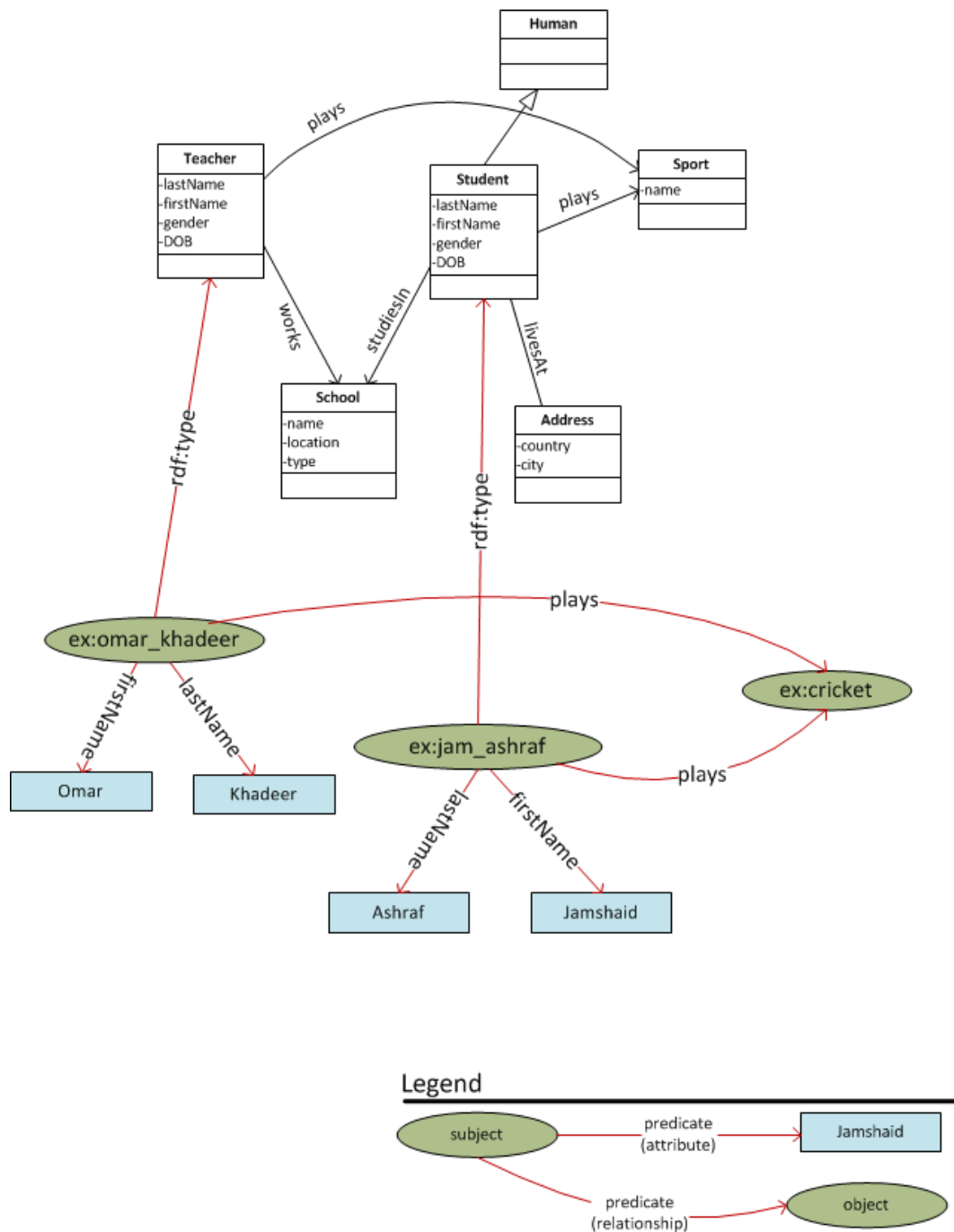


Figure 7.4: Sample Ontology and its instantiation to explain the metrics defined in QUA-AF to measure richness and usage.

7.3.1.2 Relationship Value (RV)

The relationship value reflects the possible role of the object property in creating the typed relationships between different concepts. The object property links the instances of the concepts defined as the domain of this property with the instances

of the concepts defined as the range of the property. RV is computed as follows:

$$RV(P) = |dom(P)| + |range(P)| \quad (7.2)$$

where

$dom(P)$ is the number of concepts property P has as its domain (*rdfs:domain*), and $range(P)$ is the number of concepts (property) P has as its range (*rdfs:range*).

$RV(P)$ returns an integer, reflecting the number of concepts in which the property can be used to create relationships and provide a rich description of a concept. A property with a higher RV reflects its generalization as more concepts (i.e instances of the concepts) can be linked through this property. On the other hand, a lower RV value conveys property specificity. Here, a simplified approach is employed to compute relationship richness (RV), since in OWL, the domain and range are taken as axioms and not as type constraints, thus this could have potentially far reaching effects (Rector et al., 2004). Therefore, I only refer to the authoritative description¹ of the ontology document to compute RV , rather than employing the OWL/RDFS model interpretation for domain and range constraints.

Referring to the sample ontology shown in Figure 7.4, let P be the *plays* relationship. For the computation of relationship value $RV(P)$, compute $dom(so : plays)$ and $range(so : play)$ as below:

$$dom(so : plays) = 2;$$

because the relationship *so:plays* has two concepts as its domain: *so:Teacher* and *so:Student*.

$$range(so : plays) = 1;$$

because the relationship *plays* has one concepts as its range: *so:Sport*

$$\text{Therefore, } RV(so : plays) = |dom(so : plays)| + |range(so : plays)| = 2 + 1 = 3$$

¹The authoritative description of an ontology is the formal ontology document available at the ontology namespace URI (Cheng and Qu, 2008)

7.3.1.3 Attribute Value (AV)

Attributes of a concept are the data properties used to provide literal (typed or untyped) values to the concept instances. AV reflects the number of concepts that have this data property.

$$AV(A) = |dom(A)| \quad (7.3)$$

Datatype properties are very useful and help in providing concrete values to describe the concept's instances (individuals). AV returns a zero or a positive number, reflecting the number of different concepts using it to semantically describe the concept instances. In the case of RDFS vocabularies (since RDFS does not differentiate between object and datatype properties), I only use the RV metric to measure the property value. Referring to the sample ontology shown in Figure 7.4, let A be the *lastName* attribute. For the computation of Attribute Value $AV(A)$, compute $dom(so:lastName)$ as below:

$$dom(so:lastName) = 2;$$

because the attribute *so:lastName* has two concepts as its domain: *so:Teacher* and *so:Student*.

Therefore,

$$AV(so:lastName) = |dom(so:lastName)| = 2$$

7.3.2 Measuring Ontology Usage

To analyse and quantify the use of ontologies on the Web, the following metrics are defined to incorporate the usage aspect in our framework while analyzing its adoption and uptake by publishers. Usage provides an indication of the available instantiation which eventually generates a network effect.

7.3.2.1 Concept Usage (CU)

The concept usage metric measures the instantiation of the concept in the knowledge base (KB). Here, instantiation means the number of unique URI references used to create members of the class represented by the concept. In the RDF graph, I refer to the triples in which the *rdf:type* predicate is used to create members of a given concept.

The concept usage metric is formalized as follows:

$$CU(C) = |\{t = (s, p, o) | p = rdf : type, o = C\}| \quad (7.4)$$

where

t is the triple in the dataset describing an individual (instance) of type Concept C .

$CU(C)$ returns an integer (including a value of zero) and helps in measuring the usage of each concept in the knowledge base (dataset) and ranks them based on their instantiation. But, as mentioned in Chapter 1, one of the features of the ontologies is the provision of inference which means using the ontology to either classify the data or infer (deduce) new implicit knowledge. This means that by using the axiomatic triples available in the ontology and the reasoning service², additional statements can be accessed which are not explicitly asserted in the knowledge base. This has a direct consequence on the concept usage metrics calculation since $CU(C)$ considers only explicit statements in the dataset. Therefore, an extension to the concept usage metric is proposed to include the subsumption (Shadbolt et al., 2006) aspect of reasoning by using the entail function.

$$CU_H(C) = |\{t = (s, p, o) | p = rdf : type, o entail_{rdf9}(C)\}| \quad (7.5)$$

where

$entail_{rdf9}(C)$ is a function which implements a reasoning engine based on RDFS (Hayes, 2004) and OWL-DL (McGuinness and van Harmelen, 2004) entailment rules. In the concept usage metric, the following RDFS entailment rule $rdf9$ is applied:

```
rdf9 :IF(uuu rdfs:subClassOf xxx
      AND vvv rdf:type uuu)
      THEN (vvv rdf:type xxx)
```

This RDF entailment rule will allow top level concepts (super concepts) to subsume the instances of their subclasses.

²Almost all open source and commercial data stores (triple stores) provide RDFS entailment support, such as Virtuoso (www.openlinksw.com) and Stardog (stardog.com)

Referring to the sample ontology and its instantiation shown in Figure 7.4, let C be the concept *Student*. The value of metric $CU(so : Student)$ is 1 as there is only one instance of type Student present in the Figure 7.4. The triple is as follows:

```
<ex:jam_ashraf> <rdf:type> <so:Student>
```

RDF data therefore has a usage value of only 1.

7.3.2.2 Relationship Usage (RU)

The relationship usage metrics calculates the number of triples in a dataset in which the object property is used to create relationships between different concept instances. From the RDF Graph, the relationship usage value is determined by:

$$RU(P) = |\{t := (s, p, o) | p = P\}| \quad (7.6)$$

where

t represents the triples in the dataset having p as their predicate.

The result of RU is a positive integer (including a value of zero). RU is helpful in indexing the properties in combination with RV to support efficient information retrieval. It is also helpful in developing knowledge base applications where relevant data is automatically retrieved and presented, based on the available data space. Referring to the sample ontology and instance data shown in Figure 7.4, let p be the relationship *plays*. There are two triples in the Figure 7.4 which have *plays* relationship (predicates). These triples are:

```
<ex:jam_ashraf> <so:plays> <ex:cricket>
<ex:omar_kadeer> <so:plays> <ex:cricket>
```

Thus, $RU(so : plays)$ is computed as follows:

$$RU(so : plays) = 2$$

7.3.2.3 Attribute Usage (AU)

The attribute usage metric measures how much data description is available in KB (dataset) for a concept instance. From an RDF graph, AU is calculated as:

$$AU(A) = |\{t := (s, p, o) | p \in A, o \in L\}| \quad (7.7)$$

where

t is the triple specifying the attribute value for s .

o in the triple is the datatype property defined in the ontology to provide factual information about the resource being described.

Referring to the sample ontology shown in Figure 7.4, let A be the *so:firstName* attribute. There are two triples in which *so:firstName* is used to describe the resources. These two triples are :

```
<ex:jam_ashraf> <so:firstName> "Jamshaid"
<ex:omar_kadeer> <so:firstName> "Omar"
```

So, the $AU(so:firstName)$ is 2.

7.3.3 Measuring Incentive

The incentive metric measures the benefits to the user as a result of using an ontology. The user can be either a data publisher or a data consumer. It hypothesizes the commercial benefits (driving factor) or immediate advantages available in the marketplace (i.e. Semantic Web dataspace) to the users as a result of using the ontology. As mentioned in Section 7.2, in the absence of any statistical data regarding the commercial benefits or advantages available to early adopters of vocabularies in annotating Web information, to quantify the incentive, heuristics based on empirical findings are applied.

In the QUA-AF framework, only data publishers are used when measuring the incentive metric. From the data publishers point of view, the incentive metric determines what benefits the user will obtain as a result of publishing data that is semantically annotated using a particular ontology. For measuring the incentives for data publishers in the QUA-AF framework, the immediate benefits available to them by search engines in indexing the vocabularies which they use to publish the information are analysed. Efforts have been made by several search engines to provide

a more powerful search experience for users by including semantically marked data in search results, for example, Yahoo! introduced SearchMonkey in 2008 and a year later, Google announced Rich Snippet.

In the QUA-AF, in order to capture such an initiative to measure the incentive for using a particular vocabulary/ontology, the direct benefits available to data publishers by search engines in indexing the vocabulary which they use to publish their information are analysed.

Definition (Incentive of term). Let $S = \{S_1, S_2 \dots S_n\}$ be the set of search engines³ which implements the support of a given v (ontology/vocabulary) in their search results.

The incentive value for using an RDF term⁴ of an ontology O is calculated as:

$$Incentive_{term} = \frac{1}{n} \sum_{j=1}^n w_j * s_j \quad (7.8)$$

where,

$$\left\{ \begin{array}{l} term, \quad \text{the components of ontology} \\ S_j = 1, \quad \text{if term is supported by search engine, otherwise 0} \\ \sum_{j=1}^n w_j = 1 \\ n \quad \text{number of search engines} \end{array} \right.$$

where,

w_j is a weight factor (i.e $W_j \in [0, 1]$) and the sum of the weight cannot be more than 1 to incorporate the relative importance of various search engines, based on their ranking approach and also country coverage.

$Incentive_{term} \in [0, 1]$ of a term is a measure of how incentivized the term (concept, property or attribute) is among all the search engines on average. For example, consider the Yahoo and Google indexing service and $t = foaf:surname$ as the term in focus. If it is assumed that term t is recognized by Yahoo but not by Google and each of them is given a weight of 0.5, then the incentive value will be $Incentive_i = 0.5$. The incentive measure can be formulated to include different aspects such as the number

³Here only traditional search engines which primarily index non-structured information are considered.

⁴Here *RDF term* refers to the terminological knowledge of an ontology comprising concepts, object properties and datatype properties.

of tools providing building support for the ontology as a whole or for a certain set of terms of the ontology. I strongly believe that such an incentive serves as a motivating factor for early adopters and helps in bootstrapping the Web of Data on the Web.

7.3.4 Ranking based on Usage Analysis

The objective of usage analysis is to identify the most highly used terms based on their richness, usage, and available incentives. To rank the terms based on empirical data, the ranks of a given term t of an ontology O are calculated by aggregating the richness, usage and incentive measures, using their respective metrics. To offer preferential aspects to the ranking, weights are used to adjust the priority of each measure accordingly. To have a consistent representation of each metrics, the measure to generate the value in the range of [0,1] is normalised.

The overall rank value of each term is computed as follows:

$$Rank_{t \in O} = W_R Richness_t + W_U Usage_t + W_I Incentive_t \quad (7.9)$$

where

if t is concept $c \in C$ then

$$Richness_{c=t} = \frac{CR(c)}{\max(CR(c))}$$

$$Usage_{c=t} = \frac{(CU_H(c))}{(\max(CU_H(c)))}$$

if t is relationship $p \in P$ then

$$Richness_{p=t} = \frac{RV(p)}{\max(RV(p))}$$

$$Usage_{p=t} = \frac{(RU_H(p))}{(\max(RU_H(p)))}$$

if t is attribute $a \in A$ then

$$Richness_{a=t} = \frac{AV(a)}{\max(AV(a))}$$

$$Usage_{a=t} = \frac{(AU_H(a))}{(\max(AU_H(a)))}$$

and

W_R, W_U and W_I are the corresponding weights of each measure and are adjusted accordingly to the required priority.

Eq.7.9 computes the rank of the terms of the ontology and provides detail on how different measures are computed.

In the next section, a case study is presented which will explain the working of the QUA-AF framework

7.4 Case study: Quantitative Analysis of Domain Ontology Usage

Any typical eCommerce store (website) has numerous web pages pertaining to the various products they offer (product catalogue), promotions (deals) they make, policy-related information, press releases, terms and conditions, warranties, and testimonies from their customers. In order to improve content accessibility, interoperability and the visibility of their products to a wider group of potential customers (which can be both human and machine agents), the eCommerce store owner would like to annotate the web content using existing ontologies to offer a better means of information dissemination.

To decide which ontologies to use and what component in those ontologies to use, the eCommerce store owners should:

- identify the ontologies / vocabularies which have some uptake and adoption
- understand their instantiation for a better network effect
- obtain immediate and tangible benefits for semantic annotation offered by search engines

To achieve the abovementioned requirements, certain tasks which need to be performed in order to understand the importance of ontology usage analysis and its role in implementing such use cases are identified. First, to annotate (semantically describe information) the data, one needs to identify the Web ontologies available for use on the Web. Second, after identifying the available ontologies, one has to understand usage and adoption to identify the suitable terms in existing ontologies. The identification of suitable or highly used terms (concepts, properties/attributes) promotes the reusability of existing terms which maximizes the portability of data by consuming applications ([Heath and Bizer, 2011](#)).

In this case study, for succinctness, only those ontologies being used in a domain-focused corpus which is collected by crawling the eCommerce data sources matching the scenario presented in the above use case are identified.

7.4.1 Dataset and Ontology Identification

To conduct an empirical study on the RDF data and analyse the vocabulary usage in a specific (focused) domain, a dataset is built to serve as a representative sample of the Web of Data currently published on the Web. The collected dataset is sufficiently representative to provide a snapshot of actual domain-specific semantic data patterns, enabling meaningful measurements to be made to enhance our understanding of how data is really being used. Using the hybrid crawler and a seed set comprising of 259 web domains (web sites), a corpus comprising Semantic Web data is built, as discussed in Section 6.6.

Table 7.1: List of ontologies identified in the dataset

Prefix	Ontology URL
foaf	http://xmlns.com/foaf/0.1/
gr	http://purl.org/goodrelations/v1#
v	http://rdf.data-vocabulary.org/#
dc	http://purl.org/dc/terms/
og	http://opengraphprotocol.org/schema/
rev	http://purl.org/stuff/rev#
vCard	http://www.w3.org/2006/vcard/ns#
virt	http://www.openlinksw.com/schemas/virttrdf
comm	http://purl.org/commerce#
frbr	http://vocab.org/frbr/core#
vso	http://purl.org/vso/ns#
pto	http://www.productontology.org/id

Table 7.1 lists the ontologies found in the dataset used by the data publishers to semantically describe eCommerce-related data. Table 7.1 list only the ontologies for which the authoritative ontology description document from the specified ontology namespace URI were found on the Web. There were some ontologies for which the authoritative description document were not found and hence were discarded in this case study. The retrieved ontology documents are stored in the Ontology Library repository to be used by the QUA-AF framework to perform ontology usage analysis.

In the next section, the developed metrics for the QUA-AF framework are applied to analyse the usage of different ontologies identified in the dataset. For brevity, the analysis is limited to two largely-used ontologies, GoodRelations and FOAF, in the dataset, to provide a detailed discussion on the findings.

7.5 GoodRelations Ontology Usage Analysis

The GoodRelations (GR) ontology (Hepp, 2008) is an open Web ontology, developed for the eCommerce domain which allows businesses (and individuals) to describe their *offering*, *business*, and *products* on the Web. In the latest version of the authoritative document⁵, the ontology comprises 32 concepts, 49 object properties and 46 data properties, including a few deprecated terms. From a high level view, the GR model is based on three main concepts, each focusing on a separate aspect of the eCommerce domain. These three concepts include:

- business entity to represent the business organization selling or seeking products;
- offering to represent offers with details of the price; and
- product or service to conceptually describe the product included in the offer made by the business entity.

The QUA-AF framework is applied on the GR ontology to analyse ontology usage by measuring the concept richness, usage and incentives, as shown in Table 7.2. The table displays the concepts in the order of their final rank value which is computed using Eq. 7.9.

For the computation of incentives, Eq.7.8 is used after deciding on the S set. In this chapter, three search engines, Google, Yahoo and Bing i.e. $S = \{google, yahoo, bing\}$ are considered the sources which recognize structured data on web pages and particularly meta-data using Web ontologies. For example, Google publishes⁶ the list of GoodRelations terms it supports by recognizing their presence in web pages and uses this in the ranking and searching process. It is important to note here that with the emergence of Schema.org (which provides the family of schemas to allow web developers to specify structure and unique identifiers to their information, recognizable by the Google, Bing, and Yahoo search engines), the computation of incentives becomes trivial after establishing correspondence between Schema.org and the respective ontology.

7.5.1 Computation

The ranking approach allows the specification of the relative importance of each measurement by setting an appropriate weight for richness, usage and incentives at

⁵<http://purl.org/goodrelations/v1.owl>; retr. 24/03/2012

⁶<http://support.google.com/webmasters/bin/answer.py?hl=en&answer=146750>; retr. 23/3/2012

0.3,0.5,0.2, respectively. The numeric values of each measurement are calculated by accessing the knowledge base containing both the terminological statements (T-Box) to measure richness and assertional statements (A-Box) to measure the usage of a given ontology. For example, the *CR* value of `gr:BusinessEntity` in Table 7.2 is calculated by querying the ontology graph which returns 5 relationships and 9 attributes giving 14 as the raw value of *CR*. For *CU*, SPARQL query (See Figure 7.5) returns 62,347 instances of business entity type. Given, $t = \text{gr:BuisnessEntity}$ is a concept and its incentive value is 0.433, the normalized values with respective weights in Eq. 7.9 is:

$$\text{Rank}_t = (0.3 * 14/31) + (0.5 * 62,347/989,638) + (0.2 * .433) = 0.254$$

Rank_t computes the rank value for t . For this experiment, we have given, comparatively, the most weight to the actual usage aspect of the concepts, followed by richness and the lowest weight to the incentives which are adjustable. Moreover, in the computation of incentives, three traditional search engines are considered: Google, Yahoo and Bing and 0.5,0.3 and 0.2 weights are given based on their popularity⁷.

```

1  SELECT ?instance, COUNT(?cls) as ?freq
2  WHERE
3    {
4      ?cls a ?instance .
5      {
6        SELECT ?instance
7        FROM <http://purl.org/goodrelations/v1#>
8        WHERE
9          {
10         ?instance owl:Class .
11        }
12      }
13    }

```

Figure 7.5: SPARQL query to compute CU metrics value

7.5.2 Observations

In Table 7.2, it can be seen that the highest ranked concept is `gr:Offering` because it has the highest value of all the three measures, however, there are different concepts which are rich in terms of their description, but due to the usage factor, they have a low ranking score. For example, `gr:ProductOrService` and `gr:Individual` are concepts with a high richness value coming in 3rd and 4th position in the *CR* index but due to their lower instantiation (usage), they are placed in the 6th and 7th position in

⁷Note that I didn't follow any formal search ranking or popularity index and only used expert judgement to estimate the weight

the overall ranking. There are six concepts which have no usage (instantiation) in the dataset: `gr:ProductOrService`, `gr:DeliveryChargeSpecification`, `gr:PaymentChargeSpecification`, `gr:QualitativeValue`, `gr:License`, `gr:Brand`. The last two concepts `gr:License` and `gr:Brand` are new concepts recently added to the ontology model, therefore their usage is not evident from the dataset.

Another important observation to make is the CU of `gr:ProductOrService` concept which has the 4th highest richness value in the table (see Table 7.2). This is because `gr:ProductOrService` is the super-class of its taxonomical (is-a) hierarchy, having three more specialized concepts, allowing users to annotate data with the most specialized concepts. This use of specialized concepts promotes specificity in describing semantic information, but on the other hand, while querying the RDF data, the user might use the highest upper-level concept instead. Here, we recall that taxonomic hierarchy implements subsumption behaviors, which in OWL, means necessary implication (Rector et al., 2004). This means that all the instances (or individuals) of sub-concepts (sub-classes or leaf-concepts) are also instances of the super concepts (super-class or upper-class). In order to allow the upper-level concepts to reflect the usage of their lower-level concepts, I extended the concept usage metric which implements the sub-class axioms to subsume the instance of sub-concepts (see Eq. 7.5). The results shown in Table 7.2 are obtained from the dataset by considering the knowledge available in the dataset and not the one which can be inferred using ontology reasoning.

In the incentive column, only very few concepts can be seen with non-zero values. This is due to the fact that there is very limited evidence available on what is being used from these ontologies by these search engines to index the structured data annotated using explicit semantics. Based on our previous study (Ashraf et al., 2011) in which we investigated Google RichSnippet (Steiner and Hausenblas, 2010) results to map the concepts with a Rich Snippet component, a list of concepts that are approximately used by Google is built. In addition to this, we also analyzed the Schema.org mappings⁸ to find equivalent terms in other ontologies to create a rudimentary list of terms (of different ontologies) syntactically and semantically matching the terms defined by Schema.org.

7.5.2.1 Usage related observations

After discussing the approach used to calculate the ranking of ontology terms based on different measures, we present the usage analysis of GR terms. In Table 7.3, the terms are arranged into three groups: concepts, object properties (relationships)

⁸<http://schema.rdfs.org/mappings.html>; retr : 28/3/2012

Table 7.2: GoodRelations Concepts Usage Analysis and their rank considering richness, usage and incentive measures.

Concept Terms	CR	CU	Incentive	Rank
gr:Offering	1	1	0.433	0.887
gr:SomeItems ¹	0.806	0.459	0.167	0.505
gr:ProductOrServiceModel	0.871	0.231	0.233	0.423
gr:UnitPriceSpecification	0.452	0.525	0	0.398
gr:TypeAndQuantityNode	0.194	0.476	0	0.296
gr:ProductOrService	0.710	0	0.233	0.260
gr:BusinessEntity	0.452	0.063	0.433	0.254
gr:Individual ¹	0.774	0.001	0	0.233
gr:QuantitativeValueFloat	0.323	0.243	0	0.218
gr:Location ¹	0.194	0.006	0.333	0.128
gr:DeliveryChargeSpecification	0.419	0	0	0.126
gr:PaymentChargeSpecification	0.387	0	0	0.116
gr:PriceSpecification	0.355	0.001	0	0.107
gr:QuantitativeValueInteger	0.323	0.003	0	0.098
gr:QualitativeValue	0.290	0	0	0.087
gr:QuantitativeValue	0.226	0.035	0	0.085
gr:OpeningHoursSpecification	0.226	0.023	0	0.079
gr:License	0.194	0	0	0.058
gr:DayOfWeek	0.129	0.001	0	0.039
gr:WarrantyPromise	0.129	0.001	0	0.039
gr:PaymentMethod	0.065	0.002	0	0.020
gr:PaymentMethodCreditCard	0.065	0.002	0	0.020
gr:BusinessEntityType	0.065	0.001	0	0.020
gr:BusinessFunction	0.065	0.001	0	0.020
gr:DeliveryMethod	0.065	0.001	0	0.020
gr:DeliveryModeParcelService	0.065	0.001	0	0.020
gr:WarrantyScope	0.065	0.001	0	0.020
gr:Brand	0.065	0	0	0.019

¹ These are the new concepts in the replacement of deprecated concepts. For further details visit <http://www.heppnetz.de/ontologies/goodrelations/v1.html>; retr 28/3/2012.

and datatype properties (attributes) of the GoodRelations ontology. The rank of each term is calculated by incorporating the three aspects, namely the richness of the term in the ontology; the use of each term in the dataset; and the incentives based on the term's acceptance in different traditional search engines. 15 concepts listed in descending order with `gr:Offering` being the highest ranked in the list, 17 object properties creating relationships between entities, and 17 datatype properties to provide textual description to the entities. In the concept list of Table 7.3, it can be seen that `gr:SomeItems`, `gr:ProductOrServiceModel` and `gr:Individual` have a higher ranking than `gr:ProductOrService` which does not even have significant usage but its richness and incentive values have helped it to have a close rank with its specialized concepts. In the object property list, relationships are listed according to their rank value. From the list, one can gain a better understanding of how the entities of different types are linked with each other to create a semantic description of the overall eCommerce data. As expected, the properties with highly ranked concepts such as `rdfs:range` or `rdfs:domain` also have a high rank in their listing. For example, the top five properties have `gr:Offering` as their domain with `gr:offers` as range. This helps in realizing the sub-model of the ontology which has high use, forming a light ontology which is useful in different scenarios such as data integration and prioritizing the indexing strategy. The third group is a list of attributes with their rank values. These attributes are very useful in exploring the textual description of entities.

The availability of different attributes with statistics about their usage is important for querying the data, particularly on the Web where no predefined (contrary to the relational databases) schema is available. Knowing which attributes are frequently used allows user interfaces to be built for exploratory search- and knowledge-driven applications. From the datatype property list (Table 7.3), `gr:description` is top of the list, for it is not only heavily used but also because it has the highest richness value. This attribute allows the provision of textual information about entities, thus making it highly rich in terms of its coverage and usability. Another notable attribute is `gr:legalName` which, despite having a low richness value, due to its significance in providing human readable names of companies/organization offering their product on the Web, has a high usage value. The terms listed in Table 7.3 enable users to understand the prevalent conceptual schema in a domain-specific implementation (eCommerce, in our use case) and use this information for different application scenarios, including the use case requirement highlighted in Subsection 7.4.1.

Table 7.3: GoodRelations ontology terms and their ranking

Concept		Object Property	
Term	Rank	Term	Rank
Offering	0.887	hasBusinessFunction	0.667
SomeItems	0.505	offers	0.575
ProductOrServiceModel	0.423	availableAtOrFrom	0.566
UnitPriceSpecification	0.398	includes	0.554
TypeAndQuantityNode	0.296	hasPriceSpecification	0.496
ProductOrService	0.260	typeOfGood	0.393
BusinessEntity	0.254	acceptedPaymentMethods	0.388
Individual	0.233	hasManufacturer	0.388
QuantitativeValueFloat	0.218	eligibleCustomerTypes	0.367
Location	0.128	includesObject	0.345
DeliveryChargeSpecification	0.126	eligibleTransactionVolume	0.333
PaymentChargeSpecification	0.116	hasEligibleQuantity	0.300
PriceSpecification	0.107	hasMakeAndModel	0.300
QuantitativeValueInteger	0.098	isAccessoryOrSparePartFor	0.300
QualitativeValue	0.087	isConsumableFor	0.300
		isSimilarTo	0.300
		hasInventoryLevel	0.280

Datatype Property	
Term	Rank
description	0.864
eligibleRegions	0.433
name	0.406
validFrom	0.246
validThrough	0.246
hasUnitOfMeasurement	0.229
hasStockKeepingUnit	0.169
hasCurrencyValue	0.126
hasEAN_UCC-13	0.125
hasCurrency	0.124
valueAddedTaxIncluded	0.092
legalName	0.084
amountOfThisGood	0.081
hasValueFloat	0.058
hasMaxCurrencyValue	0.043
hasMinCurrencyValue	0.043
hasMinValue	0.037

7.6 FOAF Ontology Usage Analysis

In this section, the FOAF ontology (Brickley and Miller, 2004), which is regarded as one of the earliest⁹, most highly used¹⁰ and well researched (Ding and Finin, 2006; Ding et al., 2005; Sleeman and Finin, 2010) ontologies by the Semantic Web community is examined. In the latest version (accessed on 3/4/2012), the FOAF ontology comprises 19 classes, 40 object properties and 27 datatype properties. The FOAF ontology provides the vocabulary to express information about people, their interests, relationships and activities. In Table 7.4, the usage analysis of the FOAF ontology is presented, based on the use of FOAF terms in the dataset.

7.6.1 Observation

The first column in Table 7.4 lists the most highly used concepts in descending ranking order. foaf:Person is the mostly instantiated concept used to defined the person entity followed by foaf:Agent and foaf:Document. It is interesting to note that similar to GR, only a few concepts are used in the implementation of these ontologies on the Web for semantic annotation. In the FOAF ontology, 58.82% of concepts, 40% of object properties and 37% of datatype properties are used with varying frequency, making approximately half of the terms in use and others without instantiation. This usage trend is similar to that reported in (Ding et al., 2005) and in (Ashraf et al., 2011), which reported that a small part of the ontologies are, in fact, being used by a large number of data publishers. Such usage patterns are somewhat desirable to promote a consistent schema to represent entities of interest such as people, place and documents in describing social network information. Referring back to our use case and reflecting on the requirements highlighted under the use case scenario section, the first requirement was to identify the applicable ontologies which is accomplished by identifying all the ontologies which are presently being used by the relevant community. A domain-specific dataset is used to achieve relevance and specificity. The identified ontologies were then analysed to measure their usage and understand the usage patterns available. The usage analysis helps in identifying the terminological knowledge that has better prevalence and prominence in the published data and uses it to construct the web schema to be used for our semantic annotation on the Web.

The next section describes how the results obtained by the QUA-AF framework can be used to realize the application of OUA.

⁹The FOAF homepage (<http://www.foaf-project.org/about>; retv 3/4/2012) states that FOAF project started in 2000.

¹⁰PingTheSemanticWeb.com (<http://pingthesemanticweb.com/stats/namespaces.php> ;retv. 3/4/2012) ranks FOAF at number one in terms of its usage in the documents indexed by them.

Table 7.4: FOAF ontology terms and their ranking

Concepts		Object Properties	
Term	Rank	Term	Rank
Person	0.606	homepage	0.664
Agent	0.442	Img	0.576
Document	0.427	thumbnail	0.554
Image	0.339	page	0.471
Organization	0.301	member	0.406
PersonalProfileDocument	0.201	maker	0.399
OnlineAccount	0.147	isPrimaryTopicOf	0.369
Group	0.139	depiction	0.357
OnlineChatAccount	0.119	based_near	0.256
Project	0.114	mbox	0.250
		primaryTopic	0.155
		account	0.147
		Made	0.147
		Knows	0.116
		topic	0.113
		logo	0.103

Datatype Properties	
Term	Rank
name	0.584
familyName	0.474
lastName	0.452
gender	0.266
firstName	0.065
mbox_sha1sum	0.056
accountName	0.054
status	0.05
givenName	0.018
title	0.018

7.7 Quantitative Analysis Evaluation

The objective of the QUA-AF framework is to quantitatively analyse the use of ontologies and transfer the analysis into actionable information which can then help in realizing the benefits offered by Ontology Usage Analysis. In Section 7.4, a use case is presented which uses the QUA-AF framework to obtain the required insight to implement the use case scenario. Recall that the use case contains the following three main requirements:

- identify the ontologies applicable to the Web site about the eCommerce domain
- understand which terms of a given ontology are highly used and should be reused to achieve a positive network effect
- provide a summarized view on the prevalent use of ontologies in the form of a Web Schema to facilitate the data publishing process, based on the quantitative analysis.

The next subsection discussed how using the results obtained by QUA-AF helps in addressing the abovementioned requirements of the use case

7.7.1 Requirement 1: Identify the ontologies application to an eCommerce website

The primary objective of using ontologies and publishing information using ontologies is to enable consuming applications to understand the information in such a way that they can automatically source and link the required information over the the Web. Therefore, for data publishers, it is very important to publish information using ontologies which is not only relevant to their application domain but is also recognised by the community. This will help in improving the reusability of ontologies which generates a positive network effect which enables the benefits of Semantic Web technologies to be realised.

The QUA-AF framework provides a list of ontologies used in the eCommerce domain which have usage and adoption on the Web. Table 7.1 lists the vocabularies being used covering the different types of information often used to structure eCommerce-related information. For a data publisher or data consumer, it is very useful to know what ontologies are in a given application area. For example, for eCommerce, *foaf*, *gr*, *v*, *dc*, *og*, *rev*, *vCard*, *virt*, *comm*, *frbr*, *vso*, and *pto* ontologies are being used as shown in Table 7.1 . Each of these vocabularies covers a specific

domain however, they are related to each other when co-used to describe information covering different aspects. An eCommerce website needs to describe information about different entities representing the respective domain. These entities include, but are not limited to, "company", "product", "offering", and "location".

7.7.2 Requirement 2: Identify the ontological terms to be used

After learning what ontologies are being used in a given domain, the next question to arise concerns what exactly is being used from these ontologies. This involves the identification of different terms which are being used and their relationships across different ontologies. The QUA-AF framework considers three dimensions: ontology richness, ontology usage and commercial incentives, to quantitatively analyse the use of ontologies, which provides the necessary insight to address the abovementioned requirement. The ranking of terms helps in filtering the most used and influential terms which data publishers need to consider while describing data on the Web. For example, in the case of the FOAF ontology which is often used to describe agents (human and/or non-human) and its attributes, the most used concepts are *Person*, *Agent*, *Document*, and *Image* as shown in the Concepts column in Table 7.4. Likewise, the object properties with highly ranked values are *homepage*, *img*, *thumbnail*, *page*, *member* and *maker* as shown in the Object Properties column in Table 7.4. The five top-ranked data properties are *name*, *familyName*, *lastName*, *gender* and *firstName* as shown in the Datatype Properties column Table 7.4.

Such insight into which the use of ontology-specific terms are quantified using multi-dimension criteria helps data publishers understand the present use of a particular ontology from different aspects. In addition to being of assistance to regular data publishers, this information is useful to ontology engineers and domain experts where the availability of such usage-related information provides feedback to inform future thinking. Ontology owners or domain experts can gain a better understanding of the conceptualised model when it has been ranked using different aspects. If a concept has a high richness value, which indicates how rich it is in terms of its semantic description and relationship, and also high usage, then it is safe to assume that there is a correlation between the richness of a concept and usage.

7.7.3 Requirement 3 : A summarized view on the prevalent use of ontologies in a given application area

For the GR ontology, there are 15 concepts ranked in descending order with `gr:Offering` ranked the highest, 17 object properties creating relationships between entities, and 17 datatype properties providing a textual description of the entities as shown in Figure 7.3. The terminological knowledge of the GR ontology which has usage and adoption can be obtained from Figure 7.3 to obtain a consolidated view of the given domain ontology. Application developers can use these results to develop prototypical SPARQL queries to retrieve the RDF graph containing the required data elements. Using these statistics, the data publisher can create semantic mapping with other ontologies, knowing which entities can further be described with rich semantics.

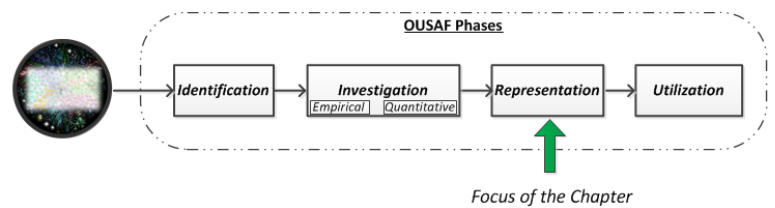
Constructing the ontological model based on the usage analysis (i.e. terminological knowledge represented in Table 7.3) will help the ontology developer to extract a light version of the ontology being highly used which could help in ontology evolution. Since the usage analysis can be applied in different scenarios, a threshold value can be given to the value for each metric to obtain a partial list of terms which can be adjusted, depending on the requirements. For example, setting the threshold value of 0.6 for rank would return only one concept i.e `gr:Offering` from Table 7.2 . If the user is interested in annotating the information to have a better position in the search engine result pages, then the top terms with higher incentive values would be required and, hence, the threshold value can be adjusted accordingly. Likewise, if the user is building a Semantic Web application to consume the data, then using a lower threshold value would help in querying the dataset to have high recall. Applying usage analysis on all relevant ontologies will provide the knowledge patterns dominant in each ontology to obtain a meaningful summarization of a web schema presently dominating the Semantic Web dataspace.

7.8 Conclusion

In this chapter, the QUA-AF framework is presented to quantitatively analyse the use of domain ontologies on the Web. The QUA-AF framework measures usage from three dimensions: ontology richness, ontology usage and commercial incentives. The inclusion of these dimensions provides erudite insight and a multi-dimensional view on the state of ontology usage and its adoption. Against each dimension, metrics are developed to measure ontologies from different aspects and then a ranking approach is used to obtain a consolidated ranking of the terms of an ontology. A web schema

generation use case is used to realise the benefits of the QUA-AF framework in which, based on the usage of different ontologies in the eCommerce domain, a schema representing the prevalent structure over the Web is generated. The generated Web Schema helps data publishers, data consumers and application developers understand what ontologies and their components are being used in real world settings, which helps them decide what they should consider. In the next chapter, the representation phase of the OUSAF framework is presented. In the representation phase, the ontology usage analysis is conceptualised using a formal representation model to allow applications and users to access the analysis results for further processing.

Chapter 8 - Representation Phase: Ontology Usage Ontology (U Ontology)



8.1 Introduction

As mentioned in earlier chapters, there has been a huge increase in the use of ontologies and Semantic Web data. This has increased the need for the availability of usage-related information to assist different stakeholders (or users) make effective use of currently available semantic information. The stakeholders can be different groups of users such as ontology developers, domain experts, application developers and data publishers. Each of whom has a view-specific requirement of the same information. The identification and investigation phases of the OUSAF framework, discussed in Chapters 5-7 help us to identify and measure usage-related information. However, once the usage-related information has been determined, it needs to be presented to the stakeholders in a structured format, therefore, providing granular access to ontology usage-related information meets the needs of each stakeholder. This is done in the *Representation* phase of the OUSAF framework, in which an ontology usage ontology (the U Ontology) is developed to represent ontology usage analysis-related information. In this chapter, the conceptual framework of the U Ontology is presented.

The rest of the chapter is organized as follows. Section 8.2 discusses the different aspects and the high level requirements to be considered while representing ontology usage. In Section 8.3, a customized methodology and the different phases of developing the U Ontology is presented. The activities performed in each phase of the methodology are presented in the next sections. Section 8.4 describes the specification phase which defines the scope and captures the ontology requirements. Section 8.5 describes the conceptualization phase in which the conceptual model is developed. Section 8.6 describes the formalization phase in which the ontology conceptual model is formally represented. Section 8.7 presents the implementation phase of the adopted methodology which implements the ontology by encoding it, using formal ontology language. Section 8.8 concludes the chapter.

8.2 Different Aspects to be Considered while Representing Ontology Usage

In chapters 5-7, by using the OUSAF framework, domain ontologies are identified and analysed from different aspects is an attempt to establish a detailed understanding on how ontologies are being used. The objective of measuring ontology usage is to provide erudite insight into the usage statistics and usage patterns to facilitate further adoption and uptake. As mentioned previously, such insight helps in influencing the reusability, evolution and even the future thinking on ontology development and reuse. The obtained quantified measures pertaining to usage need to be made available for its consumption and further utilization. Two aspects need to be considered: *users* and *structure* while representing ontology usage-related information. Different types of users are interested in different parts of the information pertaining to ontology usage, therefore their needs should be analysed and considered. Additionally, the structure of information and the mechanism to disseminate usage-related information need to be considered while implementing the representation and utilization phase of the OUSAF framework. These two aspects are discussed in the following sub-section.

8.2.0.1 Different type of user

Often, different people become involved in different stages of the ontology lifecycle and thus, they need to access information which is relevant to them. Therefore, each user, based on his role in the ontology lifecycle model, may require different views of the information to perform the tasks. The ontology lifecycle model, as described in Chapter 1, mainly comprises the development stage and in-use stages. The different types of

users who interact with ontologies can be categorised as ontology developers/owners, domain experts, application developers and data publishers.

- *Ontology developers* are interested in knowing how the developed ontology is being used and which components of the ontology are either ignored or under-used. The availability of empirical analysis on the use of ontologies provides needed insight for the developers to plan changes in their ontologies. So, ontology developers are interested to know the usage statistics of the ontology component which includes concepts, relationships, attributes and axioms. Identifying the use of different concepts, appearing in multiple data sources, and their relationships provides a snapshot of the invariant knowledge patterns on the ontology. Such information is useful to ontology developers in terms of understanding the needs and usage behaviour of the users.
- *Application Developers* are interested in knowing what sort of data is available for their applications. Information about the use of ontologies helps the developer to anticipate the nature and structure of data to develop data-driven applications and interfaces. In the case of linked data-driven applications ([Iqbal et al., 2009](#)), developers can take advantage of the available terminological knowledge to support development activities.
- *Data publishers* are interested in knowing which ontologies are highly used and given an ontology, which fragment of the ontology is more dominant. One of the immediate benefits which motivates data publishers to publish semantically rich structured data is the availability of machine understandable and processable information on the Web. Prominent search engines like Google (www.google.com) have also started parsing the structured data embedded within Web pages, which motivates publishers to publish their data with them. However, to take advantage of the benefits presently available, it is important for them to reuse the vocabularies which are already used by the community ([Heath and Bizer, 2011](#))

8.2.0.2 Structure and format

The second aspect to consider is the format and structure in which the ontology usage analysis needs to be represented. Ontologies, which are based on Web architecture ([Berners-Lee, 1998a](#)) are meant to be equally useful to humans and machines, so information-related ontology usage should be preferably based on these architectural principles. Hence, while representing usage analysis, the future possibilities of information processing, the availability of globally accessible data accessible and the

definition of the canonical terms which can promote information interoperability need to be considered.

Considering the abovementioned key aspects (users and structure), which also can be seen as non-functional requirements, a domain independent, machine-readable conceptual model in the form of an **Ontology Usage Ontology (U Ontology)** is proposed to represent domain ontology usage. The U Ontology is an ontology which formalizes the representation of ontology usage analysis by standardizing the domain knowledge related to ontology usage analysis. It represents the use of an ontology, the use of its different components, usage statistics, and co-usage with other ontologies.

In order to develop the U Ontology, in the next section, the ontology development methodology and the different developmental stages are presented.

8.3 Methodology Adopted for U Ontology

As discussed in Chapter 2, several methodologies are proposed in the literature to support ontology development-related activities. For the development of the U Ontology, different methodologies were studied (Changrui and Yan, 2012) , (Fernandez-Lopez et al., 2002) to identify the methodology that is suitable for U Ontology construction. Three methodologies which were studied in depth are: METHONTOLOGY (Fernández-López et al., 1997); Ontology Development 101 (Noy and McGuinness, 2001); and Mike Uschold (Uschold, 1996) as depicted in Figure 8.1. From METHONTOLOGY, four major phases are adopted for the development of the U Ontology, and the other two methodologies (Ontology Development 101 and Uschold) are implemented in the realization of these phases. The combination of these methodologies provides the flexibility to adopt activities which are suitable for the development of the U Ontology. METHONTOLOGY (Fernández-López et al., 1997) is one the better known ontology building methodologies due to its suitability for building ontologies either from scratch or by reusing other ontologies. The methodology comprises four main phases, namely *Specification*, *Conceptualization*, *Formalization* and *Implementation*.

The four phases of the U Ontology development methodology and the set of activities involved in each phase are discussed in the following sub-sections.

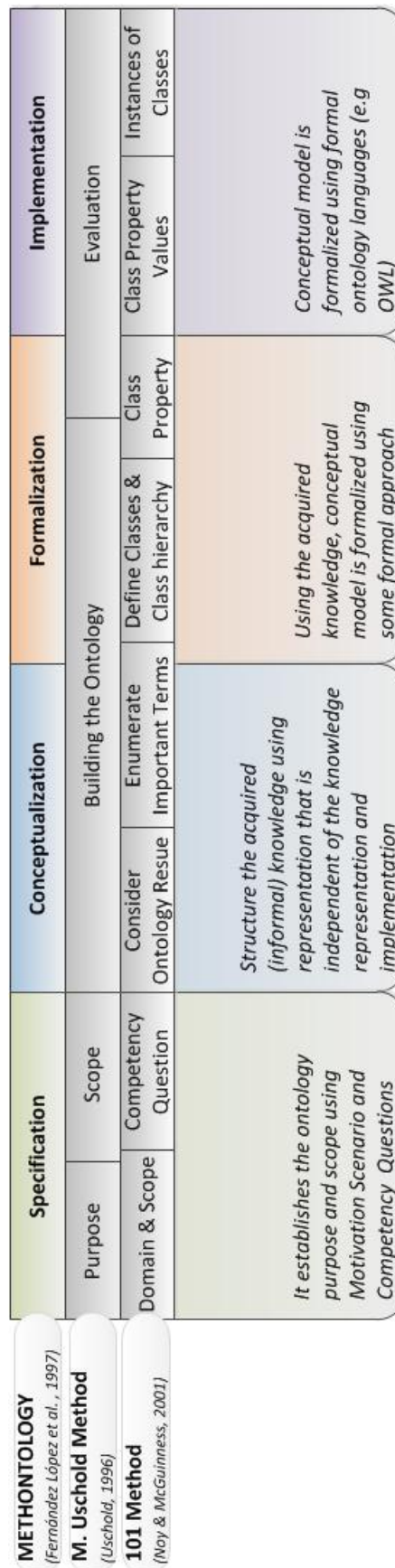


Figure 8.1: Different ontology development methodologies and their relationship.

8.3.1 Specification phase

The aim of the specification phase is to capture and document the ontology requirements. This includes activities such as, but not limited to, identifying the intended user, use cases and motivation scenarios. The Purpose and Scope phases of Uschold's (Uschold, 1996) unified approach and the Domain and Scope identification and Competency Question activities of Noy's 101 Method (Noy and McGuinness, 2001) closely relate and overlap with the definition of the specification phase and are used to identify the scope and boundary of domain knowledge. The required formality and format to represent and document the ontology specifications are not specified by any methodology however, Uschold (1996) classifies the level of formality of ontologies as: highly formal, semi-informal, semi-formal or rigorously formal. They propose to use the motivation scenarios and informal competency questions to specify the scope and capture the requirements of the ontology. In other work Sure et al. (2002a) suggests an ontology requirement specification document which describes the following set of information:

- Domain and goal of the ontology
- Guidelines for designing the concepts, instances and conventions to be followed
- Scope, including the terms to be represented and the background to capture the prior knowledge
- Users, use cases and application support for ontology

In light of the abovementioned discussion, the specification of an ontology is required to capture the scope of the ontology and elicit the requirements specifications to be used for ontology modeling. Therefore, for the development of U Ontology, the requirements of the specification phase are:

- Capture the scope of the ontology
 - Identify the key users
 - Describe the common use case scenarios to spell out the requirements in a more descriptive way.
 - From each scenario, extract the key requirements in the form of competency questions to develop the detailed ontology requirements specifications.
-

8.3.2 Conceptualization phase

The goal of the conceptualization phase is to structure the domain knowledge in the form of a conceptual model. This means that the vocabulary which represents the domain needs to be identified and documented to specify the terminology of the domain. The terminology helps in building a common vocabulary within a domain to identify the basic concepts and the relationship between these concepts. The steps in this phase are :

- Considering the common vocabulary of the domain, build a complete Glossary of Terms (GT). Since the terms represent the common vocabulary, it contains the concepts, instances, and relationships and (attribute) properties.
- Group the identified terms into concepts and relationships. Consequently, for each group of closely related terms, a concept classification tree is built.

These steps help in producing the conceptual model which can then be used to verify the models usefulness and usability. For the conceptualization phase, [Corcho et al. \(2005\)](#) proposed 11 steps to carry out the required activities based on METHONTOLOGY. In their methodology, the activity proposed by [Noy and McGuinness \(2001\)](#) for the ontology reusability stage is to identify the potential ontology candidates for reusability. After the identification of terms and the grouping of terms, it is worth checking to see if someone has already undertaken conceptualization for a similar domain. Aside from reusing existing ontologies as best practice, sometimes it is even a requirement if the system needs to interact with other systems that have already committed to a particular ontology.

In light of this, the conceptualization of an ontology is required to capture the key terminology describing the domain knowledge. Therefore, for the development of the U Ontology, the requirements of the conceptualization phase are:

- Identify the key terminologies describing the domain knowledge
 - Based on the identified terminology (vocabulary), search for similar terminologies in existing ontologies
 - Evaluate the potential reusable terms to verify their applicability to the U Ontology
 - Structure the key terms based on their relationship
-

8.3.3 Formalization phase

The goal of this phase is to formalize the conceptual model developed in the previous phase (conceptualization). Formalization refers to the creation of a neutral ontology formulation that is independent of the underlying language and platform (Guebitz et al., 2012). The transformation of the conceptual model can be performed at different levels of formalization, ranging from a fully formal model to a semi-computable model, depending on the implementation requirements of the ontology. One of the common formalisms that is preferred is the object-oriented modelling language due to familiarity with object-oriented modelling paradigm and the availability of its tools (Graham et al., 2001). Since Noy and McGuinness (2001), methodologies mainly focused on the conceptualization stage of ontology development, therefore this suggests the development of class hierarchies and the specification of values for properties defined in the conceptual model.

Fernández-López et al. (1997) also included integration at this stage to integrate all the definitions considered for reuse into the formalized model. This helps to identify any inconsistencies which may have occurred due to the inclusion of concepts defined in other ontologies which can then be resolved before the implementation stage.

In light of the abovementioned discussion, the formalization of an ontology requires the representation of a conceptual model using a formal approach that is independent of the underlying platform and application settings. Therefore, for the development of the U Ontology, formalization requires:

- selecting a formal model for formalizing the U Ontology's conceptual model
- using the selected formal approach to formalize the conceptual model
- Integrating the formal model with the other ontological models that are selected for reuse (in the conceptualization phase).

8.3.3.1 Implementation phase

The goal of the implementation phase is to encode the formalized model using a formal ontology language. Usually, implementation tools such as ontology editors are used for ontology implementation. For implementation, there are several ontology development environments available such as: Protégé (Knublauch et al., 2004), NeOn Toolkit (Haase et al., 2007), TopQuadrants TopBraid Composer (TopQuadrant, 2011) and OntoEdit (Sure et al., 2002a). Regarding the ontology language, depending on the expressivity requirements, different language with varying expressivity are available (Gomez-Perez and Corcho, 2002). However, the W3C-based ontology languages such

as RDFS and OWL (including its different species such as OWL-Lite, OWL-DL and OWL-Full) are commonly used to encode ontologies in a formal ontology language.

Hence, the implementation of an ontology requires the encoding of the formal conceptualised model. Therefore, for the development of the U Ontology, the requirements of the implementation phase are:

- choosing the formal language to be used to encode the ontology
- selecting the ontology development environment to support the ontology construction-related activities.

Using the adopted methodology, the steps in the development of the U Ontology are as follows.

8.3.4 Steps involved in the development of U Ontology

For the development of the U Ontology, a customized approach is adopted. The set of activities, methods and tools involved in the customized methodology are depicted in Figure 8.2 and discussed in the following sub-sections.

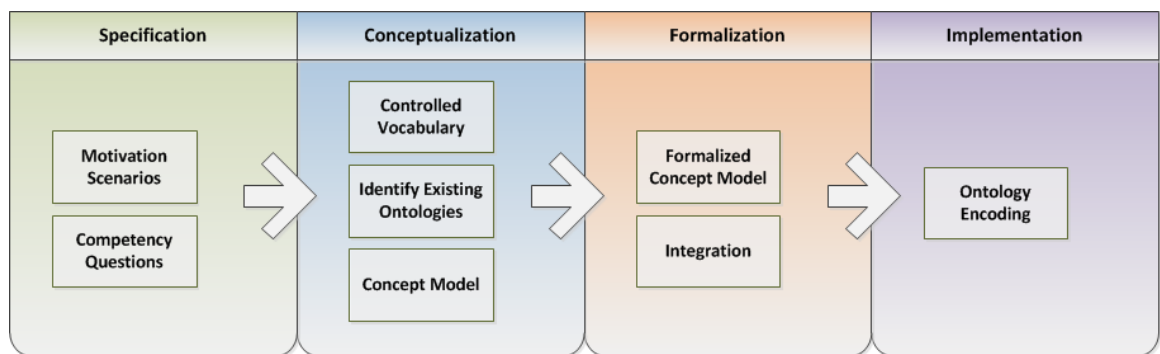


Figure 8.2: Customized ontology development methodology for the U Ontology.

8.3.4.1 Step 1 : Develop Motivation Scenario

The goal of this step is to develop detailed motivation scenarios. The purpose of motivation scenarios is to obtain a clear picture of the scope of the ontology (Uschold, 1996). Another advantage of developing motivation scenarios is that it helps in gleaning the concepts, terms and relationships that will help in developing the controlled vocabulary and conceptual model. As suggested by (Uschold, 1996), the

motivational scenario should start with a general scenario and evolve into specific ones in a hierarchical manner.

8.3.4.2 Step 2 : Develop Competency Questions

The goal of this step is to develop competency questions based on the developed motivation scenarios. The purpose of competency questions is to express the requirements and usage scenarios in a detailed and thorough manner to assist in capturing the complete domain knowledge expected to be represented by the ontology. Competency questions further help in verifying whether the ontology fulfils the use cases mentioned in the motivation scenarios.

8.3.4.3 Step 3 : Develop Controlled Vocabulary

The goal of this step is to develop a glossary of terms that represents the domain knowledge. The purpose of controlled vocabularies is to have a complete list of terms being used by the users and the domain experts during discourse. The glossary of terms helps in classifying the different components of speech, such as nouns and verbs, and identifies the entities being represented by the domain. As suggested by [Fernández-López et al. \(1997\)](#), the motivation scenarios and their competency questions are if well documents, they will become the main source of input for building the glossary of terms.

8.3.4.4 Step 4 : Identify Existing Ontologies

The goal of this step is to identify the existing vocabularies and consider reusing their definitions where possible. The purpose of reusing existing ontologies is to encourage reusability in ontologies, as by definition, ontologies are understood as a means of sharing and reusing knowledge ([Simperl, 2009](#)). After building the glossary of terms, it is necessary to check existing terminologies (vocabularies) to determine if any of these can be reused instead of developing the required terms from scratch. To do this, one can access ontology libraries and/or undertake a web search to find ontologies which have similar definitions of terms. In order to find the similarity between these terms, a manual effort is required to decide which terms to reuse or develop from scratch.

8.3.4.5 Step 5 : Develop Conceptual Model

The goal of this step is to develop the conceptual model of the domain in focus. The purpose of the conceptual model is to arrange the identified concepts (terms) in structural and hierarchical order to group the relevant terms. The conceptual model remains the same, regardless of the formal model and formal language used later to model and implement the ontology. During this step, the five components (classes, relationships, functions, instances and axioms) are included in the conceptual model along with their constraints. Relationships, particularly taxonomical relationships, are included in the model to reflect the is-a relationships present in different terms (concepts).

8.3.4.6 Step 6 : Formalize Conceptual Model

The goal of this step is to formalize the conceptual model. The purpose of formalization is to create a neutral ontology formulation that is independent of the underlying implementation language that can be used to serialize the ontology (Guebitz et al., 2012). One of the preferred choices for formalizing the conceptual model (Cranefield and Purvis, 1999) is the use of the Unified Modelling Language (UML). UML is a (industry-based) standard modelling language which provides graphical notation, a set of diagrams and other components necessary for developing the software engineering models. A UML Class diagram is used to represent the concepts of ontologies and mainly comprises three elements: the name of class, attributes of class and operations of class, as shown in Figure 8.3(a). In the context of ontology modelling, classes are known as concepts and attributes are known as attributes of concepts, however, operations of classes are not required as ontologies do not have operations. Therefore, the modified notational diagram to represent the concept of an ontology is shown in Figure 8.3(b).

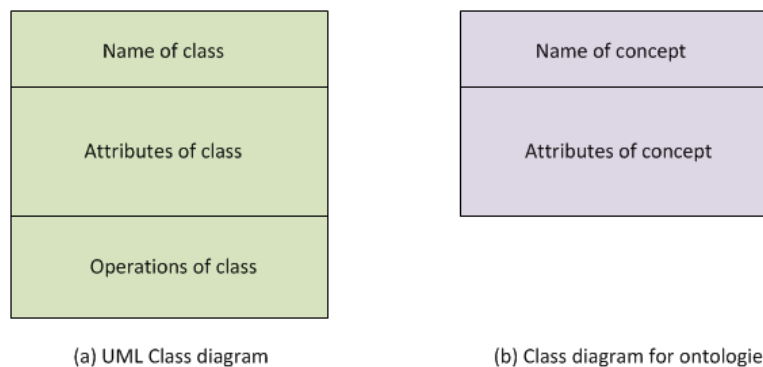


Figure 8.3: UML Class diagram for ontology modeling.

8.3.4.7 Step 7 : Integrate with other ontologies

The goal of this step is to integrate the formalized conceptual model with the external ontologies considered for reusability. The purpose of integration is to verify that the set of new reused definitions are aligned and based on the set of new basic terms. This helps in determining, in advance, if any inconsistencies exist before implementation. This involves verifying the semantic similarity between terms that are syntactically similar, but may not necessarily be semantically similar.

8.3.4.8 Step 8 : Ontology encoding (implementation)

The goal of this step is to codify the conceptual model using a formal ontology language. The purpose of ontology implementation is to develop an artifact which encodes the domain knowledge in a format that is understandable by different type of users, including humans and machines. An ontology development environment such as Protégé ([Knublauch et al., 2004](#)) is used to develop the ontology and syntactic and lexical analysers assist in resolving the syntactical and lexical errors ([Fernández-López et al., 1997](#)). A modern development environment such as Protégé allows the user to plug in different semantic (OWL) reasoners ([Bock et al., 2008](#)) to help users avoid any inconsistencies which may arise in the development of the ontology model or during integration with external ontologies. For the development of the U Ontology, Protégé is used as the development environment and encoding is based on OWL-DL expressivity.

In the next section, each of these steps is discussed in detail along with the implementation details.

8.4 Specification phase: Motivation Scenarios and Competency Questions

In this section, first the motivation scenarios are presented and then using these scenarios, competency questions are defined.

8.4.1 Motivational Scenarios to capture the requirements of users

Four motivational scenarios are presented to describe the requirements of ontology usage analysis from each users perspective.

8.4.1.1 Scenario 1 : Ontology Developers (owners)

Ontology owners/developers are usually interested in the following information:

1. Which components of the ontology are being used and what is the level of usage? This may include the use of different concepts and the use of different relationships to interlink different concepts.
2. Which attributes are being used by the ontology users to provide the instance data describing the entities defined by these concepts?
3. What other terms (vocabularies) are being more frequently used along with the ontology? This will assist in knowing the coverage of the instance data being described using the ontologies and will help in evaluating the scope of the ontologies and can be considered as input for ontology evolution.
4. Which components of the ontology have good adoption and which components are marginally used? This information can be obtained via the feedback loop (as depicted in Figure 1.5) which is based on the actual ontology instantiation.
5. In which application areas is the ontology being used, for example, for semantic annotation, data integration or building ontology-based knowledge applications?

8.4.1.2 Scenario 2 : Application developers

Application developers are usually interested in knowing the availability of different ontologies and how its different components are used in a given domain to consume the available information in a more effective manner hence, they would be interested in the following information:

1. What is being used and what is its adoption level in a given ontology? The availability of this information helps in making effective use of Semantic Web data.
-

2. How are the different ontologies linked? For the development of data-driven applications that are primarily based on the data published on the Web, information regarding ontology usage and co-usage with different ontologies helps to obtain a snapshot of the data structure present in the published data.
3. How are different textual properties used (also referred to as labelling properties)? This information is useful for developing user interfaces for data-driven applications.
4. What are the statistics on the use of different concepts, properties and attributes? This information will assist in anticipating the data load and planning data management accordingly.

8.4.1.3 Scenario 3 : Data publishers

For data publishers, it is very important to know and learn about the already adopted ontologies as their reuse offers better value for their publishing effort. Data publishers are usually interested in the following information:

1. Which ontologies are being frequently used and what is their level of adoption? This information will help in semantically annotating (application area specific) information on the Web. Increasingly, search engines are recognizing structured data embedded in Web pages which has been annotated using ontologies, therefore using the supported ontologies is highly desirable for data publishers to increase the visibility of their data. Reusing already used concepts which reflect community consensus generates a positive network effect, increasing data visibility and value.
2. What are the usage statistics? This information will assist in quantifying usage in order to decide which ontology or ontology components they should use to achieve the desired objectives.

8.4.1.4 Scenario 4 : Semantic Web practitioners/users

Semantic Web researchers/users are usually interested in the following information:

1. What are the prevalent knowledge patterns available on the Web? This information helps Semantic Web users and researchers know the prevalent structure of information invariantly published on the Web which assists them in analysing and inferring the relationships between the ontology conceptual model and the and the data structure prominent of the Web.
-

2. Which data patterns which are semantically annotated using domain ontologies are available on the Web?
3. How are different ontologies used by data publishers to semantically describe an entity?

The insight gained from this information will help ontology engineers and Semantic Web users understand the common needs of data publishers which can influence future thinking and research agendas.

To summarize, in order to address the requirements specified in the four motivating scenarios, a detailed and multi-dimensional analysis of ontology usage is needed. Aside from the identification of such information, a mechanism is required to represent the information (pertaining to ontology usage) in such a way that it can be accessed programmatically for automatic processing. The high level requirements that need to be identified and represented can be summarized as follows:

- Obtain the list of ontologies that are being used in a given application area
- Obtain an analysis of ontology usage covering different aspect of ontology usage
- Obtain a list of different ontologies to semantically describe the entities of the specific domain
- Identify prevalent knowledge and data patterns
- Obtain the usage statistics of ontology components such as concepts, relationships, attributes and axioms.

The acquisition of this information will help to identify the scope of the U Ontology which conceptualises the domain of ontology usage and its analysis.

8.4.2 Competency Questions to capture the scope of representation

Based on the four motivation scenarios and the detail required to perform ontology usage analysis empirically and quantitatively, the competency questions (e.g CQ1) and the sub-questions under them (e.g. CQ1.1) are listed below to specify the precise requirements for the U Ontology.

CQ1 What are the ontologies being used in a given application area (domain)?

CQ1.1 What are the namespaces of the ontologies being used in a given applications area?

CQ1.1.1 What is the namespace of a given ontology?

CQ1.1.2 What is the prefix used for a given ontology?

CQ1.2 What are the components of a given ontology?

CQ1.2.1 How many classes does an ontology have?

CQ1.2.2 How many relationships does an ontology have?

CQ1.2.3 How many attributes does an ontology have?

CQ1.2.4 How are different axioms being used in a given ontology?

CQ1.3 How is a given ontology's conceptual model structured?

CQ1.3.1 How many relationships does a concept have?

CQ1.3.2 What are the relationships a concept has?

CQ1.3.3 How many attributes does a given concept have?

CQ1.3.4 What are the attributes of a given concept?

CQ1.4 How are the relationships of a given ontology structured?

CQ1.4.1 How many concepts are in the domain of a given relationship???

CQ1.4.2 How many concepts are in the range of a given ontology?

CQ2 What is the richness of a given ontology?

CQ2.1 What is the richness value of a concept? is the richness value of a relationship?

CQ2.2 What is the richness value of an attribute?

CQ3 How is a given concept being used in real world implementation?

CQ3.1 What is the instantiation of a given concept?

CQ3.2 How are the entities of a given concept type semantically described?

-
- CQ3.3** What relationships from the ontology are used to describe the entities?
- CQ3.4** What attributes are used to provide the (factual) instance data describing entities?
- CQ3.5** What are the other concepts (of other ontologies) of which the given subject (entity) is an instance?
- CQ3.6** What are the other ontologies that are being used together to describe the entity?
- CQ4** How are the textual descriptions attached to the entities?
- CQ4.1** What are the (W3C-based vocabularies) label properties which are being used?
- CQ4.2** What is the usage of these (W3C-based vocabularies) label properties
- CQ4.3** What other (domain-specific) labeling properties are being used?
- CQ4.4** What is usage of these (domain-specific) label properties?
- CQ5** List the terms of a given ontology which are recognized by the search engines?
- CQ5.1** Is the given concept being recognised/supported by the X search engines?
- CQ5.2** Are the given relationships being recognised/supported by the X search engines?
- CQ5.3** Is the given attribute being recognised/supported by the X search engines?
- CQ6** What are the common knowledge patterns in the implementation of a given ontology?
- CQ6.1** What is the maximum path length in the traversal path (knowledge pattern)?
- CQ6.2** How many unique paths are leading from an entity?
- CQ6.3** What are the path steps in the traversal path?
- CQ6.4** What is the frequency of a given path step?
- CQ7** What ontology components have either no or limited usage?
-

CQ7.1 What concepts have not been used by data publishers?

CQ7.2 What relationships have not been used by data publishers?

CQ7.3 What attributes have not been used by data publishers?

CQ7.4 How can the ontology components that have a usage based on the specified threshold value be accessed?

8.5 Conceptualization phase: Controlled Vocabulary, Existing Ontologies and Conceptual Model

In this section, the conceptualization phase of the adopted methodology is presented. This phase involves the identification of the terminological knowledge describing the domain, identification of the existing ontologies for reuse and the development of the conceptual model. Each of these activities is discussed in the following sub-sections.

8.5.1 Controlled Vocabulary to identify the terminological knowledge

As mentioned in Section 8.3, in the conceptualization phase, all relevant terms of an ontology are defined to obtain a controlled vocabulary. The concepts related to ontology usage identified in Chapters 5-7 and the motivation scenarios and competency questions presented in Section 8.4, provide the basis for building the controlled vocabulary for the Ontology Usage Analysis domain. The developed controlled vocabulary is presented in Table 8.1

Table 8.1: Controlled Vocabulary for Ontology Usage Ontology (U Ontology).

Term	Ontology Component	Description
AttributeUsage	Concept	Quantifies the attribute usage.

Continued on next page....

Table 8.1 – continued from previous page

Type	Ontology Component	Description
Attribute	Concept	The attributes that are used for the concept being analysed. Attributes are the datatype properties used to provide literal (static) value.
AttributeValue	Concept	The richness value computed for a given attribute.
ConceptRichness	Concept	The richness value of a concept.
ConceptUsage	Concept	The usage of a given concept.
Dataset	Concept	The dataset used to measure ontology usage.
DataSource		The dataset used to measure the ontology usage
DomainLabel	Concept	The use of data properties that is defined in the domain ontology to provide the textual description for entities.
FormalLabel	Concept	The use of data properties that is defined in the domain ontology to provide the textual description for entities.
IncentiveDim	Concept	The use of the incentive dimension to measure the commercial advantages available to data publishers.
KnowledgePattern	Concept	The knowledge pattern present in the dataset. Knowledge patterns include the concepts and relationships used to describe the information.
LabelUsage	Concept	The frequency with which a label is used. This quantifies the use of a label property.

Continued on next page....

Table 8.1 – continued from previous page

Type	Ontology Component	Description
Measure	Concept	The measures that are used for Ontology Usage Analysis. It contains all the dimensions for which metrics are defined.
OntologyUsage	Concept	The use of ontology. This is a high level concept which represents other sub-concepts related to ontology usage and usage analysis.
OntologyUsageAnalysis	Concept	Top level concepts to represent the ontology usage analysis domain.
Path	Concept	The unique knowledge pattern in the dataset. A knowledge pattern is a set of triples chained together to form a path.
PathConcept	Concept	The concepts included in the knowledge patterns. Knowledge patterns comprise path steps which are in the form of triples. PathConcept represents the type of subject and object resources.
PathProperty	Concept	The relationship included in the knowledge patterns. Knowledge patterns comprise path steps which are in the form of triples. PathProperty represents the predicate of the triple.
PathStep	Concept	The data or knowledge patterns. A path step represents a triple present in the dataset.
Relationship	Concept	The use of object properties to describe the instance of Concept (i.e ConceptUsage).
RelationshipUsage	Concept	The use of relationships (object properties) of the ontology.

Continued on next page....

Table 8.1 – continued from previous page

Type	Ontology Component	Description
RelationshipValue	Concept	The computed richness value of a relationship.
RichnessDim	Concept	The richness dimension to measure the structural characteristics of the ontology.
SearchEngine	Concept	The search engine that supports the particular term (concept, relationship or attribute).
SoftwareSupport	Concept	The software component which could be an application, API, database or Reasoner that support the particular term (concept, relationship or attribute).
Source	Concept	The source that is being used as input for analysing ontology usage. This is a high level concept.
Streaming	Concept	The data source which is accessed by continuously crawling the Web.
Term	Concept	A high level concept representing the terminological knowledge of the ontology
UsageDim	Concept	The usage dimension to measure the use of an ontology and its component is real world implementation
Vocab	Concept	The different vocabularies/ontologies that are being used to describe the instance of a concept.
analysesOntology	Relationship	The ontology that is being analysed by OUA.
attributeValue	Relationship	The richness value of an attribute.

Continued on next page....

Table 8.1 – continued from previous page

Type	Ontology Component	Description
hasAttribute	Relationship	The attributes that are used to provide attribute values to the entity defined by the concept.
hasAttributeUsage	Relationship	The use of attributes (data type properties) of the ontology that is being analysed.
hasConceptUsage	Relationship	The concept of the ontology whose usage is measured and analysed.
hasDomainLabel	Relationship	The domain label used for a concept.
hasFormalLabel	Relationship	The formal label used for a concept.
hasIncentiveDim	Relationship	The incentive (commercial benefits) dimension of the ontology usage.
hasKnowledgePattern	Relationship	The knowledge patterns that are of a given ontology
hasLabelUsage	Relationship	The use of label properties. The label properties are used to attach textual descriptions to entities.
hasMeasure	Relationship	The measures used to perform ontology usage analysis.
hasObjectInPath	Relationship	The object of the triple represented by the path step.
hasPath	Relationship	The specification of the paths present in the knowledge patterns.
hasPathStep	Relationship	The specification of the path steps included in the path.
hasPropertyInPath	Relationship	The predicate of the triple represented by the path step
hasRelationship	Relationship	The relationships (use of object properties) that are used to describe the entity defined by the concept.

Continued on next page....

Table 8.1 – continued from previous page

Type	Ontology Component	Description
hasRelationshipUsage	Relationship	The use of relationships (object type properties) of the ontology that is being analysed.
hasRichness	Relationship	The richness value of the concept
hasRichnessDim	Relationship	Specifies the richness dimension of the ontology usage.
hasSubjectInPath	Relationship	The subject of the triple represented by the path step.
hasTerm	Relationship	The terminological knowledge of the ontology that is being analysed. This includes concepts, relationships (object property) and attributes.
hasUsageDim	Relationship	The usage dimension of the ontology usage.
hasVocab	Relationship	The specification of the vocabularies that are being used in order to describe the entity of a concept.
isComponentOf	Relationship	The association of a term with its ontology.
isCoused	Relationship	The two ontologies which are being co-used in the dataset.
isIncentivizedBy	Relationship	The relationship between the ontology term and the term supported by the search engine.
isSupportedBy	Relationship	The relationship between the ontology term and the software that supports the term

Continued on next page....

Table 8.1 – continued from previous page

Type	Ontology Component	Description
isUsedBy	Relationship	This property allows specifying the data sources (website/pay-level-domain) which are included in the dataset. This means the data sources which are making use of ontologies to describe information on the Web.
performedOn	Relationship	The specification of the source which is used to perform the ontology usage analysis.
relationshipValue	Relationship	The richness value of a relationship
analysisTimestamp	Attribute	The date and time the reported usage analysis was performed.
description	Attribute	Textual description of the resources.
docURI	Attribute	The URL (internet address) hosting the ontology related documents.
frequency	Attribute	The occurrence of the term.
hasCousedValue	Attribute	This represents the number of other ontology which are being co-used with given ontology. This value is represents the co-usage value obtained through Co-Usage ontology network (projected network degree)
hasInstantiation	Attribute	The number of instances of a concept.
hasUsers	Attribute	This property allows to specify the total number of users a particular ontology has. This represents a numeric value to tell how many different data sources are using the ontology

Continued on next page....

Table 8.1 – continued from previous page

Type	Ontology Component	Description
incentiveValue	Attribute	This represents the incentive value computed by the incentive metric for a term.
industry	Attribute	This property reflects the domain to which the crawled data belongs. This is a manual entry to classify the industry to which apparently crawled data belongs.
name	Attribute	The name of the concept and other resources.
prefix	Attribute	This is the prefix used in the dataset and during the analysis to refer to ontology.
searchEngineName	Attribute	Specifies the name of the search engines considered to measure the incentive value of each (supported) term
timestamp	Attribute	The date and time the information is obtained.
URI	Attribute	The URI of the ontology and ontology components.
URL	Attribute	This represents the address of the data source from where the Semantic Web data is crawled
usageValue	Attribute	This is the usage value of a term (concept, relationship, attribute) computed by Ontology Usage metric
value	Attribute	The numeric value.

The next activity is the identification of existing ontologies which can be reused for the development of the U Ontology.

8.5.2 Identify existing ontologies for reuse

The next step is to evaluate the existing ontologies to identify the ontologies which have potentially reusable classes and properties

The identified domain knowledge can be easily clustered into three groups of relevant information related to ontology usage analysis: *Ontology Usage*, *Ontology Metadata* and *Ontology Application* as depicted in Figure 8.4. The Ontology Usage cluster represents the domain knowledge specific to the usage analysis aspect of the ontologies, the Ontology Metadata cluster represents the domain knowledge specific to the metadata of the ontology, and the Ontology application cluster represents the domain knowledge specific to the application areas in which the ontologies are being deployed.

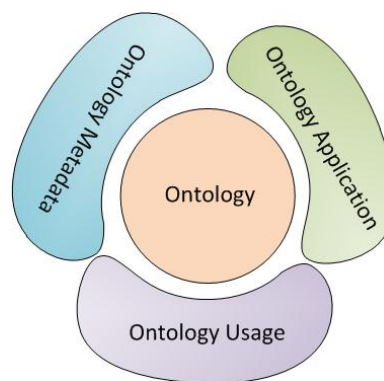


Figure 8.4: Ontology-related information clusters.

The Semantic Web community is working on the development of meta-level ontologies to capture the metadata of ontologies which can be used by other applications or ontologies to access ontology-related information. One such effort is the development of the Ontology Metadata Vocabulary (OMV)¹([Hartmann et al., 2005](#)) which is a standard proposal for describing ontologies and related entities. Members of the ontology community gather annually at the Ontology Summit and publish the summit proceedings on the ONTOLOG website². At the 2011 Ontology Summit³, the Ontology Application Framework (OAF) was presented with the aim of defining common terminology to describe applications of ontologies and the benefits that ontologies deliver within these applications.

These two projects, which are described in the following sub-sections, complement

¹The project details can be found at <http://mayor2.dia.fi.upm.es/oeg-upm/index.php/en/technologies/75-omv>; retr. 25/12/2012

²<http://ontolog.cim3.net/>; as part of retr 26/12/2012

³The summit was chaired by Professor Michael Gruninger (University of Toronto) and Dr. Michael Uschold (Semantic Arts) and the Ontology Application Framework was presented. <http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2011>; retr. 26/12/2012

the ontology usage analysis domain and relate to the Ontology Metadata cluster and the Ontology Application cluster, shown in Figure 8.4.

8.5.2.1 Ontology Metadata Vocabulary

11.41 The objective of the Ontology Metadata Vocabulary (OMV) is to provide a standard approach to describe ontologies and related entities (Hartmann et al., 2005). The OMV ontology is considered an ontology metadata standard to annotate ontologies. The use of OMV promotes the features which enhance reusability that are equally accessible for both human and machines. OMV is designed modularly and comprises OMV code and OMV extensions. The OMV code captures the key information that is relevant to the majority of ontologies, whereas the OMV extension allows ontology users to provide more specialized, application-specific, ontology-related information.

There are two main classes in OMV around which other concepts are defined. The two main classes are `OntologyDocument` and `OntologyBase`. `OntologyBase` (OB) represents the conceptualization of the ontology, whereas `OntologyDocument`(OD) represents the realization of the conceptualized ontology. The other classes in OMV core are `Person` and `Organization` to specify the party responsible for creating, reviewing, contributing and applying the ontology. The method, tools and formal language used for developing the ontology are described using `OntologyEngineeringMethodology`, `OntologyEngineeringTool`, and `OntologyLanguage` classes, respectively. The serialized form of an ontology is available at <http://omv2.sourceforge.net/> (retr., 26/12/2012) and for more descriptive details, readers are referred to 2008

8.5.2.2 Ontology Application Framework

As previously mentioned, the Ontology Application Framework (OAF) was presented at the Ontology Summit 2011 with the aim of making a case for the use of ontologies by providing concrete application examples, success/value metrics and advocacy strategies (Uschold et al., 2011). The objective of OAF is to present a common terminology for describing the application scenarios in which ontologies are being used. It also captures the benefits and value that can be achieved from the applications due to the use of ontologies. Additionally, it provides a basic vocabulary to represent benchmarks and has the ability to compare different applications of ontologies (Uschold and Jasper, 1999). Several of the areas of ontology use are: integration, decision support, semantic augmentation and

knowledge management. The conceptual representation of OAF is available at <http://ontology.cim3.net/file/work/OntologySummit2011/ApplicationFramework/DWL-Ontology/OntologyApplicationFramework-WithDocumentation.pdf> (retr. 26/12/2012)

8.5.3 Conceptual Model for U Ontology

In the conceptual model, the key concepts of the domain that have been identified earlier are structured to show their relationships with each other and specify restrictions in their relationships. As mentioned by Guizzardi (2006), the structural representation and its components remain independent regardless of the formalization language and approach which is used to serialize the conceptual model. Therefore, by using the terminology introduced in Table 8.1 and grouping the concepts that are related to each other, the following sub (conceptual) models are presented.

8.5.3.1 Ontology Usage Analysis sub-model

The sub-model shown in Figure 8.5 relates the core concepts of the ontology usage analysis domain model. *OntologyUsageAnalysis* is a high level concept that represents the ontology usage analysis domain and its activities. As shown in Figure 8.5, it comprises three main components *Source*, *Measure* and *OntologyUsage* which are explained in the following sub-sections:

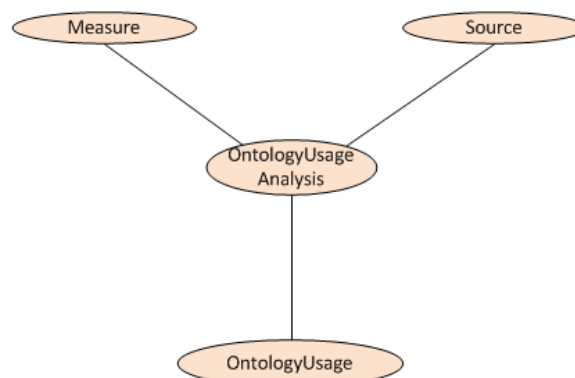


Figure 8.5: Ontology Usage Analysis sub (conceptual) model.

Source: Source refers to the data source that is being used by OUA to obtain the required inputs for its analysis. Source is an abstract entity that generalizes the data sources providing the ontology usage related information to the OUSAF. Two specialized sources which are Dataset and Streaming are included in the domain

model as the potential sources of the input data. Dataset represents the repository holding the Semantic Web data crawled from the Web and Streaming represents the continuous crawling which provides the input data to OUA through RDF data stream processing (Martínez-Prieto et al., 2012).

Measure : Measure represents the different dimensions from which the use of ontologies are measured. Measure is a generalized concept which can generalize any of the dimensions that are applicable to ontology usage analysis. In this model, three dimensions are used to analyse ontology usage and their corresponding concepts are UsageDim (defined in Section 6.4.3 and 7.3.2), RichnessDim (defined in Section 7.3.1), and IncentiveDim (defined in Section 7.3.3). UsageDim represents the "usage" dimension in which ontology usage is measured; RichnessDim represents "richness" which measures the ontology component's structural characteristics; and IncentiveDim represents the incentive dimension which captures the commercial advantages available to data publishers due to the use of ontologies.

OntologyUsage : This is the central or pivotal concept of the ontology usage analysis domain. As indicated by its name, it represents the overall discipline in which ontologies and their components are analysed. OntologyUsage further represents the usage analysis of ontology components. It conceptualizes usage analysis for each component through specialized concepts as shown in Figure 8.6. ConceptUsage represents the different aspects from which a concept (here concept represents the class of ontology that is being analysed) of a given domain ontology is analysed. The ConceptUsage sub-model is discussed in the next sub-section. RelationshipUsage represents the usage analysis of the object properties and similarly, AttributeUsage represents the use of attributes defined in the domain ontology.

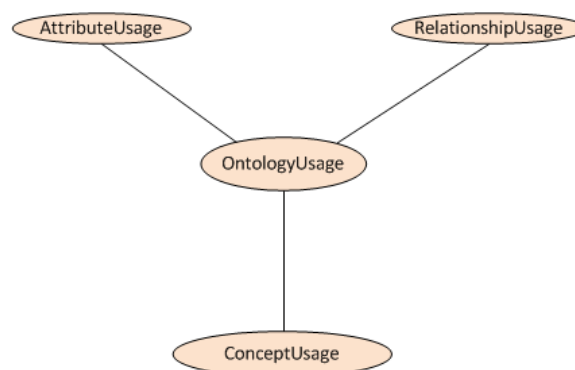


Figure 8.6: OntologyUsage and related concepts.

8.5.3.2 Concept Usage sub-model

As mentioned previously, `ConceptUsage` (defined in Section 6.4.3) is the concept which represents all the aspects from which a concept is being analysed. The `ConceptUsage` sub-model, depicted in Figure 8.7, comprises four aspects that are analysed for a given concept: `Vocab`, `Relationship`, `Attribute` and `LabelUsage`. Each of these concepts is discussed as follows:

Vocab: `Vocab` represents the different vocabularies that are being used to describe the entity (the instance of the concept). As a commonly required and recommended best practice, the entities (concept's instance) are described by using the relationships that are defined by the domain ontology and also the other ontologies/vocabularies that are common in the respective community. Therefore, in order to establish a comprehensive understanding, it is important to know what vocabularies/ontologies are being used by data publishers to semantically describe the entities. `Vocab` captures all such ontologies whose terms are used to describe the resource.

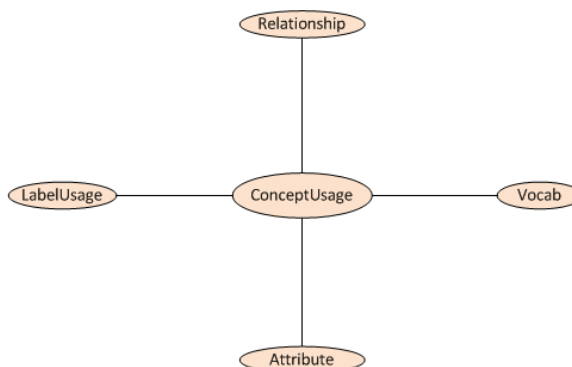


Figure 8.7: Concept Usage sub-model.

Relationship: `Relationship` captures the use of different object properties to semantically describe the entities. The used relationships can come from the domain ontology that is being analysed or from other ontologies as well. Therefore, it is important to learn about all the object properties that are used by different data publishers as this covers the entity relationships with other entities.

Attribute: `Attribute` captures all the datatype properties that are used to provide the attribute description of the entities. These are normally literal values which provide factual statements about entities. Therefore, `Attribute` captures all the data properties that are being used by data publishers to provide factual statements describing the state of the entity.

LabelUsage: LabelUsage (defined in Section 6.4.4) captures the use of different label properties that are provide the textual description of the entities. Label properties are normally used to provide human-friendly information about entities. In the OUSAF framework, label properties are categorised into two: Domain Labels (DL) and Formal Labels (FL). The Domain Label represents the use of label properties that are defined by the domain ontology that is being analysed and the Formal Label represents the use of label properties which are defined by the W3C-based vocabularies and are considered as standard labeling properties for providing textual information. Therefore, DomainLabel and FormalLabel are the specialized concepts of LableUsage to capture domain labels and formal labels, respectively.

8.5.3.3 KnowledgePattern sub-model

KnowledgePattern (defined in Section 6.4.5) captures the invariance in usage patterns across the dataset. It represents the presence of different triples that are frequently used by several data publishers. Knowledge Patterns comprise Path which has PathStep to represent each triple in the path. PathStep which represent a single triple comprises PathConcept and PathProperty to specify the concepts and predicates of triples, respectively, as shown in Figure 8.8. These are described as follows:

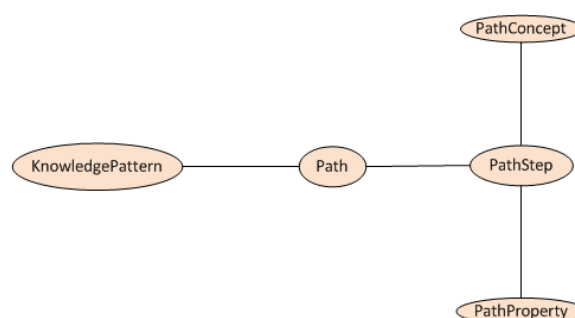


Figure 8.8: KnowledgePattern sub-model.

PathConcept: PathConcept represents the concepts of the entities that are present in a triple. This includes both the concept used as a subject and object. PathConcept is shown in Figure 8.9 which depicts the anatomy of the knowledge pattern. PathConcept captures the concept of the ontology that is being used to instantiate the subject and the object of the triple described using PathProperty.

PathProperty: PathProperty represents the predicate present in the triple which

forms the PathStep. PathProperty captures the object property that describes the subject by creating a typed relationship with another resource, as shown in Figure 8.9.

PathStep: PathStep represents the single triple that is included in the Path. PathStep is shown in Figure 8.9 by a dotted line containing the triple inside it.

Path: Path represents the unique sequence of triples (PathStep) linked together to describe a portion of the domain knowledge. Path is shown in Figure 8.9 by a dotted line which contains several path steps.

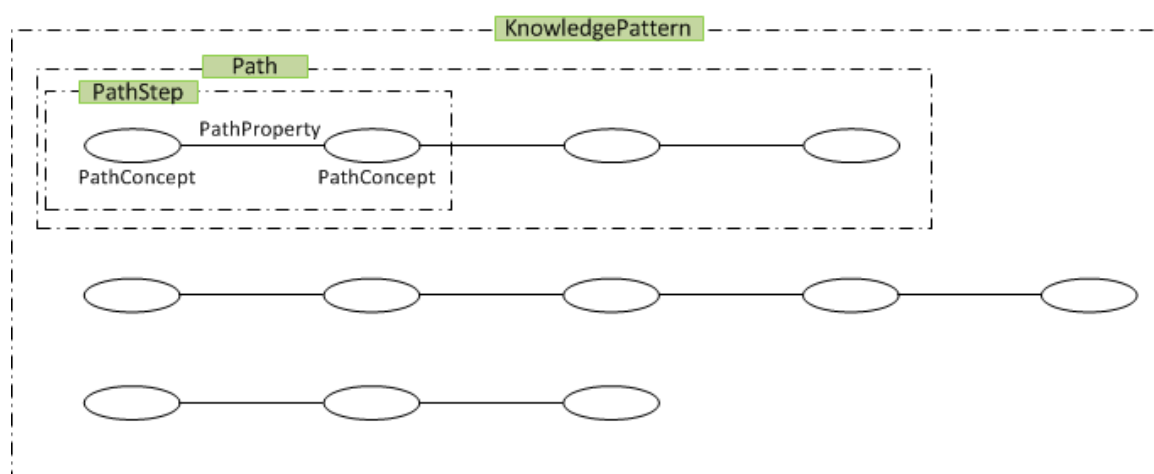


Figure 8.9: KnowledgePattern components.

8.6 Formalization Phase: Ontology Formalization and Integration

The formalization phase is the third phase in the development of the U Ontology in which the conceptual model is formalized using a formal modeling approach. This phase contains two set of activities: formalization of the conceptual model and integration with existing ontologies (for reusability). These two activities are described in the following sub-sections.

8.6.1 Formalization of Conceptual Model

As mentioned in Section 8.3, UML is considered an industry standard for modeling a conceptual model and provides a set of graphical notations to describe the model components. A class diagram is often used to formally represent the ontological model which, in the case of ontologies (see Figure 8.3), comprises concepts, attributes and

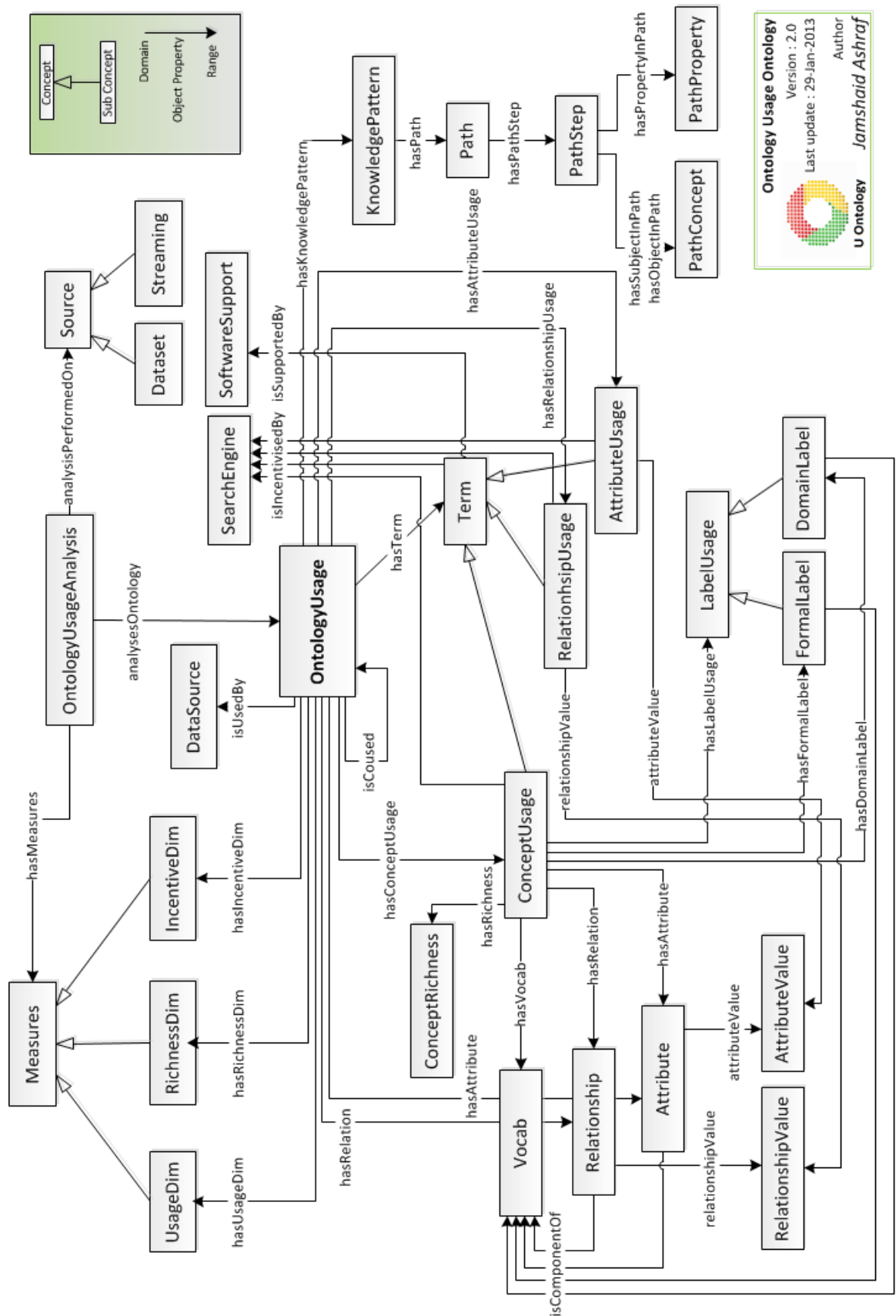


Figure 8.10: U Ontology Overview (V2.0).

object properties shown through the relationships between concepts. The formalized conceptual model of the U Ontology is shown in Figure 8.10. In order to avoid cluttering and for brevity in the conceptual model diagram, the class diagram shows only the concepts in Figure 8.10 which, in normal conversion, comes with attributes. The concepts, attributes and their relationships with other concepts are presented in this section to provide an overview of the U Ontology.

U Ontology vocabulary can be classified into the following groups, based on their objective and functionality: "*Concepts to represent Analysis Metadata*", "*Core Concepts to represent Ontology Usage Analysis*", and "*Concepts to represent Knowledge Patterns*". For each group, the constituent concepts, attributes and the relationship with other concepts are presented in the following sub-section.

It is important to note that the word "concept" will be used here in two contexts; first, to represent the concept defined in the U Ontology and second, to refer to the concepts of the domain ontology being analysed. In order to avoid homonymity, "Concept/concept" will be used to refer the U Ontology concepts and CONCEPT will refer to the concepts of the domain ontology being analysed. Similarly RELATIONSHIP, ATTRIBUTE refers to the object properties and data properties of the domain ontology and where required, TERM is used to refer to ontology components.

8.6.1.1 Concepts to represent Analysis Metadata

This group of concepts represents the portion of the U Ontology conceptual model which deals with concepts pertaining to the analysis of metadata which are high level concepts of the U Ontology and is central to `OntologyUsageAnalysis`. These concepts which represent the ontology usage analysis domain creates three relationships with `Measure`, `Source` and `OntologyUsage` concepts through `hasMeasure`, `analysisPerformedOn` and `analysesOntology` respectively, as shown in Figure 8.11. As mentioned in the previous section, usage analysis is measured from three dimensions, therefore `Measure` has three sub-concepts: `UsageDim`, `RichnessDim` and `IncentiveDim` which are linked with `OntologyUsage` through `hasUsageDim`, `hasRichnessDim` and `hasIncentiveDim` relationships, respectively. `Dataset` and `Streaming` are the sub-concepts of `Source` to identify the data source that is being used to perform usage analysis.

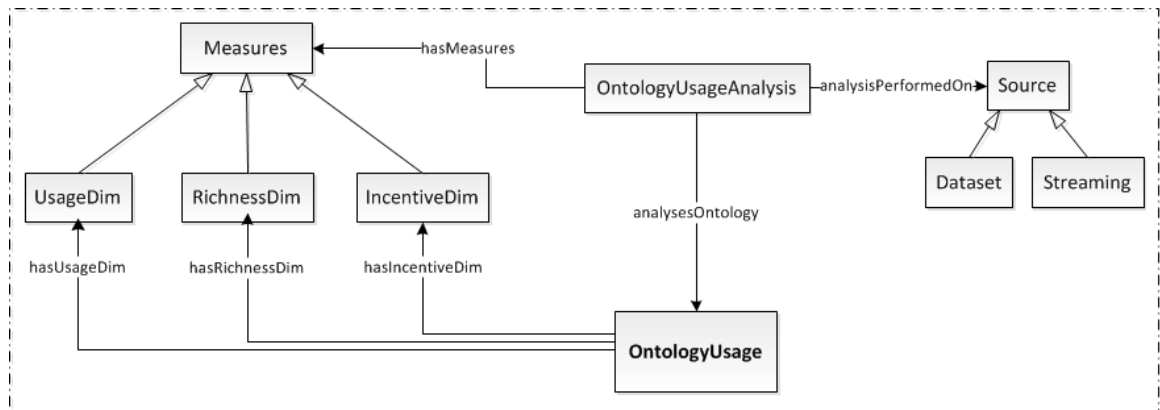


Figure 8.11: Concepts describing Analysis Metadata.

8.6.1.2 Core Concepts to represent Ontology Usage Analysis

This group of concepts represents the core concepts of the U Ontology which covers the usage analysis portion of ontology usage analysis. These concepts are divided into three areas each represented by a dotted rectangular box differentiated through colour, as shown in Figure 8.12. Each is discussed as follows:

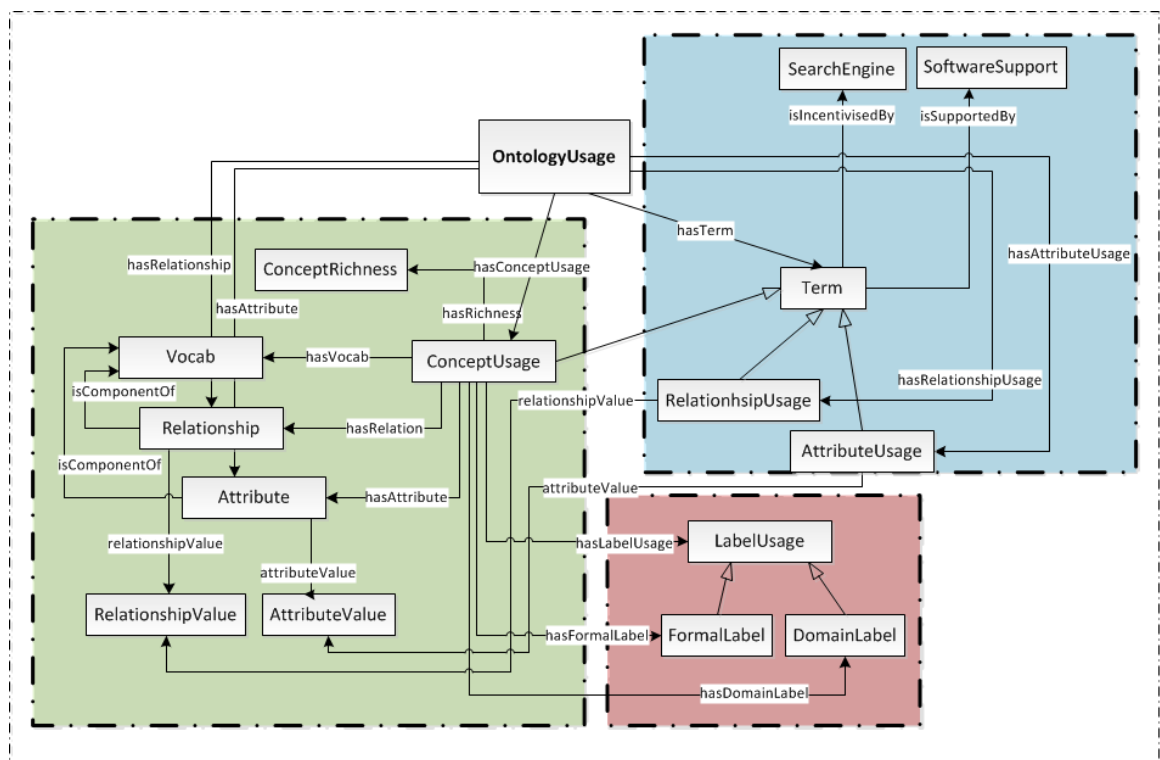


Figure 8.12: Concepts describing ontology components (concept, relationship, and attribute) analysis

- The left dotted area (in light green) covers the concept that analysis uses different CONCEPTS. As mentioned previously, CONCEPTS are analysed from

different aspects (i.e. CUT) therefore, there are a number of concepts linked with `ConceptUsage`. `ConceptUsage` represents the `CONCEPT` and is linked with `Vocab`, `Relationship`, `Attribute` through `hasVocab`, `hasRelation`, and `hasAttribute`, respectively. `ConceptUsage` is linked with `ConceptRichness` to specify the richness value which quantifies the structural characteristic of `CONCEPT`. Likewise, `RelationshipValue` and `AttributeValue` quantifies the richness value for object properties and attributes of the domain ontology, respectively.

- The right bottom dotted area (in pink) covers the use of labeling properties for `CONCEPT`. `DomainLabel` and `FormalLabel` are the sub-concepts of `LabelUsage` and capture the use of the domain ontology defined and the W3C-based vocabularies label properties, respectively. `ConceptUsage` is linked with `FormalLabel` and `DomainLabel` using `hasFormalLabel` and `hasDomainLabel` properties to specify the use of different labeling properties for `CONCEPT`.
- The right top dotted area (in light blue) covers concepts related to the usage measurement of `RELATIONSHIPS` and `ATTRIBUTES` of the domain ontology. It also includes concepts related to Incentive measurements for the `TERMS` of the domain ontologies. `Term` concept represents `TERMS` and subsumes `ConceptUsage`, `RelationshipUsage` and `AttributeUsage`. Further `Term` is linked with `SearchEngine` and `SoftwareSupport` through `isIncentivisedBy` and `isSupportedBy` properties, respectively to specify which means are used to measure the incentives (commercial benefits).

8.6.1.3 Concepts to represent Knowledge Patterns

This group of concepts represents the portion of the U Ontology conceptual model which deals with the representation of the knowledge patterns in the dataset, as shown in Figure 8.13. As discussed earlier, `KnowledgePattern` represents the prominent usage patterns that invariantly prevail in the dataset. `KnowledgePatterns` has several `Path` linked through `hasPath` to specify the unique instance of the usage pattern in the Source. A `Path` is comprised of several `PathSteps` which essentially represents a triple. In order to capture and represent the subject and object of `PathStep` (triple), `hasSubjectInPath` and `hasObjectInPath` links with `PathConcept` to specify the `CONCEPTs`. The predicate of `PathStep` is specified by `PathProperty` linked through `hasPropertyInPath`.

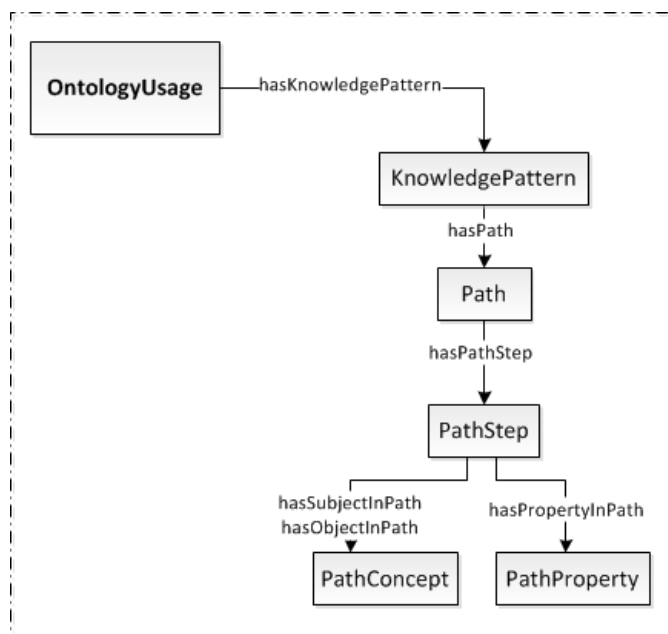


Figure 8.13: Concepts to represent Knowledge Patterns in dataset.

The second set of activities in the formalization phase is integration which is discussed in the next sub-section.

8.6.2 Integration with other ontologies

As discussed in Section 8.5.2, the two identified ontologies which conceptualize the domain relevant to OUA are the Ontology Metadata Vocabulary (OMV) and the Ontology Application Framework (OAF). OMV enables the specification of the metadata of ontology which includes the ontology conceptualization model (OntologyBase), ontology documentation (OntologyDocument), tools, language, methodology and organization involved in developing and maintaining ontology, whereas the OAF specifies the application areas in which ontologies are used and the roles ontologies play.

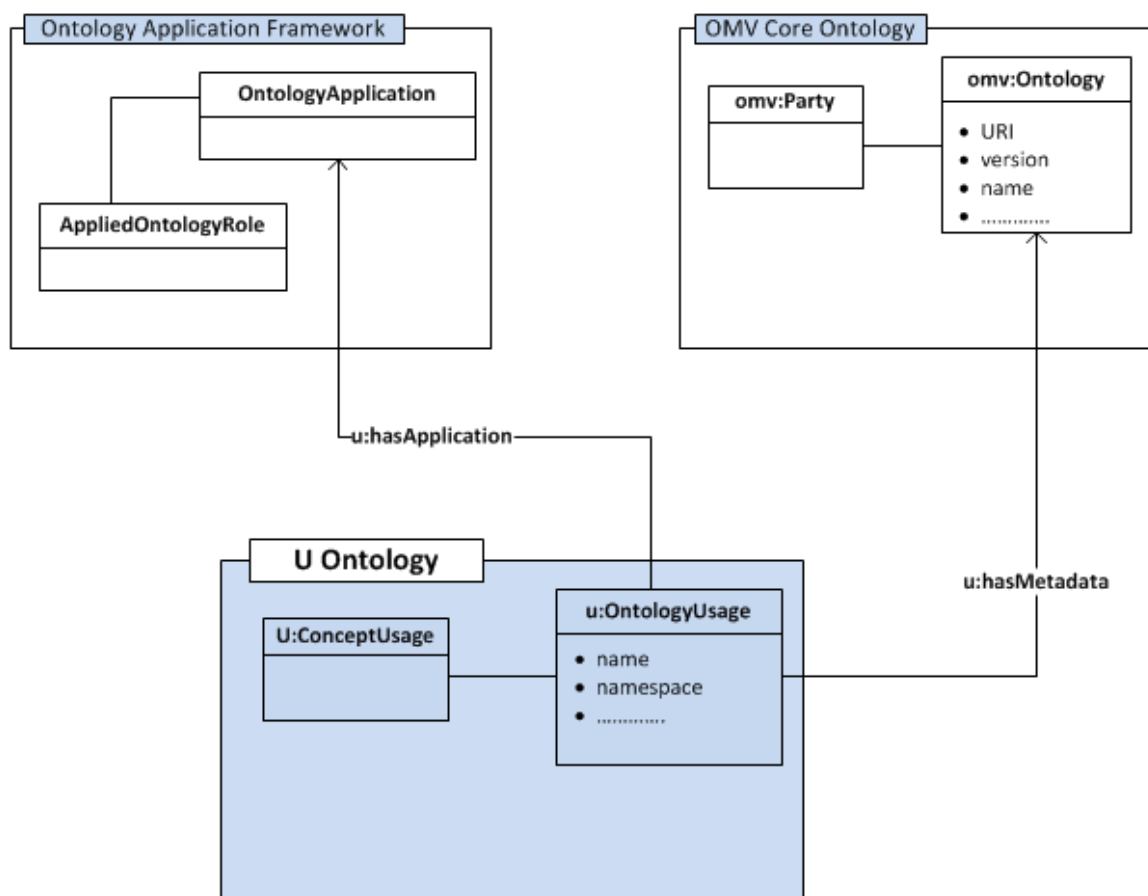


Figure 8.14: Integration of the U Ontology with other ontologies.

In order to integrate these two ontologies with the U Ontology to allow the reuse of the concepts defined in them, the U Ontology provides interlinking properties, as shown in Figure 8.14. The details are as follows:

- The metadata of the domain ontology being analysed is provided by linking the U Ontology's `OntologyUsage` concepts with `omv:OntologyDocument` through `hasMetadata` property. Instead of reinventing the concepts needed to describe the ontology metadata, the OMV ontology's concepts are used for that purpose.
- The Ontology Application Framework (OAF) is used for specifying the application areas in which the domain ontology that is being analysed is used. The U Ontology provides a property `hasApplication` to integrate the U Ontology's `OntologyUsage` with the `OntologyApplication` concept of OAF.

In addition to these two ontologies which are interlinked with the U Ontology, a few terms (URIs) from other common vocabularies/ontologies are used in the U Ontology as described in Table 8.2

Vocabulary	Dublin Core
Namespace	http://purl.org/dc/elements/1.1/
Term	title
URI	http://purl.org/dc/elements/1.1/title
Label	Title
Definition	A name given to the resource. Typically, a title will be a name by which the resource is formally known
Vocabulary	Dublin Core
Namespace	http://purl.org/dc/elements/1.1/
Term	description
URI	http://purl.org/dc/elements/1.1/description
Label	Description
Definition	An account of the resource. The description may include but is not limited to: an abstract, a table of contents, a graphical representation, or a free-text account of the resource
Vocabulary	RDFS
Namespace	http://www.w3.org/2000/01/rdf-schema
Term	label
URI	http://www.w3.org/2000/01/rdf-schema#label
Label	Label
Definition	Used to provide a human-readable version of a resource's name
Vocabulary	RDFS
Namespace	http://www.w3.org/2000/01/rdf-schema#
Term	comment
URI	http://www.w3.org/2000/01/rdf-schema#comment
Label	Comment
Definition	Used to provide a human-readable description of a resource
Vocabulary	FOAF
Namespace	http://xmlns.com/foaf/0.1/
Term	name
URI	http://xmlns.com/foaf/0.1/name
Label	Name
Definition	A name for something and is written in a simple textual string.

Table 8.2: Reused Terms

8.7 Implementation Phase: Ontology Implementation

The U Ontology is intended for use on the Web (based on Semantic Web Architecture) to enable users to access usage-related information about ontologies. Therefore, it is desired that the developed ontology should be able to make use of existing ontology tools such as OWL Reasoner and Triple store, and should be based on the formalism that is largely supported by the community. Based on the literature review on the formalism used for ontologies of a similar nature (such as GoodRelations (Hepp, 2008), DQM (Fürber and Hepp, 2011)), OWL DL expressivity is used for the U Ontology. Therefore, for the implementation of the U Ontology, OWL-DL syntax is used which comprises the following language elements:

- owl:Ontology
- owl:Class
- owl:ObjectProperty
- owl:DatatypeProperty
- rdfs:subClassOf
- rdfs:subPropertyOf
- rdf:datatype
- rdf:type
- rdfs:domain
- rdfs:range

The choice of the abovementioned language elements will allow users to use the OWL-DL syntax for RDFS elements. This encoding approach limits the ontology coding to RDFS elements which is the subset of closure of OWL DLP and RDFS-based reasoners can be used for inferencing (De Bruijn et al., 2005). As suggested in (Hepp, 2008) such an ontology implementation approach will allow "*the ontology to be used with other OWL-DL ontologies and knowledge bases without making the resulting ontology become OWL Full*". In encoding, the use of rdfs:domain and rdfs:range are used to facilitate the data creation, generation and population process by developing a

user interface and input form and should not be used to compute the inference closure by the repositories.

Certain decision are made pertaining to specifying the datatype properties of the U Ontology concepts. Particularly for concepts which represent the URIs of the domain ontology, CONCEPTS, RELATIONSHIPS, and ATTRIBUTES such as ConceptUsage, Relationships, Term the following datatype properties are used in the definition of each concept.

- name
- termURI
- description
- prefix
- label
- comments

termURI attribute and other datatype properties which represent the URIs of the domain ontologies are defined with datatype `xsd:anyURI` to allow the specification of CONCEPTs, RELATIONSHIPs and ATTRIBUTEs URIs as objects. For the encoding of the U Ontology, Protégé (Knublauch et al., 2004) is used as an ontology editor which provides all the necessary services needed for the construction of ontologies. Different reasoners provide plug-ins for Protégé which makes it easy for developer to validate the conceptual model and perform consistency checking to resolve discrepancies which arise during encoding.

An overview of the full U Ontology coding and a description of the elements is presented in Appendix A.

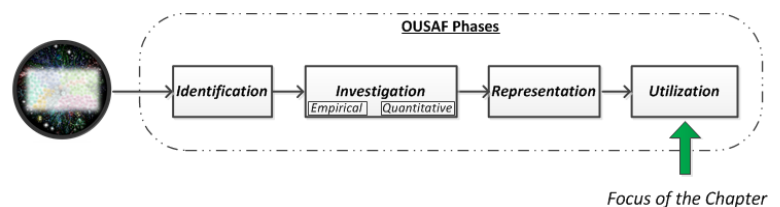
8.8 Conclusion

In this chapter, the Ontology Usage Ontology (U Ontology) is presented for the *Representation* phase of the OUSAF framework. For the development of the U Ontology, a customized development methodology is adopted based on the three existing methodologies, comprising *Specification*, *Conceptualization*, *Formalization* and *Implementation* phases. In the specification phase, motivational scenarios and competency questions are developed to define the scope and elicit the requirements of

different stakeholders. Based on the identified requirements, the domain knowledge controlled vocabulary is developed as part of the conceptualization framework. Ontologies that are relevant to ontology usage and which overlap with its domain knowledge are identified to be considered for reuse. Based on the terms defined in the controlled vocabulary, a conceptual model is presented to facilitate the formalization of the model. In the formalization phase, the conceptual model of the U Ontology is formalized using the Unified Modeling Language (UML). The models components such as concepts, relationships and attributes are discussed. Ontologies which were identified for reusability are integrated with the U Ontology to access their components. In the final implementation phase, using Protege ontology development tools, the U Ontology is encoded in OWL-DL syntax.

The next chapter focuses on the utilization phase of the OUSAF framework which is the implementation of the U Ontology to demonstrate the application of Ontology Usage Analysis.

Chapter 9 - Utilization Phase: Utilization of OUSAF Framework



9.1 Introduction

As mentioned in earlier chapters, the objective of this thesis is to propose a pragmatic solution for measuring and analysing the use of ontologies on the Web. To achieve this, a methodological approach is adopted and implemented in the form of an OUSAF framework. For the realization of the OUSAF framework, several models, methods, processes, and strategies are developed as discussed in Chapters 5-8, that provide the technical infrastructure for monitoring, measuring, and analysing the use of ontologies. Once they are developed, the next phase of the OUSAF framework is the utilization phase where the analysed usage results are made available to the users for them to be utilized and applied. This is achieved by using the core infrastructural components of the proposed solution, such as the collected *dataset*, *OUN-AF*, *EMP-AF*, *QUA-AF*, and *U Ontology*. These infrastructural components exhibit the practicability of the proposed solution and will be used to demonstrate the utilization of the ontology usage analysis for different types of users.

In order to demonstrate the utilization of the OUSAF framework in this chapter, a methodological approach is adopted which provides a systematic flow of activities and the interaction between different components to analyse the utilization. This methodological approach is presented in Section 9.2. Section 9.3 presents details

on the construction of the dataset that is used to demonstrate the utilization phase. In Sections 9.4-9.6, the utilization of the different phases of the OUSAF framework is presented. Section 9.7 summarizes the achievements of the utilization phase and Section 9.8 compares usage results from the OUSAF framework with existing approaches from the literature. Section 9.9 concludes the chapter.

9.2 Approach to demonstrate the utilization of the OUSAF framework.

As mentioned earlier, the objective of the utilization phase is to allow users to make use of the OUSAF framework and obtain the required information and insight regarding the use of ontologies. The different computational frameworks (i.e. OUN-AF, EMP-AF, and QUA-AF) developed for the OUSAF framework are accessed through the U Ontology which conceptualizes the domain of ontology usage analysis. Each computational framework – *which are, in fact, the contributions of the thesis* – performs certain operations to measure the usage of ontologies from different perspectives and generate analysis that can be used by the end users. In order to show how these computational frameworks will be accessed, the following points need to be addressed to form a methodological approach for utilization:

- A systematic approach to *demonstrate the utilization of each contribution*, i.e. OUN-AF, EMP-AF, and QUA-AF
- Qualitatively analyse the *usability* of the results obtained for each contribution and their *adequacy*.

By "demonstrate the utilization of each contribution", it is intended to show by performing a certain set of activities, how the required output is obtained for a use case and how this can be used by the specified users to achieve their specified goals (requirements). The term usability refers to how the results obtained from the OUSAF framework can be used further by the users and the term adequate indicates whether the results are sufficient to be useful. There are two observations from the underlined words; first, a methodological approach is required so that for each type of user (based on their requirements), the OUSAF framework is accessed and results are obtained; second, a qualitative discussion will take place to examine if the obtained results address the aim of the user. The first observation can be implemented by forming a systematic approach to demonstrate the utilization, and for the second observation, a discussion is presented to understand the usefulness and adequacy of the obtained

results. Figure 9.1 shows the infrastructural components, the flow of information, and the applicable observation to demonstrate the utilization of the OUSAF framework. The flow is as follows: based on the users' requirements (using use cases), the OUSAF framework is accessed to generate the output accessible through the U Ontology which is then assessed for usability and adequacy.

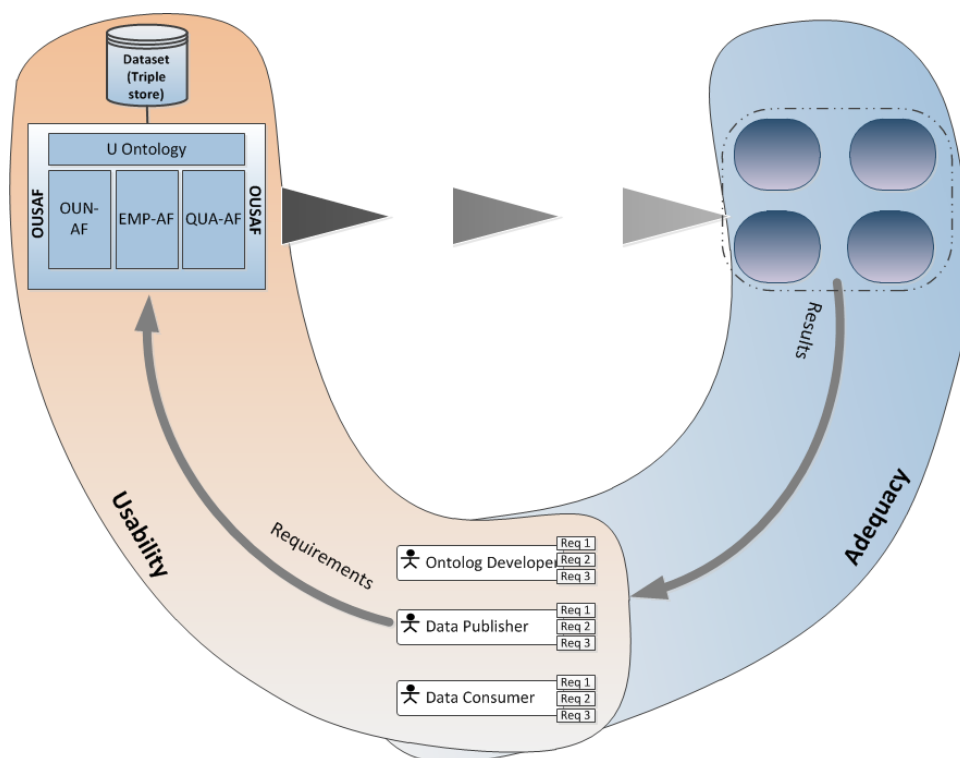


Figure 9.1: Approach for measuring Usability and Adequacy

In order to describe the methodological approach to demonstrate the utilization of the contributions, in the following sub-section, the different types of users are considered and their roles are discussed along with the criteria for the utilization analysis of OUSAF contributions.

9.2.1 Types of Users

To demonstrate the utilization of the OUSAF framework, three groups of users who will interact with OUSAF framework to obtain the required information are identified: They are:

- *Ontology developer:* This group of users are those involved in the construction of the ontologies and normally takes the form of ontology owner, domain experts, and ontology engineers. More or less, the primary function of these different users is to facilitate the construction and management of ontologies.

- *Data consumer*: This group of users are those who consumes the Semantic Web data that is published on the Web by using the described ontologies. These types of users are also known as application developers.
- *Data publisher*: This group of users are those who publish the Semantic Web data that is described using (domain) ontologies. These types of users are also known as semantic annotators and dataset publishers.

9.2.2 Contributions and Criteria for Utilization Analysis

The different contributions of the OUSAF framework and the criteria used to analyse the utilization of each contribution is presented in Table 9.1

Table 9.1: Contributions of OUSAF Framework

#	Contribution	Criteria	Description
1	A framework for ontology identification (i.e. OUN-AF)	Analyse the utilization, usability and adequacy of the obtained results	-specify the different types of users of the identification framework - specify the use cases for each user type - analyse the adequacy of the framework in implementing the use cases
2	A framework for empirically analysing the domain ontology usage (i.e. EMP-AF)	Analyse the utilization, usability and adequacy of the obtained results	-specify the different types of users of empirical analysis - specify the use cases for each user type - analyse the adequacy of the framework in implementing the use cases
3	A framework for quantitatively analysing the use of ontologies (i.e. QUA-AF)	Analyse the utilization, usability and adequacy of the obtained results	-specify the different types of users of quantitative analysis - specify the use cases for each user type - analyse the adequacy of the framework in implementing the use cases
4	Formalization of the conceptualized ontology usage analysis domain (U Ontology)	Evaluate the quality of the U Ontology	- specify the methodology which will be used for evaluation - specify the aspects from which ontologies is need to be analysed - the methods to analyse the aspects

Contribution 1, 2 and 3 will be analysed in Sections 9.4, 9.5, and 9.6, respectively. The fourth contribution, which is the U Ontology, will be evaluated in Chapter 10.

9.2.3 Components and Sequence of activities involved in Analysing the Utilization of each contribution of the OUSAF framework

The components and the flow of activities involved in the methodological approach followed for demonstrating the utilization of each contribution is shown in Figure 9.2 and its component description is discussed below.

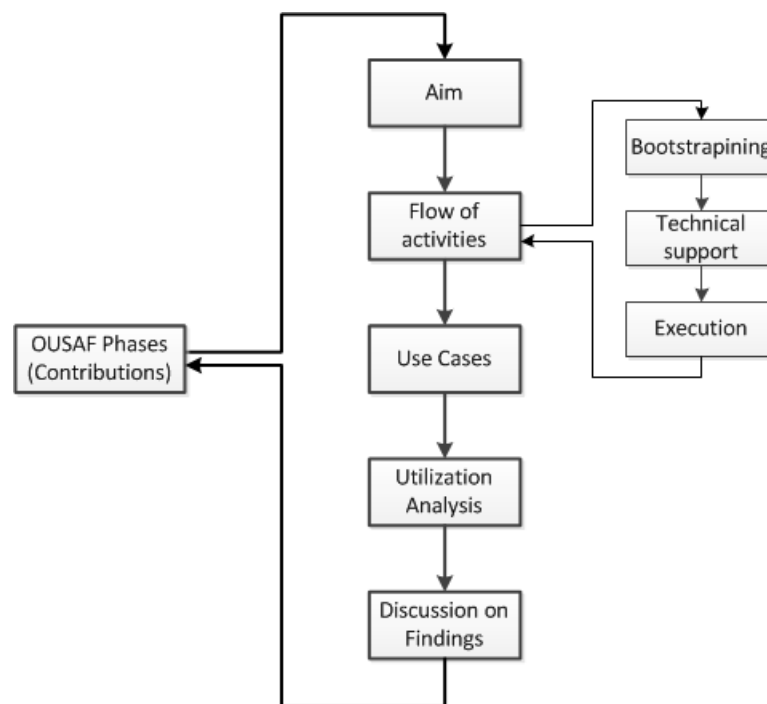


Figure 9.2: Components and Sequence of activities involved in analysing the utilization

Contribution: Contribution refers to the individual contribution made through this thesis. The contributions are listed in Table 9.1 which will be considered in this chapter to analyse the utilization. The process of analysing each contribution involves several steps which are explained below.

Aim: Aim specifies the purpose of the contribution and how it impacts the overall proposed solution. Before proceeding with the utilization analysis steps, the aims of the contribution and the stakeholders (users) who are interacting (or being impacted) with the contribution need to be specified. The stakeholders include both the human user and the machine user because certain contributions are equally accessible to both type of users.

Flow of activities. Flow of activities specifies the sequence in which the activities need to be performed in order to observe the benefits achievable through the contribution. This helps to communicate the steps, activities, and their sequence to obtain the desired results. This component comprises several subcomponents which are described below:

- **Bootstrapping:** Bootstrapping refers to the set of activities that are needed to be performed before analysing the particular contribution. This involves all the technical and non-technical arrangements required to set up the environment to demonstrate the contributions. Generally, it involves pre-processing the input data into a format that is accessible and usable for the respective framework.
- **Technical support:** Technical support refers to the activities involved to provide the computational arrangement for each contribution. This provides the infrastructural level services to integrate different components of the framework which implements the respective framework/solution of the contribution. It involves data manipulation, data structure and access to the dataset.
- **Execution:** Execution refers to carrying out the computational task to implement the contribution. Generally, it involves the execution of the framework to generate output that will be then analysed.

Use Cases: A use case scenario represents the series of actions that need to be carried out to address the specific requirement. Through a use case scenario, user requirements are highlighted and using one of the possible approaches, output which addresses the requirement is obtained.

Utilization Analysis: Utilization analysis refers to the activity in which the contribution in question is analysed against the use case. The output achieved through the contribution and the requirements extracted from the use case are analysed to assess the usefulness and adequacy of the obtained results to the user.

Discussion of Findings: Findings represent the conclusive observations made about the specific contribution. It provides a discussion on the obtained results in order to help in summarizing the utilization analysis of the contribution.

In next section, the dataset used to analyse the use of ontologies is described.

9.3 Dataset for demonstrating the utilization of OUSAF

To conduct the empirical study on the Semantic Web data and analyse the use of ontologies and a specific (focused) domain, a dataset is built to serve as a sufficient representative sample of the semantically annotated structured data currently published on the Web. A hybrid crawler is developed for this purpose which crawls the Web based on the specified parameter and populates the data repository (i.e. triple store). This section describes the design and implementation of the hybrid crawler which collects the snippets of the structured data that are embedded in the HTML pages by publishers to provide machine-readable information. Using the hybrid crawler which has crawled approximately 5.2 million document (mostly HTML pages), 480 million triples¹ are loaded to the triple store to be used for the analysis of domain ontologies. The specific requirements of the data, crawler, its specifications and the implementation of the crawler are described below. This collected dataset will be used in the remaining section of this chapter to demonstrate the utilization of the OUSAF framework.

9.3.1 Data Requirements

The obvious requirements of the data are that it has to be Semantic Web data described using the RDF data model. The common practice of the community is to publish RDF data using Linked Data principles ([Heath and Bizer, 2011](#)) and recommended best practices ([Dodds and Davis, 2010](#)) and make them available in the form of a dump file for download. As reported in ([Jain et al., 2010](#)), most of the datasets which are included in the LOD cloud either make no use of ontologies or minimal, which makes the RDF data available as part of the LOD cloud of limited interest due to the shallow representation of ontologies. The trend in the use of domain ontologies on the Web gained momentum when the incentives (Section 7.3.3 for more detail) were available to data publishers in the form of improved visibility in the search engines and applications were being developed to take advantage of the presence of explicit semantics. Therefore, to collect the dataset that is primarily annotated using domain ontologies to provide Semantic Web data over the Web, a hybrid crawler needs to be developed. However, the requirement for the crawler which considers the Semantic data annotated with ontologies has some unique requirements from other crawlers.

¹in fact triples were converted to quad to add the context of the triple which is discussed in subsequent sections

Generally, crawlers can be grouped into the following categories (Hogan, 2011a):

- *Topic-focused crawling* : In this category, hyper links () and anchor text () are used to identify pages similar to the topic, based on string matching and link analysis algorithms. Such crawlers are proposed in (Chakrabarti et al., 1999; Almpandis et al., 2007).
- *Ontology-Focused Crawling*: In this type of crawler, ontologies are used to match the concept based on the terms defined in the ontology vocabulary. Examples of such crawlers are proposed in (Ehrig and Maedche, 2003; Dong et al., 2008).
- *Learning-focused crawling*: In this type of crawler, machine learning techniques are employed to learn about the relevant links and draw similarities among pages. Examples of such crawlers are proposed in (Pant and Srinivasan, 2005; Richardson et al., 2006; Batsakis et al., 2009)
- *Semantic Data-focused Crawling*: In this type of crawler, different techniques are used to focus on Semantic Web data that is published on the Web. Such crawlers crawl the pages (or documents) that are made available on the Web by describing information using the RDF data model. Examples of such crawlers are proposed in (Decker et al., 1999; Ding et al., 2004; Dodds, 2006; Harth et al., 2006).

The first three types of crawlers focus on crawling Web pages (or documents) based on their similarity with the topic (subject) of the pages. However, each type has a different approach toward deciding which page to consider and how to route the crawling procedure, like crawl-by-depth or crawl-by-breadth. The second type of crawler (i.e. Ontology-Focused Crawling) makes use of ontologies but does not necessarily operate on Semantic Web data as such. In such an approach, ontologies are used to find neighbouring and similar concepts matching the topic by allowing the crawling to expand to similar related concepts based on the ontology conceptual model. The last type of crawler (i.e. Semantic Data-focused Crawling) is focused on crawling the RDF documents which are published on the Web. These types of crawlers apply a specific filter to consider only those documents that match the criteria, such as MIME type and Content-type.

As mentioned earlier, not all the Semantic Web (RDF) data published on the Web uses domain ontologies to describe information and more emphasis is placed on publishing structured data on the Web with or without the use of explicit semantics. Therefore, merely considering RDF data which is normally made available in the form of dump files, does not provide a fair representation of the structured data that is

annotated using domain ontologies. Therefore, the requirements of the data to be considered for measuring and analysing the use of domain ontologies are as follows:

- Semantic Web data that is based on the RDF data model
- Semantic Web data that is described using domain ontologies
- Semantic Web data that is published either as an RDF document or embedded within HTML pages

A crawler was developed by extending (Isele et al., 2010) to crawl RDF data. However, the results were not encouraging as most RDF documents did not use domain ontologies, except for the use of W3C-based vocabularies such as RDF, RDFS and a few constructs of OWL. The other two vocabularies which had a reasonable presence in RDF documents was FOAF and DC as both are considered well established to provide a textual description of the resources. Another observation was that there is minimal use of out-bound links to external documents (or resources). In light of these observations, a new crawling strategy was devised to focus on the Web pages that have semantically annotated structured data embedded in them. The detail of this strategy is discussed in next section.

9.3.2 Data Collection Strategy

Initially, a crawler was implemented by extending the LDSpider (Isele et al., 2010) but the collected data was not interesting as only 8.52% of the 1.8 million triples were described using authoritative ontologies, excluding W3C-based vocabularies. Based on the statistics obtained through the crawler implemented by extending. To obtain a dataset that addresses the abovementioned requirements, a new strategy was devised to customize a crawler that is capable of collecting the required Semantic Web data. In our previous study (Ashraf et al., 2011), it was observed that the new publishing trend is to add Semantic Web data using the RDFa standard which allows adding RDF snippets within exiting HTML pages. In previous research, it was observed that 90% of Semantic data is published using RDFa when it is embedded within existing Web pages and annotated with ontologies.

Based on our experience and considering the recommendation proposed by Thelwall and Stuart (2006), a data collection strategy is devised as described below.

- Using Semantic Web search engines such as Swoogle, Watson, Sindice, extract the list of domain names (pay-level-domain (PLD)) that are publishing data annotated using ontologies to generate seed URIs.

- Instead of focusing on the PLD which provides `Content-type:application/rdf+xml`, also consider `application/xhtml+xml` in order to include HTML documents which have AN RDFa snippet embedded in them.
- Exclude web pages in which structured data is embedded using A non-RDFa standard such as `microdata`² and `microformats`³.
- Exclude URI schemes from the seed URI list and a crawler process which includes *ftp*, *telnet*, *maitto* and *file*.
- Exclude digital resources such as *image*, *pdf*, and *cvs* files.

Based on the abovementioned guidelines, the data collection strategy is formed and is used for the data collection process which is discussed in the next subsection.

9.3.3 Data Collection Process

The data collection process specifies the steps and components involved in the collection of data using a hybrid crawler. Figure 9.3 shows the components involved in the overall process of data collection. The role of each component is explained below.

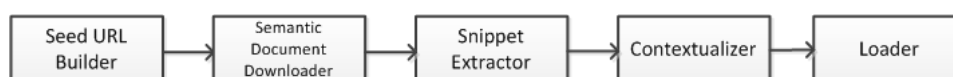


Figure 9.3: Components involved in data collection

The crawler proceeds based on the seed URIs collected through Seed RUI Builder which is responsible for preparing the seed URI to initiate the crawling process. From the seed URI, the RDF document (or HTML page) is retrieved to obtain the contents of the URIs. The obtained contents are then parsed to transform them into the required format before specifying the context of the retrieved RDF document using the contextualization phase. Afterwards, the contextualized content is loaded into the triple store for analysis.

9.3.3.1 Seed URL Builder

The crawler operates on a list of unvisited URLs which is known as *frontier*. The list (of URLs) is initialised with seed URLs which are often collected through another

²<http://www.w3.org/html/wg/drafts/microdata/master/>; retr. 14/12/2012

³<http://microformats.org/>; retr. 12/12/2012

program or manually supplied. The quality of the data retrieved by the crawler depends on the quality of the seed URLs. The role of the Seed URL Builder is to provide a list of URLs to initiate the crawl and specify the frontiers. To obtain high quality URLs to provide the frontier for the first round, different semantic search engines are accessed to retrieve the URLs of the websites (data publishers) publishing data using domain ontologies. Two semantic search engines, namely Watson (d'Aquin et al., 2007) and Sindice (Tummarello et al., 2007) and one traditional search engine i.e. Google is accessed using their APIs to obtain a list of URLs, as shown in Figure 9.4. In Google, to retrieve the RDF document filetype:rdf attribute of advanced search is used to narrow the search to only documents with the specified extension. To specify the quality, the number of namespaces defined in the RDF document (HTML pages), aside from W3C-based vocabularies, are measured and a weight $Seed_w$ is specified.

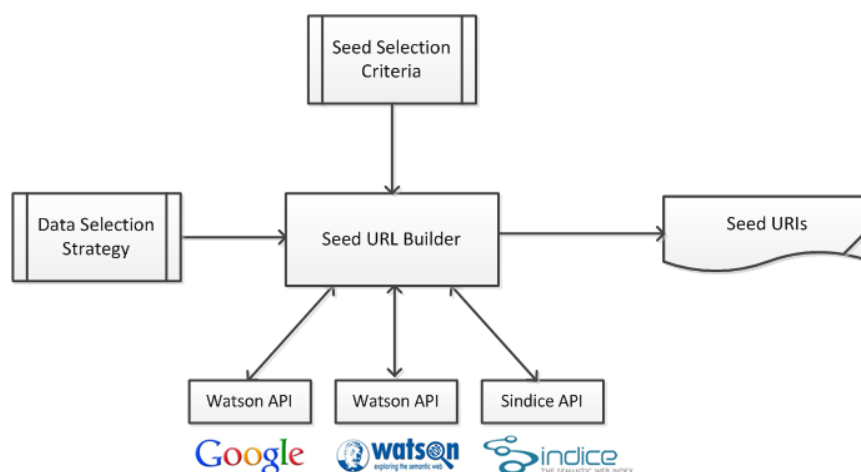


Figure 9.4: Seed URL Builder

9.3.3.2 Semantic Document Downloader

Following the general flow (Pant et al., 2004) in each round, the URL is picked from the frontier to retrieve the document corresponding to the URL using the HTTP request. The retrieved document is then parsed to obtain the content and the links to the external documents. These links are then evaluated for inclusion in the frontier, based on their quality value.

9.3.3.3 Snippet Extractor

Snippet Extractor extracts the RDFa snippets from the HTML pages and transforms the snippet into an RDF/XML-based RDF document. So, Snippet Extractor, using the parser retrieves the content of document and extracts triples annotating the

information. Any23⁴ and RDFaDistiller⁵ services are used to extract and transform the triples into RDF/XML serialized format.

9.3.3.4 Contextualizer

From a data management point of view, the context of the retrieved documents and their provenance details need to be added to the extracted RDF graph. For this purpose, the Named Graph (Carroll et al., 2005) approach is used to convert the triple into quads by adding a new resource specifying the context of the transformed graph.

9.3.3.5 Loader

Loader loads the quads into the triple store for usage analysis. Provenance details such as date and time when the data was collected, the source origin detail such as the PLD and the original data format is gathered.

9.3.4 Crawling

Figure 9.5 shows the sequential flow of the crawler implemented to collect Semantic Web data to measure ontology usage. The crawler initiates by populating the Seed URLs. The URLs which need to be visited are called frontiers and in one round, these frontiers are covered. For each URL, the crawler obtains the URL and retrieves the robots.txt file to ensure the required politeness in the crawling process. Filter criteria (which is mentioned in data collection strategy) is used to decide how to fetch the RDF documents (HTML pages) from the PLD. If the required page is allowed to be fetched, it is downloaded from the Web to extract its content. Since the web pages with RDFa snippets embedded in them are crawled, the RDF triples are extracted using RDFa parsers. From the extracted RDF/XML graph, URLs (the resources URI) referring to external resources are evaluated and enqueued to the seed URL list. The parsed RDF/XML graph is then contextualised by converting the triple into a quad and provenance details are added before loading the graph to the triple store.

⁴<http://any23.org/>; retr. 10/01/2013

⁵<http://www.w3.org/2012/pyRdfa/>; retr. 12/01/2013

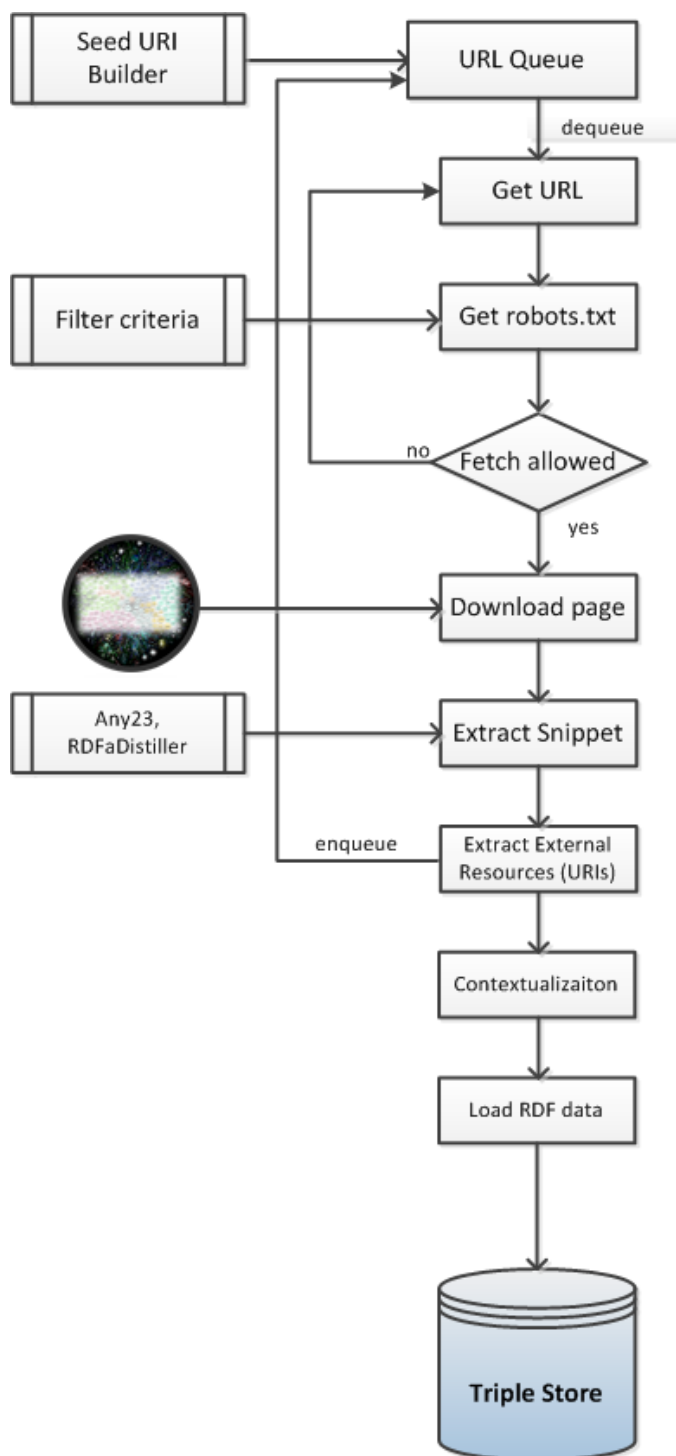


Figure 9.5: Basic Flow of the Crawler's Activities

A crawling policy was developed to run the crawler without adding unnecessary load on the server from which the Semantic Web data is being crawled. First, the restriction and permission outlined in `robots.txt` file are strictly observed and in case not published by the server, incorporate politeness in crawling as suggested by [Thelwall and Stuart \(2006\)](#). Regarding fetching the multiple pages from the

same PLD, three fetch requests per second were used and after 1000 requests, it is paused for 10 seconds in an attempt to ensure the server was not over busy with the fetching routine. For transforming the extracted RDFa snippets to RDF/XML graph, REST-based services are used and only the final RDF graph was loaded to the triple store. While accessing the Any23 and RDFaDistiller REST based services, a delay of 1 second was applied to not overload the servers hosting the service. The hybrid crawler was developed based on LDSpider and deployed on an ordinary machine (PC) Intel Core 2 with 2GB RAM.

9.3.5 Dataset statistics

In order to obtain the required dataset to measure ontology usage analysis, a hybrid crawler was developed based on LDSpider and deployed on an ordinary machine (PC) Intel Core 2 with 2GB RAM. The data was collected over a period of one month in different intervals by taking 186 machines hour in total. During the crawl, using 12,000 seed URLs, 480 quads are loaded from 5.2 million documents (90% HTML pages). The first two Linked Data principles (see Section 1.2 for more detail) require the use of HTTP URI to uniquely name resources and make them accessible through HTTP request as defined by RFC3986⁶. In URI lookup, 79.27% of URIs returned the 200 OK code which is the standard response for successful HTTP requests. For some, may it come as a surprise because normally datasets in the Semantic Web are available through redirect, but in the case of the e-Commerce web of data, most of the structured information is published using RDFa; embedded within HTML documents, hence making it available through a standard HTTP request. The 5XX code represents a server error and 1.22% of the URI returned the 5XX code. 18.05% of URIs returned the 404 Not Found code which means that the requested resource could not be found but may be available again in the future. 1.46

⁶<http://labsapache.org/webarch/uri/rfc/rfc3986.html>

Table 9.2: Content Type of HTTP Response

Content Type	% of documents
text/html; charset=iso-8859-1	7.32
application/octet-stream	0.24
application/rdf+xml	22.93
application/xml	0.73
text/html	22.20
text/html; charset=UTF-8	40.73
text/plain	2.44
text/xml	2.20
text/xml; charset=UTF-8	1.22

Using the approach presented in Section 9.2, the utilization of the OUN-AF, EMP-AF, and QUA-AF is presented in following subsequent sections, respectively.

9.4 Utilization of the Identification Framework (OUN-AF)

9.4.1 Aim

The aim of the identification phase is to identify the use of different ontologies and the interlinking between them based on their instantiation. This means, at the instance level, what different ontologies are being used to semantically describe domain specific entities. Also, it identifies the usage patterns that are prevalent across different data publishers and similarly, the ontology co-usage patterns between different ontologies. To obtain the abovementioned insight, the OUN-AF framework is proposed and its implementation is discussed in Chapter 5.

9.4.2 Flow of Activities

The OUN-AF framework is based on the affiliation network which comprises two sets of nodes, namely ontologies and data source. OUN is constructed based on the data sources and ontologies that are present in the dataset. OUN is used to measure the use of ontologies by different data sources and also identify the co-affiliation which exists between different ontologies based on their co-usage in describing information of a data source. The projection approach is used to transform the two-mode network to a one-mode network and measure the co-affiliation factor.

9.4.3 Use Cases

The three types of users are defined in Section 9.2.1, and based on their function and role, require a different set of information pertaining to ontology identification. For the identification phase, the information requirements of each type of user is described, which will be then used to demonstrate the utilization of the framework.

Ontology developer's requirements

Req. 1) What is the level of usage of a given ontology?

Req. 2) Is the given ontology being used alone or along with other ontologies and if yes, what are these?

Data consumer's requirements

Req. 3) What ontologies are being used in a given domain?

Req. 4) What the data sources are using a given ontology to publish their information?

Data publishers' requirements

Req. 5) What cohesive groups of ontologies have similar usage?

The abovementioned requirements are used to analyse the utilization of the OUN-AF framework.

9.4.4 Utilization Analysis

9.4.4.1 Req. 1: What is the level of usage of a given ontology?

In this requirement, the user is interested in knowing how many data sources are using a particular ontology. This requires the return of a number of data sources (ontology users) which have used the ontology components (at least one term of the ontology) to describe the information published on the Web.

The Ontology Usage Distribution (OUD) metric (Section 5.7.1) of the OUN-AF framework measures the number of different data sources a particular ontology has. This measure is also represented in the U Ontology (Chapter 8) which conceptualises the domain of ontology usage analysis. The `OntologyUsage` concept has the `hasUsers` attribute which captures the value of the OUD metric. Figure 9.6 displays the SPARQL query which queries the U ontology to retrieve the usage of the given

(domain). The filter clause of the query represents the input of the user indicating the particular usage in which they are interested. In the query, the name, uri, prefix and the number of data sources using the given ontology (e.g. foaf ontology) are displayed.

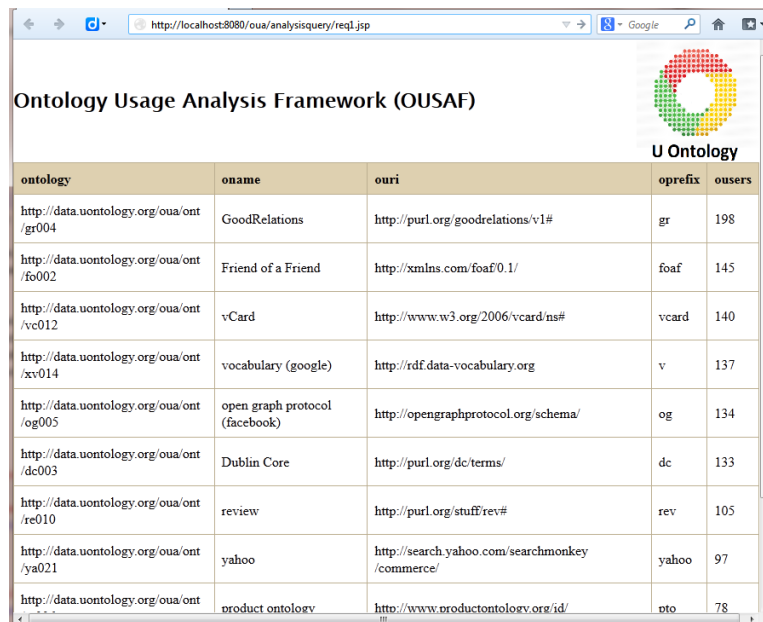
```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?ontology ?oname ?ouri ?oprefix ?ousers
4 WHERE {
5     ?ontology rdf:type uo:OntologyUsage;
6             uo:name      ?oname;
7             uo:uri       ?ouri;
8             uo:prefix    ?oprefix;
9             uo:hasUsers  ?ousers.
10     FILTER regex(?oprefix, "foaf", "i")
11 }

```

Figure 9.6: SPARQL query to display the list of ontologies and their usage.

From Figure 9.6, it can be seen that the U Ontology represents and captures the concept and attribute through which the user can obtain the information pertaining to the use of different ontologies in the dataset.



ontology	oname	ouri	oprefix	ousers
http://data.uontology.org/oua/ont/gr004	GoodRelations	http://purl.org/goodrelations/v1#	gr	198
http://data.uontology.org/oua/ont/fo002	Friend of a Friend	http://xmlns.com/foaf/0.1/	foaf	145
http://data.uontology.org/oua/ont/vc012	vCard	http://www.w3.org/2006/vcard/ns#	vcard	140
http://data.uontology.org/oua/ont/xv014	vocabulary (google)	http://rdf.data-vocabulary.org	v	137
http://data.uontology.org/oua/ont/og005	open graph protocol (facebook)	http://opengraphprotocol.org/schema/	og	134
http://data.uontology.org/oua/ont/dc003	Dublin Core	http://purl.org/dc/terms/	dc	133
http://data.uontology.org/oua/ont/re010	review	http://purl.org/stuff/rev#	rev	105
http://data.uontology.org/oua/ont/ya021	yahoo	http://search.yahoo.com/searchmonkey/commerce/	yahoo	97
http://data.uontology.org/oua/ont/pt001	product ontology	http://www.productontology.org/id/	pto	78

Figure 9.7: Result of SPARQL query shown in Figure 9.6

Figure 9.7 displays the results of the query which appeared in Figure 9.6. In order to display other ontologies aside from *foaf*, while executing the query, the FILTER clause was removed. The results display other ontologies and for each ontology, its object reference, name, namespace URI, prefix and the number of users (data publishers who have used the ontology) is given. It can be seen that there are 134

different data publishers in the data set who have used the Open Graph Protocol vocabulary (5th row).

This insight regarding the usage of different vocabularies help users (particularly ontology developers) to know the present adoption level and uptake of ontologies on the Web. Ontology developers can learn from these ontologies which have a good adoption rate by studying their structural and semantic characteristics and applying these to their own ontology development process. For data consumers, this provides a list of well adopted ontologies to consider for their own semantic annotation needs.

9.4.4.2 Req. 2: Is the given ontology being used alone or with other ontologies and if yes, what are these?

In this requirement, the user would like to know the other ontologies that are being co-used with the given ontology. This helps the user in identifying the ontologies that cover the concept related to their domain and are being frequently used by the community (Semantic Web data publishers).

The Ontology Usage Network (OUN) is two-mode network with relationships (edges) between distinct type of node. In the case of Req 2, it is necessary to know the relationships between the same types of nodes in order to know which ontologies are being co-used with a given ontology. Using the projection technique (Section 5.5.2), an ontology-to-ontology network is obtained which represents the relationships between ontologies. As explained in Section 5.8.4, in the projected one-mode network, two ontologies are linked only if both have been used by the same data source which shows their co-usability factor in the network.

The U Ontology provides the `isCoused` relationship which links two ontologies if they have an edge in the ontology-to-ontology one-mode (projected) network.

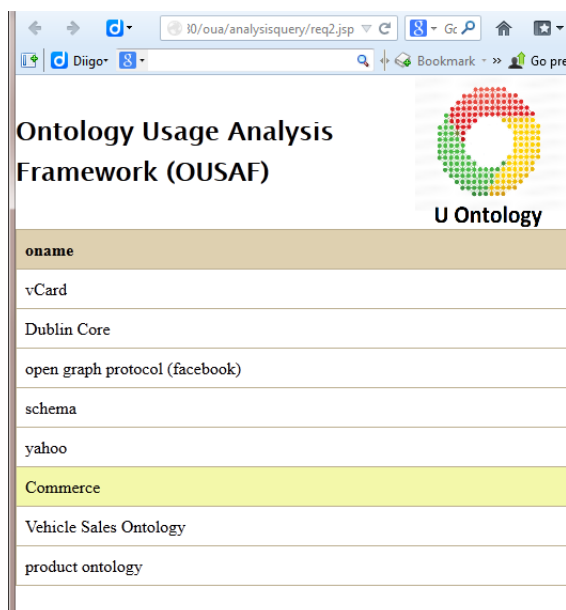
```
1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?oname
4 WHERE {
5     <http://data.uontology.org/oua/ont/gr004> rdf:type uo:OntologyUsage;
6         uo:isCoused ?coonto.
7     ?coonto uo:name ?oname.
8 }
```

Figure 9.8: SPARQL query to display the names of the ontologies being co-used.

Figure 9.8 shows the SPARQL query which lists the names of the ontologies that are being co-used with a given ontology. In the listing, `<http://data.uontology.org/oua/ont/gr004>` is the URI of the given ontology (e.g.

FOAF) and using `uo:isCoused` object property, the URIs of the ontologies which are being co-used are obtained and `uo:name` data property displays the name of ontologies.

The query listed in Figure 9.8 shows that the U Ontology model is able to capture information regarding the co-usability factor obtained by projecting OUN to an ontology-to-ontology network.



oname
vCard
Dublin Core
open graph protocol (facebook)
schema
yahoo
Commerce
Vehicle Sales Ontology
product ontology

Figure 9.9: Result of SPARQL query shown in Figure 9.8.

The query shown in Figure 9.8 requires all other ontologies which have been used along with a particular ontology i.e. "gr" to be displayed. Figure 9.9 shows the names of the ontologies which have been used by different data publishers to semantically describe e-Commerce-related information. Knowing what other ontologies are being used with a given ontology helps ontology developers to know what other ontologies are sharing the conceptual description related to the domain being captured by the given ontology. It provides data publishers with a list of ontologies they need to consider while deciding on the potential ontologies for their information annotation needs.

9.4.4.3 Req. 3: What ontologies are being used in a given domain?

In this requirement, the user is interested to know the different ontologies that are presently being used on the Web in a specific application area. This is one of the common requirements for all types of users because it provides high level but useful information regarding ontology usage. This requirement is closely matched with Req 1 (Section 9.4.4.1) however, here the user is interested to know all the ontologies that are being used in the dataset.

```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?ontology ?oname ?ouri ?oprefix ?ousers
4 WHERE {
5     ?ontology rdf:type uo:OntologyUsage;
6             uo:name      ?oname;
7             uo:uri       ?ouri;
8             uo:prefix    ?oprefix;
9             uo:hasUsers  ?ousers.
10 }

```

Figure 9.10: SPARQL query to display the name of the ontologies present in the dataset.

Figure 9.10 lists the query which retrieves all the ontologies that are being used in the dataset. The name, prefix, URI, and usage of all ontologies are obtained to provide the required information for the users.

This query is similar to the one shown in Figure 9.6 and the obtained result is similar as of shown in Figure 9.7.

9.4.4.4 Req. 4 : What data sources are using a given ontology to publish their information?

In this requirement, the user is interested in knowing about the different data sources (pay level domains) that are using the given ontology. The second set of nodes of OUN represents the data sources which have used the ontologies to describe information. In the U Ontology, the information about the different data sources is represented through the DataSource concept. The OntologyUsage concept is linked to the DataSource concept through isUsedBy relationships which allows different data sources which are using the given ontology to be specified.

```

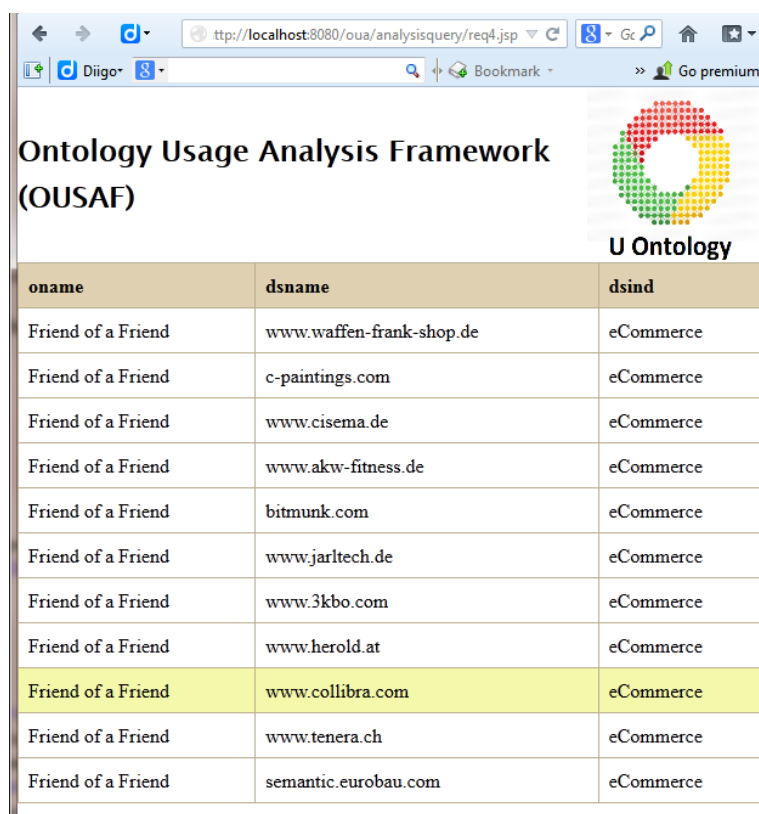
1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?oname ?dsname ?dsind
4 WHERE {
5     ?ontology rdf:type uo:OntologyUsage;
6             uo:prefix    ?oprefix;
7             uo:name      ?oname;
8             uo:isUsedBy ?ds.
9     ?ds      uo:name      ?dsname;
10            uo:industry  ?dsind.
11     FILTER regex(?oprefix, "foaf", "i")
12 }

```

Figure 9.11: SPARQL query to display the name of the data sources which have used a given ontology.

Figure 9.11 displays the SPARQL query to access the U Ontology to obtain the list of data sources which are using the particular ontology and displays the name of

the ontology, the data source name (which is actually the URL), and the industry of the data source. This information helps the data consumer to know more about the adoption and uptake of the ontology in real world implementation.



oname	dsname	dsind
Friend of a Friend	www.waffen-frank-shop.de	eCommerce
Friend of a Friend	c-paintings.com	eCommerce
Friend of a Friend	www.cisema.de	eCommerce
Friend of a Friend	www.akw-fitness.de	eCommerce
Friend of a Friend	bitmunk.com	eCommerce
Friend of a Friend	www.jarltech.de	eCommerce
Friend of a Friend	www.3kbo.com	eCommerce
Friend of a Friend	www.herold.at	eCommerce
Friend of a Friend	www.collibra.com	eCommerce
Friend of a Friend	www.tenera.ch	eCommerce
Friend of a Friend	semantic.eurobau.com	eCommerce

Figure 9.12: Result of SPARQL query shown in Figure 9.11.

Figure 9.12 displays the list of different data sources and the industry to which they belong. The query has returned the name of different data sources which have used *foaf* (Friend of a Friend) and their respective industry (application domain). Knowing who is using a particular ontology and their domain helps ontology developers to perform a detailed analysis on these data sources to investigate exactly how the ontology components are being used.

9.4.4.5 Req. 5: What cohesive groups of ontologies have similar usage?

In this requirement, the user would like to know the different cohesive groups based on their co-usage from the dataset. This helps to identify the different ontologies which have some commonality (this could be semantic similarity in terms of describing related but different concepts) to enable users to understand or analyse their characteristics. In the OUN-AF framework, the Cohesive Subgroups metric is defined to measure the k-core of the Ontology Co-Usage network. The U Ontology

provides the attribute to specify the k-core value to which the ontology belongs. The `OntologyUsage` concept has attribute `hasCousedValue` to represent the k-core value to which it belongs. The value is, in fact, the degree of the node (given ontology) of the projected Ontology Co-Usage network which indicates how many ontologies are being co-used with it.

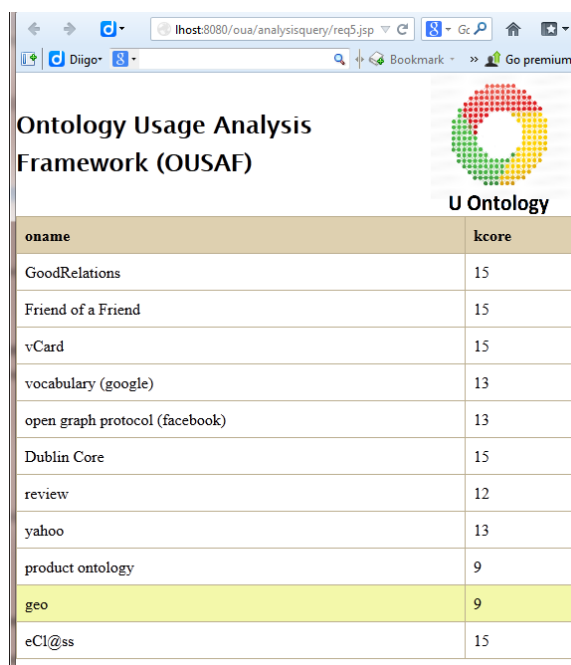
```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?oname ?kcore
4 WHERE {
5     ?ontology rdf:type uo:OntologyUsage;
6                 uo:name           ?oname;
7                 uo:hasCousedValue ?kcore;
8 }

```

Figure 9.13: SPARQL query to extract the k-core value ontologies

Figure 9.13 lists the SPARQL query which displays the k-core value of the ontologies. The ontologies with the same k-core value belong to the same cohesive group. In order to obtain the ontologies belonging to a particular cohesive group, the `?kcore` variable can be used to limit the ontologies belonging to a group.



oname	kcore
GoodRelations	15
Friend of a Friend	15
vCard	15
vocabulary (google)	13
open graph protocol (facebook)	13
Dublin Core	15
review	12
yahoo	13
product ontology	9
geo	9
eCl@ss	15

Figure 9.14: Result of SPARQL query shown in Figure 9.13.

Figure 9.14 displays the names of the ontologies and the cohesive group to which they belong. From the list, it can be seen which ontologies have a similar usage in the dataset. Such insight into the presence of different cohesive groups helps ontology developers to know the co-usability factor present among different ontologies.

9.4.5 Discussion of Findings

In this section, the usability and adequacy of the results from the identification phase are presented. In order to analyse the usefulness and adequacy of the results obtained through OUN-AF, use cases are presented which represent the frequently occurring requirements of the users. For each requirement, the ontology identification phase, its computational model and applicable metrics are discussed to describe their capability to address these requirements. For users to make use of the ontology usage analysis, for each requirement, the U Ontology is accessed to obtain the required information by posing the SPARQL queries. Based on the described use cases and the solution offered by the OUN-AF framework and U Ontology, the findings are summarized in following points:

1. The OUN-AF framework is able to provide the method and techniques which were necessary to address the requirements of the identification phase of the OUSAF framework. Therefore, the **OUN-AF framework is capable and its techniques and methods are adequate to provide the required insight into ontology identification.**
2. The U Ontology is able to represent and capture the concepts pertaining to the ontology identification phase. **The information required for each use case was retrieved by accessing the U Ontology, therefore the conceptual model formalizing the domain of ontology usage is considered adequate to address the users requirements.**

9.5 Utilization of the Empirical Analysis Framework (EMP-AF)

9.5.1 Aim

The aim of the investigation phase of the OUSAF framework is to analyse the use of ontologies on the Web. In order to do this, the analysis is performed at two levels: the empirical level and quantitative level. The EMP-AF framework empirically analyses ontology usage from different aspects to provide insight into the use of an ontology and its different components which includes the use of different concepts, other ontologies/vocabularies used to describe the entities, and the use of different relationships and data properties to provide factual statements about entities.

9.5.2 Flow of Activities

The ontology identified by the OUN-AF or provided by user is empirically analysed using the EMP-AF framework. The framework makes use of the dataset collected by the hybrid crawler presented in Section 9.3. The EMP-AF for each concept applies the Concept Usage Template (CUT) which analyses the concepts from different aspects and metrics are used to measure them. The obtained analysis and ontology components are then populated into the U Ontology for the dissemination of usage analysis.

9.5.3 Use Cases

The EMP-AF framework empirically analyses the use of ontologies and provides detailed insight into the use of ontologies and its components by different data sources (data publishers). The different aspects which are analysed pertaining to concepts are their instantiation, the use of other ontologies to describe the entities instantiated by the concept, the relationship it has with other entities, the use of label properties and prevalent knowledge patterns in the dataset.

The three types of users defined in Section 9.2.1, based on their function and role, require a different set of information pertaining to the ontology usage analysis. The information requirements of each type of user are described in the following and will be used to demonstrate the utilization of the framework.

Ontology developer's requirements

- Req. 1) What is the adoption level of a given ontology?
- Req. 2) How are the entities of a given concept described?

Data consumers requirements

- Req. 3) How are the entities textually described?

Data publishers requirements

- Req. 4) What knowledge patterns are available in the dataset?

The abovementioned requirements are used to demonstrate the utilization of the EMP-AF framework.

9.5.4 Utilization Analysis

9.5.4.1 Req. 1: What is the adoption level of a given ontology?

In this requirement, the user is interested in knowing how a particular ontology is being adopted by the end user. It could be that the ontology developer is interested in their own developed ontology or would like to know about another ontology to observe the usage trends in similar ontologies. The adoption of an ontology is a generic observation which can include several components to provide a comprehensive overview of how an ontology and its components are being used. Here, the user is interested in knowing the terminologies of the ontology that have some usage on the Web (in real world implementation).

The EMP-AF framework empirically analyses the use of different terms of the ontology in the dataset. The CUT template and other metrics defined as part of the EMP-AF framework helps in generating the terminological knowledge of the ontology that is being used and adopted by data publishers. The U Ontology captures the components of the ontology along with their usage to provide an overview of their usage uptake. Figure 9.15 displays the SPARQL query to retrieve the terminologies of the ontology which are being used on the Web. Here, since the objective is to obtain the list of terms which have been used, irrespective of their usage level, the query does not retrieve their usage frequency.

```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?ocon, ?oatt, ?orel
4 WHERE {
5     ?ontology uo:prefix      "gr".
6
7     OPTIONAL {?ontology uo:hasConceptUsage ?ocon.}
8     OPTIONAL {?ontology uo:hasAttributeUsage ?oatt.}
9     OPTIONAL {?ontology uo:hasRelationshipUsage ?orel.}
10 }
11 LIMIT 50

```

Figure 9.15: SPARQL query to display the terms of the ontology which have usage in the dataset.

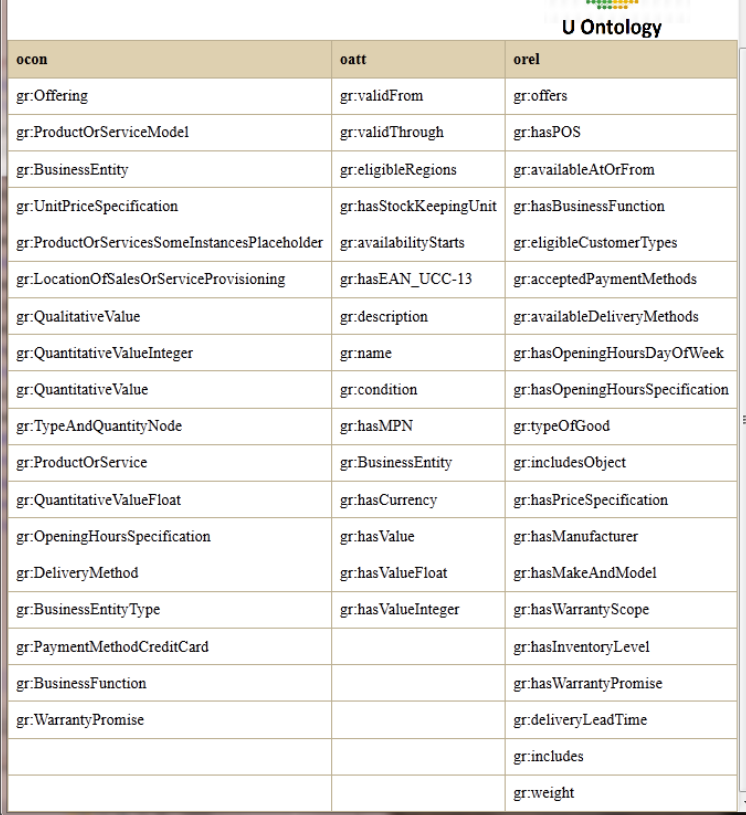
The query shown in Figure 9.15 lists all the ontology components, including the concepts (classes), object properties (relationships) and data type properties (attributes) of the ontologies which have instantiation in the dataset. The U Ontology model captures all these components through ConceptUsage, RelationshipUsage and AttributeUsage concepts. However, as can be seen in the U Ontology model (Figure 8.10), these three classes are subclasses of the class Term.


```
1 SQL>SPARQL
2 DEFINE input:inference "http://example.uontology.org/inference/rdf9"
3 SELECT ?tname
4 WHERE{
5     ?ontology uo:prefix "gr".
6     ?ontology uo:hasTerm ?oterm.
7     ?oterm uo:name ?tname.
8 };
```

Figure 9.16: Query exploiting RDFS entailment rule (rdfs9)

This subsumption relationship allows the retrieval of all the instances of the subclasses though the use of RDFS entailment rules. Applying the axiomatic triples available in the ontology and rdfs9 rule set, the implied information at the instance level can be retrieved. The Virtuoso (open source) triple store ([OpenLink Software, 2009](#)) which provides RDFS rule-based reasoning support, inference context (i.e. `http://example.uontology.org/inference/rdf9`), is defined to retrieve all the terms of the ontology that have usage through the term concept. Figure 9.16 display the SPARQL query which retrieves a similar result but through inference.

Figure 9.17 displays the names of the "gr" terms that have been used in the dataset. In order to provide a comprehensive list of ontology terms, the result screen is edited to show the concepts, object properties and the attributes in the first, second and third columns, respectively. This insight is useful for all types of users because it provides a consolidated view of the ontology usage. Ontology developers can use it to know which concepts are being instantiated and analyse those which are not being used. This also helps in implementing changes to the ontology, as based on what is being used, ontology developers can choose a suitable approach.



U Ontology		
ocon	oatt	orel
gr:Offering	gr:validFrom	gr:offers
gr:ProductOrServiceModel	gr:validThrough	gr:hasPOS
gr:BusinessEntity	gr:eligibleRegions	gr:availableAtOrFrom
gr:UnitPriceSpecification	gr:hasStockKeepingUnit	gr:hasBusinessFunction
gr:ProductOrServicesSomeInstancesPlaceholder	gr:availabilityStarts	gr:eligibleCustomerTypes
gr:LocationOfSalesOrServiceProvisioning	gr:hasEAN_UCC-13	gr:acceptedPaymentMethods
gr:QualitativeValue	gr:description	gr:availableDeliveryMethods
gr:QuantitativeValueInteger	gr:name	gr:hasOpeningHoursDayOfWeek
gr:QuantitativeValue	gr:condition	gr:hasOpeningHoursSpecification
gr:TypeAndQuantityNode	gr:hasMPN	gr:typeOfGood
gr:ProductOrService	gr:BusinessEntity	gr:includesObject
gr:QuantitativeValueFloat	gr:hasCurrency	gr:hasPriceSpecification
gr:OpeningHoursSpecification	gr:hasValue	gr:hasManufacturer
gr:DeliveryMethod	gr:hasValueFloat	gr:hasMakeAndModel
gr:BusinessEntityType	gr:hasValueInteger	gr:hasWarrantyScope
gr:PaymentMethodCreditCard		gr:hasInventoryLevel
gr:BusinessFunction		gr:hasWarrantyPromise
gr:WarrantyPromise		gr:deliveryLeadTime
		gr:includes
		gr:weight

Figure 9.17: Result of SPARQL query shown in Figure 9.15.

9.5.4.2 Req. 2: How are the entities of a given concept described?

In this requirement, the user is interested in knowing how the entities of a specific type (instance of a concept) are being semantically described on the Web. This information helps not only ontology owners know the prevalent entity schema, but is also of interest to data publishers and consumers to know the entity schema. For ontology developers, it provides insight regarding the use of different relationships to describe the related entities and their aspects, and attributes to provide factual statements about the entity.

```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?rname, ?rvname, ?aname, ?avname
4 WHERE {
5     ?concept uo:name      "BusinessEntity";
6             uo:prefix    "gr".
7     {
8         ?concept uo:hasRelation ?rrel;
9             uo:name      ?rname;
10            uo:isComponentOf ?rvocab.
11    ?rvocab uo:name      ?rvname.
12    }
13    UNION
14    {
15        ?concept uo:hasAttribute ?aatt;
16            uo:name      ?aname;
17            uo:isComponentOf ?avocab.
18    ?avocab uo:name      ?avname.
19    }
20 }
21 LIMIT 50

```

Figure 9.18: SPARQL query to display the use of different relationships and attributes.

The CUT of the EMP-AF framework provides the model to capture the relevant aspects of the entities. The U Ontology represents the components of the Concept Usage Template (CUT) to enable users to access the relevant components of the CUT template. Figure 9.18 list the SPARQAL query which accesses the U Ontology to retrieve the semantic description used to define the entity of a specific type (concept). The query displays the semantic description of the entity which is the instance of the BusinessEntity concept of the GoodRelations ontology. In the query, all the object properties and the different data type properties which have been used in the dataset to describe the entity are displayed. Since it is a common practice to reuse terms defined in other ontologies, in addition to term names, their respective ontology is also retrieved in the query.

rname	rvname	aname	avname
gr:offers	GoodRelations	gr:legalName	GoodRelations
gr:hasPOS	GoodRelations	gr:hasISICv4	GoodRelations
vCard:add	vCard	gr:hasNAICS	GoodRelations

Figure 9.19: Result of SPARQL query shown in Figure 9.18.

Figure 9.19 displays the different properties that are being used to describe BusinessEntity entity. It can be seen that terms from other ontologies are being reused to describe the instance of the entity. This helps ontology developers to know

which other concepts are being frequently used to describe the entity. For data publishers, this helps to know which other terms should be considered for semantic annotation.

9.5.4.3 Req. 3: How are entities textually described?

In this requirement, the user, particularly the data consumer (application developer), is interested in knowing what label properties are being used to describe a particular entity. Semantic web data defines resources using URIs which are opaque and do not provide human (reader) friendly detail about the resource. The data publisher makes use of label properties to provide a textual description about the resources and allows application developers to make use of these properties to either know more about the resource or use them to develop the application interfaces.

As mentioned in Section 6.4.4, in the EMP-AF framework, two types of label properties are defined: formal labels which are part of the W3C-based vocabularies; and domain labels which are defined by the particular ontologies. Figure 9.20 lists the query which retrieves the use of the domain and the formal labels used by data publishers to provide a textual description for the entities.

```
1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?filename, ?domainname
4 WHERE {
5     ?concept uo:name "BusinessEntity";
6             uo:prefix "gr".
7     {
8     ?concept uo:hasFormalLabel ?flab;
9             uo:name ?filename.
10    }
11    UNION
12    {
13    ?concept uo:hasDomainLabel ?dlab;
14            uo:name ?domainname.
15    }
16 }
```

Figure 9.20: SPARQL query to access the formal and domain labels used for a concept.

The U Ontology provides the `FormalLabel` and `DomainLabel` concepts to capture the label properties used in the dataset and are subclasses of `LabelUsage`. In Figure 9.20, the query retrieves all the labels used for the instances of the `BusinessEntity` concept of the `GoodRelations` ontology. The names of the label properties are displayed. In case a distinction is not required, the `LabelUsage` with RDFS entailment rules can be used to access the same result (similar to Req. 1).

fname	dname
rdfs:label	gr:legalName
rdfs:comment	gr:category

Figure 9.21: Result of SPARQL query shown in Figure 9.20.

Figure 9.21 shows the use of different label properties to provide a textual description about the entity for human readability or user interface. In the case of entities of type `BusinessEntity`, these are described using both the domain and formal labels which includes `rdfs:label`, `rdfs:comment`, `gr:legalName` and `gr:category`. Application developers and data publishers can use this information to develop the user interface and provide the textual description in the place of opaque URIs.

9.5.4.4 Req. 4: What knowledge patterns are available in the dataset?

In this requirement, the user is interested in knowing the knowledge patterns that are prevalent in the published Semantic Web data. The knowledge patterns provide terminological knowledge in the sequence of paths comprising different path steps. Each path step is of a concept-predicate-concept pattern and different path steps which are linked (chained) in the RDF graph constitute a path.

The EMP-AF framework implements the technique and method to extract the knowledge patterns present in the dataset to allow users to know the prevalent structure of the schema level graph. The U Ontology represents all the conceptual elements to capture the components of the knowledge pattern as depicted in Figure 8.9.

```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?kpsubname, ?kpprename, ?kpobjname
4 WHERE {
5     ?ontology a uo:OntologyUsage.
6     ?ontology uo:hasKnowledgePattern ?kp.
7     ?kp uo:hasPath ?path.
8     ?path uo:hasPathStep ?pathstep.
9     ?pathstep uo:hasSubjectInPath ?kpsub;
10    uo:hasPropertyInPath ?kppre;
11    uo:hasObjectInPath ?kpobj.
12    ?kpsub uo:name ?kpsubname.
13    ?kppre uo:name ?kpprename.
14    ?kpobj uo:name ?kpobjname.
15 }

```

Figure 9.22: SPARQL query to display the knowledge patterns in the dataset.

Figure 9.22 lists the query which displays the knowledge patterns found in the dataset. The U Ontology implements the knowledge pattern conceptual model to

allow users to represent and access the schema level triples included in it. The query displays all the schema level triples constituting the knowledge pattern.

9.5.5 Discussion on Findings

In this section, the usability and adequacy of the results from the EMP-AF framework is presented. To demonstrate the utilization, use cases are presented to reflect the common requirements of different type of users. Each use case represents the frequently occurring requirements from different users perspectives. For each requirement, the method, technique and metrics implemented as part of the EMP-AF is discussed. In terms of allowing the user to make use of the output of empirical analysis, the U Ontology is queried against each requirement. Based on the use cases and the solution offered by the EMP-AF framework and U Ontology, the findings are summarized in the following points:

1. The EMP-AF framework implements the methods and techniques which were required to empirically analyse the use of ontologies. Through the implementation of the framework, it has successfully addressed the requirements of the use cases related to the investigation phase of the OUSAF framework. Therefore, **the EMP-AF framework, its techniques and methods are adequate to provide the required insight about the empirical analysis of ontology usage.**
2. The U Ontology is able to represent and capture the concepts pertaining to the Concept Usage Template, labelling and knowledge patterns. **The information required for each use case was retrieved by accessing the U Ontology, therefore the conceptual model formalizing the domain of ontology usage is considered adequate to address the users requirements.**

9.6 Utilization of the Quantitative Analysis Framework (QUA-AF)

9.6.1 Aim

As mentioned in the previous section, the aim of the investigation phase is to empirically and quantitatively analyse the use of ontologies. The focus of this section is quantitative analysis. In quantitative analysis, using the identified aspects, key

dimensions which are important for measuring ontology usage are defined. The quantitative analysis is performed from three dimensions: richness, technology and business. Based on these dimensions, the use of an ontology and its components is quantitatively measured and using a ranking approach, a quantified rank of each term is obtained. To quantitatively measure ontology usage, the QUA-AF framework is proposed and its implementation is discussed in Chapter 7.

9.6.2 Flow of Activities

The QUA-AF framework comprises three phases: data collection, computation and application. In QUA-AF, the dataset is analysed from three dimensions and each dimension requires a different type of dataset. For richness, the formalized ontological model and form technology, the same dataset is used which is described in Section 9.3. For the business dimension, separate data is collected comprising semantic mark-ups that are supported by search engines. The methods, techniques and metrics developed for the QUA-AF framework are then used to develop the web schema which provides a snapshot of the terminological knowledge that is published on the Web in a specific application area.

9.6.3 Use Cases

Quantitative analysis performed using the QUA-AF framework allows users to analyse ontology usage from different dimensions while providing a consolidated rank of terms. Each dimension is measured and captured independent from the other to allow users to access information specific to their requirements.

In order to demonstrate the utilization of the QUA-AF framework, use cases are defined to reflect the usage scenarios applicable to different types of users.

Ontology developer's requirements

Req. 1) What is the richness value of the concepts in a given ontology?

Data consumer's requirements

Req. 2) Display the ontology terms based on their usage ranking?

Data publishers' requirements

Req. 3) List the terms that are being recognized by search engines?

Req. 4) What ontologies are being used in a given application area?

The abovementioned requirements are used to analyse the utilization of the QUA-AF framework.

9.6.4 Utilization Analysis

9.6.4.1 Req. 1: What is the richness value of the concepts in a given ontology?

In this requirement, the user is interested in knowing how the concept in a given ontology is structured. Structure refers to the typological characteristics being defined in the ontology to conceptually describe a concept. A concept is described by creating a type relationship with other concepts and specifying the attributes to capture the factual state of the entity conceptualized by the concept. The richness value as mentioned in Section 7.3.1 quantifies the structural and typological characteristics of the concept, relationships and attributes.

The QUA-AF framework defines the metrics to measure the value of concepts and for that matter, the ontology's authoritative documents are accessed by the framework to measure the richness value of ontology components. For each type of ontology component, respective metrics are defined to measure the richness value of concepts, relationships and attributes. These values are then represented in the U Ontology to allow users to access the quantitative analysis of the ontology usage.

```

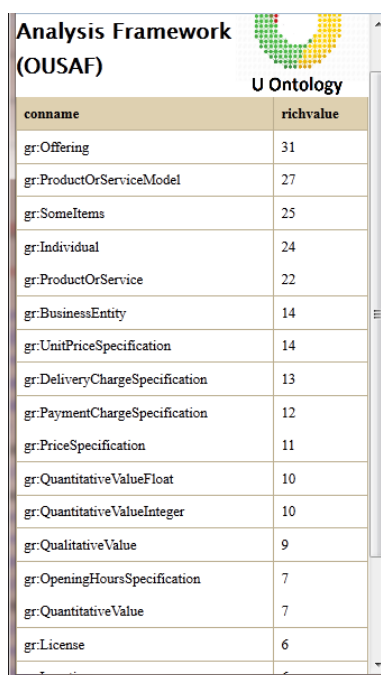
1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?conname ?richvalue
4 WHERE {
5     ?ontology rdf:type          uo:OntologyUsage;
6               uo:prefix        "gr".
7     ?ontology uo:hasConceptUsage ?oconcept;
8               uo:name          ?conname.
9     ?oconcept uo:hasRichness    ?ocrich.
10    ?ocrich   up:value          ?richvalue.
11 }ORDER BY DESC (?richvalue)

```

Figure 9.23: SPARQL query to display the concepts and their richness value of a specific ontology

Figure 9.23 displays the SPARQL query to list all the concepts of an ontology and their richness value. The list includes the concepts which have been used on the Web and their richness value is computed using the metrics defined as part of QUA-AF framework (Section 7.3.1). The U Ontology captures the concepts and attributes which

are necessary to represent the richness value of the ontology components, including relationships and attributes. In the above query, the concepts of a given ontology (i.e. GoodRelations, which has “gr” as a prefix in the triple store) are accessed along with their richness value. The query displays the name and the value of the concepts in descending order.



conname	richvalue
gr:Offering	31
gr:ProductOrServiceModel	27
gr:SomeItems	25
gr:Individual	24
gr:ProductOrService	22
gr:BusinessEntity	14
gr:UnitPriceSpecification	14
gr:DeliveryChargeSpecification	13
gr:PaymentChargeSpecification	12
gr:PriceSpecification	11
gr:QuantitativeValueFloat	10
gr:QuantitativeValueInteger	10
gr:QualitativeValue	9
gr:OpeningHoursSpecification	7
gr:QuantitativeValue	7
gr:License	6

Figure 9.24: Result of SPARQL query shown in Figure 9.23.

Figure 9.24 displays the different concepts of the GoodRelations ontology with their concept richness values (Section 7.3.1). This helps in understanding how the entities are being conceptualised and semantically described in their formalized model. The concept richness value is combined with other metrics to generate a ranked list of ontology terms to allow users to use the terms based on their requirements (e.g terms with a higher usage or richness value).

9.6.4.2 Req. 2: Display the ontology terms based on their usage ranking?

In this requirement, the user is interested in obtaining a list of terms, based on their usage ranking. This includes all the concepts, relationships, and attributes defined by the ontology which have been used by data publishers. In the QUA-AF framework, for each dimension, metrics are defined and computed using their respective repositories, as each requires different types of data for computation. In order to obtain the consolidated rank comprising these three dimensions, the QUA-AF framework computes and consolidates the values based on the ranking approach (Eq

7.9) presented in Section 7.3.4. Since the priority or relevance of each dimension is controlled through weights specified by the user, the U Ontology does not capture the consolidated ranking value. However, the U Ontology defines the concepts to represent the value computed for each dimension to allow users to obtain the terms and their values of each dimension.

```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT *
4 WHERE {
5     ?ontology    uo:prefix    "gr".
6     {
7         ?ontology    uo:hasConceptUsage    ?con.
8         ?con         uo:name            ?cname.
9         ?con         uo:hasRichness      ?conrich.
10        ?con         uo:isIncentividesBy  ?sengine.
11
12        ?con         uo:hasUsageValue     ?cuvalue.
13        ?conrich    uo:hasRichnessValue  ?crvalue.
14        ?sengine    uo:hasIncentiveValue ?civalue.
15    }
16    UNION
17    {
18        ?ontology    uo:hasRelationshipUsage ?rel.
19        ?rel         uo:name            ?rname.
20        ?rel         uo:isIncentivisedBy  ?rincv.
21        ?rel         uo:relationshipValue ?rrich.
22
23        ?rel         uo:hasUsageValue     ?ruvalue.
24        ?rrich      uo:hasRichnessValue  ?rrvalue.
25        ?rincv     uo:hasIncentiveValue  ?rivalue.
26    }
27    UNION
28    {
29        ?ontology    uo:hasAttributrUsage  ?att.
30        ?att         uo:name            ?aname.
31        ?att         uo:isIncentivisedBy  ?attincv.
32        ?att         uo:attributeValue    ?attval.
33
34        ?att         uo:hasUsageValue     ?auvalue.
35        ?attval     uo:hasRichnessValue  ?arvalue.
36        ?attincv   uo:hasIncentiveValue  ?aivalue.
37    }
38 }LIMIT 50

```

Figure 9.25: SPARQL query to display the usage of given ontology terms.

Figure 9.25 displays the terms of the ontologies and their values computed from three dimensions. The query retrieves all the concept and their usage, richness and incentive values computed by the QUA-AF framework that will be published through the U Ontology. However, the query is not able to compute the final consolidated rank value for each component since weights are required for this. The computation of the final rank value can be computed, based on the data provided by the query and by specifying the weights.

Concept	cuvalue	crvalue	civalue
gr:Offering	27165	31	3
gr:ProductOrServiceModel	6275	27	1
gr:SomeItems	12465	25	2
gr:Individual	32	24	1
gr:ProductOrService	8	22	2
gr:BusinessEntity	1714	14	3
gr:UnitPriceSpecification	14265	14	0
gr:DeliveryChargeSpecification	6	13	0
gr:PaymentChargeSpecification	15	12	0
gr:PriceSpecification	101	11	0
gr:QuantitativeValueFloat	6603	10	0
gr:QuantitativeValueInteger	92	10	0
gr:QualitativeValue	950	9	0
gr:OpeningHoursSpecification	703	7	0
gr:QuantitativeValue	0	7	0
gr:License	5	6	2
gr:Location	1057	6	2

Figure 9.26: Result of SPARQL query shown in Figure 9.25.

Figure 9.26 displays the list of ontology terms along with their usage, richness and incentive measures. These values help ontology owners and data publishers to analyse the usage from different dimensions and based on their requirements, and by applying a threshold value (filter), terms with a certain usage or rank value can be obtained.

9.6.4.3 Req. 3: List the terms that are being recognized by search engines?

In this requirement, the user, particularly a data publisher, is interested in knowing which term are presently being recognized by search engines. In the QUA-AF framework, for the business dimension, commercial incentives are measured by identifying which terms are recognized by search engines when used to semantically describe information published on the Web. For this, three search engines are used and if the term is recognised by any of them, it is represented in the U Ontology. U Ontology provides the concept and attributes to allow users to access incentive-related information.

```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?comname, ?sename
4 WHERE {
5   ?component    uo:isIncentivisedBy    ?se.
6   ?component    uo:name                 ?comname.
7   ?se           uo:searchEngineName    ?sename.
8
9 }LIMIT 50

```

Figure 9.27: SPARQL query to list the terms that are being recognised by search engines.

Figure 9.28 displays the terms that are being recognized by search engines. This list helps data publishers to know which terms to consider if the user is interested in greater visibility of information on the Web. Similarly, it lets data publishers and application developers know which terms to prefer over others, based on their recognition by other search engines.

(OUSAF)	
U Ontology	
comname	sename
Address	Google
author	Google
locality	Google
Offer	Google
Organization	Google
Person	Google
photo	Google
postal-code	Google
Product	Google
Recipe	Google
Review	Google
street-address	Google
title	Google

Figure 9.28: Result of SPARQL query shown in Figure 9.27.

9.6.4.4 Req. 4: What ontologies are being used in a given application area?

In this requirement, the user is interested to know which ontologies should be considered for information relevant to their domain. Several ontologies which have a high level of usage have been published on the Web, therefore it is more beneficial to consider these. A detailed usage analysis of each ontology can be obtained, but to begin with, it is important to know which ontology to consider. In several previous requirements, ontology-specific information has been queried from the U Ontology,

however, in this requirement, the name of an ontology relevant to an application area is required.

```

1 PREFIX uo: <http://example.uontology.org/v1#>
2
3 SELECT ?oname, ?oprefix
4 WHERE {
5   ?oua    rdf:type          uo:OntologyUsageAnalysis.
6   ?oua    uo:analysisOntology ?ou.
7   ?ou     uo:name           ?oname.
8   ?ou     uo:prefix         ?oprefix.
9   ?ou     uo:isUsedby       ?ds.
10  ?ds     uo:industry        ?ind.
11
12  FILTER regex(str(?ind), "eCommerce", "i")
13 }

```

Figure 9.29: SPARQL query to list the ontologies being used in a given application area.

Figure 9.29 displays the name and prefix of the ontologies that have been used in a dataset which are relevant to the e-Commerce domain. The U Ontology captures the meta data regarding the dataset and the usage analysis to allow users to obtain information regarding when the analysis was performed, the nature of the data included in the dataset and the industry which the majority of data represent. In the query, all the ontologies which have been identified and analysed by the OUSAF framework are retrieved and using a filter, only ontologies labelled (tagged) as e-Commerce are displayed. The same query can be used to obtain a list of ontologies relevant or applicable to other application areas.

(OUSAF)	U Ontology
oname	oprefix
GoodRelations	gr
Friend of a Friend	foaf
vCard	vcard
vocabulary (google)	v
open graph protocol (facebook)	og
Dublin Core	dc
review	rev
yahoo	yahoo
product ontology	pto
geo	geo

Figure 9.30: Result of SPARQL query shown in Figure 9.29.

Figure 9.30 lists all the ontologies/vocabularies that are being used to describe information in a particular domain. The query retrieves the ontologies being used in e-Commerce domains and this information helps data publishers and application developers to know which ontologies to consider for either developing ontology-driven

application or for using them to semantically describe information on the Web.

9.6.5 Discussion on Findings

In this section, the usability and adequacy of the results from the QUA-AF framework is presented. To demonstrate the utilization, use cases are presented to reflect the common requirements of different types of users. For each use case, requirement analysis is presented and the methods, techniques and metrics implemented as part of QUA-AF framework are discussed. For each user, SPARQL queries are made to the U Ontology which represents the usage analysis-related data to provide the information needed by the user. Based on the use cases and the solution offered by the QUA-AF framework and the U Ontology, the findings are summarized in the following points:

1. The QUA-AF framework implements the methods and techniques which are required to quantitatively analyse the use of ontologies. Through the implementation of the framework, it has successfully addressed the requirements of the use cases related to the investigation phase of the OUSAF framework. Therefore, the **QUA-AF framework is capable and its techniques and methods are adequate to provide the required insight into the quantitative measures regarding ontology usage.**
2. The U Ontology is able to represent and capture the concepts pertaining to the richness, technology and business dimensions and the quantified measures obtained through the metrics defined in the QUA-AF framework. **The information required for each use case was retrieved by accessing the U Ontology, therefore the conceptual model formalizing the domain of ontology usage is considered adequate to address the users requirements.**

In the next section, the achievements obtained through the utilization phase are discussed.

9.7 Benefits of the Utilization phase

The objective of the utilization phase is to enable users to access the OUSAF framework and analyse ontology usage, based on their requirements. The communication between the users and the computational components of the OUSAF framework is facilitated through the use of the U Ontology. For each type of user, in the following subsections, the benefits obtained in the utilization phase are summarized.

9.7.1 Benefits from an Ontology Developer's perspective

An ontology developer is interested in knowing the performance of a given ontology in terms of its usage. By using the OUSAF, ontology developers can determine:

- the usage of a given ontology which includes the number of data sources that are using the ontology, the number of instances created using the ontology namespace and the ontology meta- information. In Section 9.5.4.1, the request that is made to the OUSAF framework by posing the query is shown in Figure 9.6. The query is able to provide the required information to the user who can then use it for further processing.
- the usage of different components of an ontology. The components are the concepts, object properties and attributes which have instantiation on the Web. In Section 9.6.4.1, the requirement of obtaining the ontology adoption level is achieved by posing the SPARQL query shown in Figure 9.15, which provides a list of the names of the ontology components which have been used on the Web.

9.7.2 Benefits from a Data Consumer's perspective

A data consumer is interested in not only knowing which ontologies are available for use but to also know exactly what is being used in these ontologies. By using the OUSAF, the data consumer can:

- obtain a list of all ontologies which are presently being used by the other publishers to describe information related to the domain of interest. In Section 9.5.4.3, the requirements for retrieving a list of ontologies used in a specific domain are obtained through the query shown in Figure 9.10. This query provides the name, prefix of the ontologies and also the number of different data sources who are using it. The availability of such information helps data consumers achieve their objectives.
 - determine what labeling properties are being used by the community and are provided by the ontologies to textually describe the information. The textual description provided by these labels allows human readers to understand the entity and also allows user interfaces to display labels rather than opaque URIs. In Section 9.6.4.3, the requirements for retrieving the list of label properties that are being used by different publishers is obtained through the query shown in Figure 9.20. This query is capable of providing details of label properties such as their name and the ontology to which they belong.
-

9.7.3 Benefits from a Data Publisher's perspective

Data publishers prefer to reuse ontologies to benefit from the advantage of the existing support and acceptance in their respective community. To do so, they need to learn about the current usage level of different ontologies and their co-usability among different ontologies. The benefits available to data publishers through the utilization of OUSAF framework are described below.

- One of the new motivations for using ontologies on the Web is the support they are given by different search engines. For data publishers, it is important to know which terms are being recognised by which search engines as this will help them improve the visibility of their information on the Web. In Section 9.7.4.3, the list of the terms that are being supported by search engines and the name of the search engine which supports them are obtained through the query shown in Figure 9.27. Using the results obtained through this query helps data publishers in choosing the terms they should use for their semantic description.
- In Section 9.5.4.5, the cohesive groups of ontologies with similar usage are obtained through the query presented in Figure 9.13. The identification and availability of cohesive groups helps data publishers understand the prevalent semantic structure available in the currently published Semantic Web data and consider it for their semantic annotation.

Based on the abovementioned benefits of the OUSAF framework and the use cases in Section 9.5 to 9.7, it can be concluded that the OUSAF framework through its computational components (OUN-AF, EMP-AF, and QUA-AF) and the U Ontology is successfully able to demonstrate the utilization of the proposed framework.

9.8 Comparison of the output from the OUSAF framework with existing approaches in the literature

The purpose of this section is to compare the proposed OUSAF with existing solutions in the literature on ontology usage analysis and present the comparative analysis. As highlighted in Chapters 1-4, circa 1999–2006 ontology usage analysis was not considered as an area in the ontology lifecycle model and often ontology analysis was done purely for ontology evaluation and evolution purposes. Therefore, there is

no literature in which ontologies are being analysed from their “usage” perspective. Considering the lack of a framework for measuring ontology usage analysis in the literature, in this section, a comparison with the approaches related to ontology engineering areas such as ontology evaluation and evolution is presented. This will help in appreciating the difference between ontology usage and other subareas in ontology engineering for comparative analysis.

- No framework is proposed in the literature which empirically and quantitatively measures the use of ontologies. However, in the literature, several frameworks are proposed for ontology evaluation which measures the *quality* of the ontology whereas the OUSAF framework measures the *usage* of an ontology. In the OUSAF framework, the developed metrics are centered around measuring the use of an ontology and its components on the Web after they are developed, whereas most of the ontology evaluation techniques evaluate ontologies before they are deployed.
- No pragmatic approach is proposed in the literature which provides a feedback loop regarding the use of ontologies to the ontology evolution process. Most of the ontology evolution techniques in the literature focus on measuring the level of inconsistency that arises as a result of change, however, none of the approaches consider the usage status as one of the factors for ontology evolution. Having a feedback loop on the prevailing use of ontologies to the ontology evolution process by using the OUSAF will help to evolve ontologies pragmatically.
- Different meta-ontologies are proposed in literature to represent information about ontologies. However, in the literature, no ontology is proposed to capture the “usage” perspective of ontologies. The development of the U Ontology allows the representation of the domain knowledge of the usage analysis and the terminological knowledge of different ontologies with usage-related statistics to enable users to it access, programmatically and manually.

From the above discussion, it can be inferred that the proposed solution for measuring ontology usage provides a complete methodology for the user to identify ontologies, empirically and quantitatively analyse their usage and formally represent the usage-related information. In contrast, none of the approaches proposed in the literature provides a way to measure and analyse usage, provide a pragmatic feedback loop to the ontology evaluation and evolution process, or allow different types of users to access usage-related information through a formal conceptual model.

9.9 Conclusion

In this chapter, the utilization phase of the OUSAF framework was presented. In order to demonstrate the utilization of the OUSAF, an approach was presented which accesses each solution component i.e OUN-AF, EMP-AF, and QUA-AF from different users requirement perspectives. For each type of user, different use cases were presented to obtain the required information from the solution components. The results obtained in each use case were then analysed to see whether the information was useful and adequate for the user. A comparison of the output from the OUSAF framework was analysed with existing approaches in the literature.

In next chapter, the evaluation of the U Ontology is presented.

Chapter 10 - Evaluation of U Ontology

10.1 Introduction

In the last chapter, the utilization of the OUSAF framework and its computational components were presented. The U Ontology which conceptualizes the Ontology Usage Analysis domain was used to obtain ontology usage-related information from the OUSAF. In this chapter, the U Ontology will be evaluated using an ontology evaluation methodology to measure the quality of the developed ontology. For an ontology to be of good quality and remain useful for its users, it needs to conform to a set of good practices. These practices are analysed using different evaluation techniques which evaluate an ontology based on certain criteria to ensure the developed ontology meets the user's expectations.

The ontology evaluation methodology adopted for the evaluation of the U Ontology and the obtained observations are presented in this chapter. The criteria, which are considered by the adopted methodology to evaluate different aspects of the U Ontology, are described in Section 10.2. In Sections 10.3 – 10.8, the U Ontology is evaluated using different methods from different aspects supported by the methodology. In order to provide an overview of the evaluation made using different methods, in Section 10.9, a summary of the U Ontology evaluation is presented. Section 10.10 concludes the chapter.

10.2 Methodology for Ontology Evaluation

The purpose of ontology evaluation and its role in ontology engineering was discussed in Chapters 1-3. There are a number of frameworks (Section 2.2.1 discusses these in

detail) for ontology evaluation, all of which have the common objective of assessing the quality of a given ontology. While all the evaluation frameworks attempt to answer the question of how to assess the quality of an ontology for the Web, they differ in their approaches and techniques. [Brank et al. \(2005\)](#); [Vrandevcic \(2010\)](#) classified the ontology evaluation approaches into the following categories:

- Ontologies can be evaluated by themselves. In this category, the golden standard approach is adopted in which an ontology is assessed by comparing it with another ontology, e.g. ([Maedche and Staab, 2002](#)).
- Ontologies can be evaluated in some context. The context is often specified by including the additional artifact used to develop the ontology. The competency question also specifies the context of the ontology, e.g ([Grüninger and Fox, 1995](#)).
- Ontologies can be evaluated within an application. This means evaluating an ontology by using it within an application. This is also known as application-based ontology evaluation, e.g. ([Brank et al., 2005](#)).
- Ontologies can be evaluated in the context of an application and a task. These approaches are also known as task-based ontology evaluation, e.g. ([Porzel and Malaka, 2004](#))

While each of the above categories has a different approach, each gains from the evaluation of the other category. This means that the problems identified and rectified by the technique of one category will benefit the approaches in other category.

The evaluation methodology adopted for the evaluation of the U Ontology in this thesis is proposed by [Vrandevcic \(2010\)](#) and also discussed in ([Vrandečić and Sure, 2007](#); [Staab and Studer, 2009](#); [Völker et al., 2005](#)). The authors' proposed approach is based on the premise that a single measure to assess the quality of an ontology is elusive, and deriving concrete measures to identify the errors and loopholes in ontologies is a more practical approach. He states:

[...] instead of aiming for evaluation methods that tells us if an ontology is good, we settle for the goal of finding ontology evaluation methods that tell us if an ontology is bad, and if so, in which way."

Therefore, following the abovementioned approach, the evaluation methodology is used which provides a set of techniques and methods to evaluate an ontology from different aspects and helps in deciding how good, bad or satisfactory the ontology is.

In the next subsections, the criteria which should be considered while evaluating the ontologies to assure the quality of an ontology is discussed.

10.2.1 Criteria for Ontology Evaluation

Based on the previous proposals, in his dissertation (Vrandevecic, 2010) Vrandevecic discusses various criteria which a good ontology should meet. In the following, these criteria are briefly described. Under each criterion, there are several methods proposed which will be then used to assess the U Ontology.

- *Accuracy* : Accuracy is a criteria that states the knowledge represented by the ontology, including the axiomatic triples, complies to the knowledge of the stakeholders about the domain. A higher value of accuracy comes from the correct description of ontology components which includes classes, properties, and individuals.
 - *Adaptability* : Adaptability measures how flexible an ontology is in addressing user needs. Since ontologies are meant to be used on the Web, and their usage of the Web cannot be predicted, therefore the conceptual foundation should be capable of fulfilling the range of at least anticipated tasks.
 - *Clarity* : Clarity measures how effectively the ontology provides the understanding and meanings of the terms defined in the ontology. As a best practice, the terms defined in the ontology to name classes, properties and individuals should be understandable and unambiguous. This means that the definition of terms should be independent of the context and have interpretation by the users.
 - *Completeness*: This measures how well an ontology covers the domain of interest. The requirements within the scope of ontology should be answered. There are different aspects to analyse the completeness:
 - Completeness with regard to language (is the textual description required for a task in reasonable detail?)
 - Completeness with regard to domain (are all the key concepts representing the entities of the domain covered? Is it possible to represent all the individuals by the concepts?)
 - Completeness with regard to the application requirements (is all the data which is needed by the application present and representable by the ontology?)
 - *Computational efficiency*: Computational efficiency measures the ability of the tools to work with the ontology. The tools include: the databases (or triple stores)
-

to store the ontology and individuals, reasoners to reason over the ontology based on the axioms implemented in the ontology (and RDFS/OWL), and query processing. In particular, reasoners are important as they are often used to infer entailed knowledge, query answering, classification, and consistency checking.

- *Conciseness* : Conciseness measures whether the ontology includes the elements that are not relevant to the domain being represented through the ontology. This includes the definition of concepts which are not directly related to the domain of interest, or the presence of the concepts which gives redundant representation of the semantics.
- *Consistency* : Consistency indicates whether the ontology does not include or permit for any contradiction in the model. In accuracy, ontology compliance with the external source is measured, however, in consistency, it is observed that the ontology itself can be interpreted at all. Generally, consistency includes logical consistency and coherence and principles are defined to ensure that an ontology remains consistent and coherent.
- *Organizational fitness*: This measures how easily or challenging it is for an ontology to be implemented in an organization. This includes different aspects such as people (acceptance, resistance to change), tools (development tools, data bases, reasoners, software licenses), and technology and methodology (familiarity with the technology used in ontologies, and the methodologies part of the organizational information architecture).

10.2.2 Aspects to be analysed for Ontology Evaluation

Vrandevcic (2010) proposed six aspects which need to be considered while evaluating the ontologies using the abovementioned criteria as follows: Vocabulary, Syntax, Structure, Semantics, Representation and Context which are defined below.

- *Vocabulary* : This aspect refers to the names that are used in the ontology to describe the resources and literal values. The evaluation of the vocabulary aspect of the U Ontology is discussed in Section 10.3.
 - *Syntax* : This aspect refers to the serialization format used to encode the ontology. There are different types of syntax available and different ontologies use different syntaxes but all of them generate a graph. The evaluation of the syntactical aspect of the U Ontology is discussed in Section 10.4.
-

- *Structure* : This aspect represents how an ontology graph is arranged. Even though all ontologies are based on the RDF graph model, they can vary structurally. The evaluation of the structural aspect of the U Ontology is discussed in Section 10.5.
- *Semantics* : This aspect is about the formal meaning being represented by the ontology. The evaluation of the semantics aspect of the U Ontology is discussed in Section 10.6.
- *Representation* : This aspect represents the relationship between structure and semantics. The evaluation of the representational aspect of the U Ontology is discussed in Section 10.7.
- *Context* : This aspect covers the features of an ontology when compared with other artifacts. The evaluation of the contextual aspect of the U Ontology is discussed in Section 10.8.

In order to facilitate the evaluation of ontologies from these six aspects, [Vrandevcic \(2010\)](#) proposed 23 methods. These methods will be used to evaluate the U Ontology from the abovementioned six aspects.

For the evaluation of the U Ontology from each aspect, the applicable methods are presented with their definition (reproduced from ([Vrandevcic, 2010](#))), brief description and the evaluation result.

10.2.3 Metrics to quantify the evaluation findings

Each method is applied using the procedure, technique or process applicable as suggested by the methodology ([Vrandevcic, 2010](#)) and the results of each method are used to evaluate the U Ontology. The obtained evaluation results for each method are descriptive in nature and need to be analysed and interpreted, keeping in view the ontology and the knowledge base comprising instance data (i.e. the populated U Ontology). While these results help in understanding the ontology quality, they do not quantify the ontology's overall performance. In order to quantify the U Ontology evaluation and provide conclusive remarks about the results, the following four metrics are used.

- **Verified** : This means that the method is applied as required and the evaluation results obtained are positive. Positive indicates that no problem is found and the results are as expected.

- **Not Applicable** : This means that the method is not applicable to the ontology. This could be because the given method cannot be computed due to the ontology language or reasoner limitation.
- **Deferred** : This means that the method is applicable but could not be verified due to technical or time constraints and will be considered in future work.
- **Failed** : The method was applied but did not achieve the expected results.

At the end of the evaluation, the value of each metric will be accumulated (from the evaluation of the methods of each aspect) which will be used to summarize the evaluation of the U Ontology.

10.3 U Ontology evaluation: Vocabulary aspect

This aspect evaluates the vocabulary of the ontology. The vocabulary of an ontology is the set of all names used to define the terms (components of an ontology). Names can be either URIs or literals. URI references identify resources and thus provide a unique identifier to all the ontology components whereas literals are names that are mapped to a concrete data value. In addition to URIs and literals, ontologies have unnamed entities known as blank nodes. The methods applicable to vocabulary and their evaluation in the U Ontology are presented in the following subsections.

10.3.1 Method 1 : Check used protocols

This method is used to check the protocol used in the ontology. The definition of the method is as follows:

Method 1 (Check used protocols)

All URIs in the ontology are checked to be well-formed URIs. The evaluator has to choose a set of allowed protocols for the evaluation task. The usage of any protocol other than HTTP should be explained. All URIs in the ontologies have to use one of the allowed protocols.

Most names in ontologies are URI references (generic form of URLs) ([Berners-Lee et al., 1998](#)). URI references are strings that start with protocols. The recommended protocol for the URIs is HTTP as this allows applications (or even ontologies) to resolve the URIs. Resolving the URI means providing more information about the identified resource. The Linked Data principle recommends using the HTTP protocol

in URIs for dereferencing.

Evaluation : In the U Ontology, all the URIs use the HTTP protocol and thus are resolvable. Each URI is dereferencable and provides textual description for human readability.

```

1  <owl:Class rdf:about="http://oua.uontology.org/v1#FormalLabel">
2      <rdfs:subClassOf rdf:resource="http://oua.uontology.org/v1#LabelUsage"/>
3  </owl:Class>
4
5
6  <owl:ObjectProperty rdf:about="http://oua.uontology.org/v1#hasRichness">
7      <rdfs:range rdf:resource="http://oua.uontology.org/v1#ConceptRichness"/>
8      <rdfs:domain rdf:resource="http://oua.uontology.org/v1#ConceptUsage"/>
9      <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
10 </owl:ObjectProperty>
11

```

Figure 10.1: The URI and HTTP protocol used in the U Ontology

Figure 10.1 shows a snippet of the U Ontology in which it can be seen that URI references (i.e. `http://oua.uontology.org/v1#FormalLabel` ; Line 2) for the terms (e.g. `FormalLabel` class and `hasRichness` property) use the HTTP protocol.

Conclusion : Verified

10.3.2 Method 2 : Check response codes

This method checks the response code of the HTTP request. The definition of the method is as follows:

Method 2 (Check response codes)

For all HTTP URIs, make a HEAD call (or GET call) on them. The response code should be 200 OK or 303 See Other. Names with the same slash namespace should return the same response codes, otherwise this indicates an error.

Resolving an HTTP URI reference returns an HTTP response code along with the content related to the referenced URI. There are a predefined set of codes with special meanings to interpret the codes and the appropriate code should be provided upon the HTTP GET request. There are two types of resources on the Web: information resources and non-information resources (Bizer et al., 2008).

- Information resources: When a URI identifying an information resource

is dereferenced, the server of the URI owner usually generates a new representation, a new snapshot of the information resource's current state, and sends it back to the client using the HTTP response code 200 OK.

- Non-information resources cannot be dereferenced directly. Therefore Web architecture uses a trick to enable URIs identifying non-information resources to be dereferenced: instead of sending a representation of the resource, the server sends the client the URI of an information resource which describes the non-information resource using the HTTP response code 303 See Other. This is called a 303 redirect. In the second step, the client dereferences this new URI and obtains a representation describing the original non-information resource.

Evaluation : In the deployment of the U Ontology, the server is configured to provide both HTTP Response 200 OK (for the information resource) and HTTP Response code 303 See Other (for non-information resources). The content negotiation for non-information resources is also possible, however the hash approach (URI dereferencing) is followed which is discussed in the next method.

Note : The current version of the ontology is deployed on the intranet and when the internet server is ready, the ontology will be moved to a live server.

Conclusion : Verified

Method 3 (Look up names))

For every name that has a hash namespace, make a GET call against the namespace. For every name that has a slash namespace, make a GET call against the name. The content type should be set correctly. Resolve redirects, if any. If the returned resource is an ontology, check if the ontology describes the name. If so, N is a linked data conformant name. If not, the name may be wrong.

HTTP URIs need to be dereference-able, meaning the HTTP client can look up the URI to retrieve the description of a resource. A URI identifying a real world object is different from a URI referring to a document describing the real world object. To avoid ambiguity, two separate URIs are used to identify them. Content negotiation is used to provide the HTML for humans and RDF for machines when a URI about a resource is dereferenced. There are two strategies to name non-information resources: 303 URI and hash URI. For example, for a non-information resource Thesis, 303 URI will be `http://example.org/ontology/Thesis`, and has URI it will `http://example.org/ontologyThesis`. Both approaches have advantages and disadvantages and should be considered when deciding on the appropriate approach.

Evaluation : The U Ontology uses HTTP URI Reference to name the resources (terms). In the implementation of an ontology, the hash URI approach is adopted based on the recommendation suggested in ([Heath and Bizer, 2011](#); [Lewis, 2007](#); [Berrueta and Phipps, 2008](#)). The advantages of using URI is that it is downloaded in one pass because when the hash URI is looked up, only the namespace is resolved and the fragment identifier is not sent to the server. This approach is adopted on the fact that the ontology size is small and will not be frequently changed and instance data is not going to increase quickly and frequently. Therefore, the disadvantage of using hash URI i.e. if the namespace description (ontology) consists of a large number of triples, the hash URI approach can lead to large amounts of data being unnecessarily transmitted to the client, is not applicable, at least in the near future . The advantage which is obtained using hash URI is the avoidance of unnecessary trips to the server (courtesy of content negotiation). In Figure 10.1, lines 6-9 shows that hash URIs are used for the names of resources.

Conclusion : Verified

10.3.4 Method 4 : Check Naming conventions

This method checks the naming convention used to name terms in an ontology. The definition of the method is as follows:

Method 4 (Check naming conventions)

A proper naming can be checked by comparing the local part of the URI with the label given to the entity or by using lexical resources like Wordnet (Fellbaum, 1998). Formalize naming conventions (like multi-word names and capitalization) and test if the convention is applied throughout all names of a namespace. Check if the URI fulfils the general guidelines for good URIs, i.e. check length, inclusion of query parameters, file extensions, depth of directory hierarchy, etc.

Note that only local names from the same namespace, not all local names in the ontology, need to consistently use the same naming convention, i.e. names reused from other ontologies may use different naming conventions.

The URI standard (Berners-Lee et al., 2005) states that the URI should be treated as opaque and no formal meanings should be associated to the URIs, aside from using the appropriate protocol. However, the Semantic Web community and Linked Data best practices recommend using meaningful URIs which invoke certain denotation to help human users make some sense out of the URI. Using a pure opaque URI such as `http://example.org/abd1234` does not provide any clue about the resource it identifies whereas `http://example.org/student/JamshaidAshraf` provides some clue that this resource is about a person. There are naming conventions which a URI should follow in order to allow human users establish some understanding about the resource it denotes. There should be consistency in following naming resources and common practices wherever possible should be adopted. For example, using camel casing for multi-word names (e.g. `JamshaidAshraf` instead of `Jamshaid_ashraf` or `jamshaidashraf`).

Evaluation : The names used in the U Ontology follow the naming convention recommended by Sauermann and Cyganiak (2008) and Heath and Bizer (2011). The following conventions are adopted:

- The name used in the URI to denote the resource is closely matched to the labels given to the entity. This increases human readability and in the case where a consuming application uses it in the interface, the URI still communicates clues about the entity.

- Camel casing is used for multi-word names
- URIs do not contain any meta-data, technology clues, or query parameters as recommended in (Berners-Lee et al., 1998)
- Names for the class and properties are based on the naming convention followed in programming languages.

Figure 10.2 shows the excerpt taken from the U Ontology and will be used to demonstrate the implementation of a few of the methods. For example, regarding the naming convention line 37 in Figure 10.2, it defines a name searchEngineName which follows the camel casing and line 34 displays the name of the concept comprising two words (i.e. ConceptUsage).

Conclusion : Verified

10.3.5 Method 5 : Metrics of Ontology reuse

This method checks the reusability factor adopted in the ontology.

Method 5 (Metrics of ontology reuse)

We define the following measures and metrics:

- Number of namespaces used in the ontology N_{NS}
- Number of unique URIs used in the ontology N_{UN}
- Number of URI name references used in the ontology N_N (i.e. every mention of a URI counts)
- Ratio of name references to unique names $R_{NU} = N_{UN}/N_N$
- Ratio of unique URIs to namespaces $R_{UNS} = N_{UN}/N_{NS}$

Check the following constraints. The percentages show the proportion of ontologies that fulfill this constraint within the Watson EA corpus, thus showing the probability that ontologies not fulfilling the constraint are outliers.

$$R_{NU} < 0.5 (79 : 6\%)$$

$$R_{UNS} < 5 (90 : 3\%)$$

$$N_{NS} \geq 10 (75 : 0\%)$$

The reuse of names (terms) is highly recommended in the Semantic Web as it eases the sharing, exchange and aggregation of information. The reuse of terms defined in other ontologies has been observed in the literature, for example the terms defined in W3C-based vocabularies, terms in *foaf*, *dc*, *geonames* are reused by Semantic Web data publishers¹.

Evaluation: In the U Ontology, terms from different ontologies are reused to take advantage of their adoption and built-in support in several tools. The vocabularies from which the terms are reused are the Ontology Metadata Vocabulary (OMV), Ontology Application Framework, Dublin Core (DC), FOAF, and vCard. Details of the reused ontologies and terms are presented in Section 8.6.2.

Conclusion : Verified

10.3.6 Method 6 : Check name declaration

This method checks the presence of name declarations in the ontology

Method 6 (Check name declarations)

Check every URI to see if a declaration of the URI exists. If so, check if the declared type is consistent with the usage. This way it is possible to detect erroneously introduced punning.

Even though Web ontologies do not require names to be declared, it is recommended to declare them. Declaring them in an ontology helps reasoners to decide whether a name which appears to match with another name is a type or in fact a new resource.

Evaluation : In the U Ontology, all the names are declared before they are used in the definition.

Line 20 in Figure 10.2 declares the name of the `AttributeValue` class. Similarly, every URI to name the ontology components (class, relationship, and attribute) is explicitly declared before being used in the ontology.

Conclusion : Verified

¹Updated statistics on the use of different terms in the LOD cloud are available at <http://stats.lod2.eu/stats>; retr. 15/01/2013

10.3.7 Method 7: Check Literals and data type

This method checks if set of allowed data types is used.

Method 7 (Check literals and data types)

A set of allowed data types should be created. All data types beyond those recommended by the OWL specifications should be avoided. There should be a very strong reason for creating a custom data type. `xsd:integer` and `xsd:string` should be the preferred data types (since they have to be implemented by all OWL conformant tools).

Check if the ontology uses only data types from the set of allowed data types.

All typed literals must be syntactically valid with regard to their data type.

The evaluation tool needs to be able to check the syntactical correctness of all allowed data types.

In order to describe the states of the entities, literals are used. Literals provide factual statements about the resource. For example, Jamshaid Ashraf is the author of this Thesis. So Jamshaid Ashraf is the literal value describing the author of the Thesis resource. In ontologies, literals are typed which means each literal has a data type associated to it which tells what data value is expected for the literal. Even though Semantic Web standards allow new custom data types to be defined, they should be avoided wherever possible.

Evaluation : In the U Ontology, no custom data type is used and only the data types which are part of the standard types are used. Most of the textual properties are of type `xsd:string` and numbers are of type `xsd:integer`. The other data types which are used are `xsd:anyURI`, `xs:dateTime`, and `xsd:int`.

Figure 10.2 shows the declaration of data properties and the data type from the allowed (recommended) data types. For example, in line 37, `searchEngineName` data type properties are declared to be of type `rdfs:Literal`.

Conclusion : Verified

10.3.8 Method 8 : Check Language tag

This methods checks the use of language tags with literals.

Method 8 (Check language tags))

Check that all language tags are valid with regard to their specification. Check if the shortest possible language tag is used (i.e. remove redundant information such as restating default scripts or default regions). Check if the stated language and script is actually the one used in the literal.

Check if the literals are tagged consistently within the ontology. This can be checked by counting n_l , the number of occurrences of language tag l that occurs in the ontology. Roughly, n_l for all l should be the same. Outliers should be inspected.

Language tags can be used with plain literals (textual information) to tell tools which human language is used for literal values. The availability of different language tags, for example English and Arabic, would allow tools to be displayed based on the users local preferred language.

Evaluation : In the present version of the U Ontology, only the @en (English) language tag is used for the textual description of entities, particularly for the values of `rdfs:label` and `rdfs:comment` properties. However, as part of the future work, textual description in other languages, particularly German, French, Chinese and Arabic is planned.

In Figure 10.2, the description of the `hasAttribute` property is described and the language tag (line 29) is used to let tools know in which natural language the description is provided.

Conclusion : Verified

10.3.9 Method 9 : Check labels and comments

This method checks the use of label and comment properties for entities.

Method 9 (Check labels and comments))

Define the set of relevant languages for an ontology. Check if all label and comment literals are language tagged. Check if all entities have a label in all languages defined as being relevant. Check if all entities that need a comment have one in all relevant languages. Check if the labels and comments follow the style guide defined for the ontology.

In order to allow humans to understand the ontology, its purpose and scope,

human readable names are provided to the entities (or terms). As a recommended practice, all the terms defined in the ontology should have labels and comments in the appropriate language, marked with the language tag.

Evaluation : As mentioned in the previous method, for each term defined in the U Ontology, the `rdfs:label` property is used to provide human readable names/a description of the term (entity). However, only the English language tag is used. In future work, aside from providing textual descriptions in other languages, the use of other textual properties, such as `skos:prefLabel` will be considered.

Conclusion : Verified

10.3.10 Method 10 : Check for superfluous blank nodes

This method checks the use of necessary and unnecessary blank nodes.

Method 10 (Check for superfluous blank nodes)

Tables 2.1 and 2.2 (c.f. [Vrandevcic \(2010\)](#)) list all cases of structurally necessary blank nodes in RDF graphs. Check every blank node to see if it belongs to one of these cases. Apart from these, no further blank nodes should appear in the RDF graph. All blank nodes which are not structurally necessary should be listed as potential errors.

Blank nodes are RDF features and are used to represent a node in the RDF graph without giving it an explicit name (that is, a URI). Such nodes can be internally referred but are not exposed to the external applications.

Evaluation: In the U Ontology, no blank nodes are used and in future versions will be avoided if possible.

Conclusion : Verified

10.4 U Ontology Evaluation : Syntax aspect

This aspect evaluates the syntax that is used to serialize the ontologies. There are several serialization syntax available, each with advantages and disadvantages. The other syntax-related issues discussed in this aspect are comment style, XML validation, and the creation of XML Schema.

10.4.1 Method 11 : Validating against an XML Schema

This method checks the syntax of the implemented ontology.

Method 11 (Validating against an XML schema)

An ontology can be validated using a standard XML validator under specific circumstances. In order to apply this, the ontology needs to be serialized using a pre-defined XML schema. The semantic difference between the serialized ontology and the original ontology will help in discovering incompleteness of the data (by finding individuals that were in the original ontology but not in the serialized one). The peculiar advantage of this approach is that it can be used with well-known tools and expertise.

The conceptual model of an ontology is implemented using different serialization formats. Generally, there are two types of serialization syntax; one that describes a graph (e.g RDF/XML (Beckett, 2004), N triple (Grant et al., 2002)), and another that describes the ontology directly (e.g. Manchester Syntax (Horridge et al., 2006), OWL Abstract Syntax (Patel-Schneider et al., 2004), or the OWL XML Syntax (Hori et al., 2003)). Two main observations are made pertaining to the syntactical approach followed in ontology serialization: (a) RDF-style comments should be used rather than XML-style comments; (b) preferably the prefixes (qualified names) which are already adopted by the community to refer the ontologies in ontology serialization should be used. Additionally, the XML validation should be performed on an ontology to verify its conformance to the serialized syntax ontology on which it is built.

Evaluation: The U Ontology is serialized using RDF/XML syntax which is essentially based on XML Syntax. XML validation is performed to validate the syntax by the ontology development tool. Both XML style comments (e.g. `<!-- this is xml style -->`) and RDF style comments (e.g. a triple with RDF comments (e.g. `<Jamshaid> rdfs:comment "Jamshaid is a PhD student")`) are used in the ontology. As pointed out in (Vrandevcic, 2010), often XML style comments are lost

when two ontologies are merged or mapped, therefore RDF comments should be used. It is important to note that RDF specification as such do not have a standard way to provide in-line comments in the RDF document (similar to XML comments `<!-- text -->`) however, the community has adopted the use of the "#" to mark the followed text in the line as a comment. In the U Ontology, "#" based comments were not used, only XML style comments were used and where possible `rdfs:comment` properties were used.

Conclusion : Verified

10.5 U Ontology Evaluation : Structural aspect

This aspect evaluates the structural properties of the ontology graph. Ontologies are built using RDF which is a graph model. To analyse the ontologies structurally, several measures are proposed which offer following advantages:

- Structural measures are easy to compute from the ontology graph
- The majority of tools provide support for structural measures
- The structural measures are quantifiable and easy to interpret and visualise
- The structural measures results are programmable, interchangeable and interoperable

Different components of the ontology graph are observed which includes but is not limited to subsumption hierarchy, semantic similarity and pattern recognition.

10.5.1 Method 12 : Ontology Complexity

This method checks the structural characteristics of an ontology.

Method 12 (Ontology complexity)

We define measures counting the appearance of each ontology language feature. We do this by first defining a filter function $O_T : O \rightarrow O$ with T being an axiom or an expression type. O_T returns all the axioms of axiom type T or all axioms having an expression of type T .

We can further define a counting metric $N_T : O \rightarrow \mathbb{N}$ as $N_T(O) = |O_T(O)|$.

We also define $N(O) = |O|$.

We can then further define a few shortcuts, derived from the respective letters defining DL languages, for example:

- Number of subsumptions $N_{\sqsubseteq}(O) = N_{SubClassOf}(O) = |O_{SubClassOf}(O)|$: the number of subsumption axioms in the ontology
- Number of transitives $N_{+}(O) = N_{TransitiveProperty}(O)$: the number of properties being described as transitive
- Number of nominals $N_O(O) = N_{OneOf}(O)$: the number of axioms using a nominal expression
- Number of unions $N_{\sqcup}(O) = N_{UnionOf}(O)$: the number of axioms using a union class expression
- etc.

With these numbers we can use a look-up tool such as the description logics complexity navigator. If $N_O > 0$, then the nominals feature has to be selected, if $N_{+} > 0$ we need to select role transitivity, etc. The navigator will then give us the complexity of the used language fragment (as far as known).

We further define $H(O) : O \rightarrow O$ as the function that returns only *simple subsumptions* in O , i.e. only those `SubClassOf` axioms that connect two simple class names.

Structural measures obtained from the ontology graph can provide a number of interesting features, such as the richness of the concepts, the depth of hierarchies and the complexity of reasoning over the graph, since ontologies are developed using OWL languages and the complexity of these languages is known depending on their expressivity. However, the developed ontology's complexity cannot be simply judged through the used OWL language since it provides the upper bound, therefore knowing which constructs are used is important.

Evaluation : The U Ontology is developed using the OWL DL syntax. The ontology makes use of RDFS elements which are a subset of the closure of OWL DLP. This allows the ontology to be reasoned using a lightweight RDFS style (RDFS entailment rules) reasoner. As mentioned in (Hepp, 2008), the use of RDFS elements allows RDFS style reasoners to compute practically relevant inferences with the knowledge base and other ontologies without making the ontology OWL Full. The metrics defined to ensure ontology complexity are :

- Number of subsumptions = 10
- Number of transitives = 0
- Number of nominals = 0
- Number of unions = 0

Based on the above discussion and the metrics value, the U Ontology remains within the OWL-DL expressivity which makes it an ideal ontology for the Web as using it in knowledge bases and other ontologies would not make it OWL-Full.

Conclusion : Verified

10.5.2 Method 13 : Searching for Anti-Patterns

This method checks the presence of certain patterns in the ontology.

Method 13 (Searching for Anti-Patterns))

SPARQL queries over the ontology graph can be used to discover potentially problematic patterns. For example, results to the following queries have been found to be almost always problematic.

Detecting the anti-pattern of subsuming nothing:

```
select ?a where {  
  ?a rdfs:subClassOf owl:Nothing .  
}
```

Detecting the anti-pattern of skewed partitions:

```
select distinct ?A ?B1 ?B2 ?C1 where {  
  ?B1 rdfs:subClassOf ?A .  
  ?B2 rdfs:subClassOf ?A .  
  ?C1 rdfs:subClassOf ?B1 .  
  ?C1 owl:disjointWith ?B2 .  
}
```

Similar to object-oriented language, there are ontology design patterns to formalize common configuration of ontologies. Some of the patterns help in designing more useful ontologies as they are based on tested and trusted practices, whereas there are several patterns which need to be avoided. These patterns cause ontologies to fail or create problems at later stages when usage increases. Ontology design patterns were proposed by [Presutti et al. \(2008\)](#); [Blomqvist et al. \(2009\)](#) and their implementation was discussed in detail.

Evaluation : Different patterns are checked by posing a SPARQL query to the U Ontology to verify the inclusion or exclusion of certain patterns. In addition to the queries presented in ([Vrandevcic, 2010](#)), the queries used in our previous work ([Ashraf et al., 2011](#)) are also used to identify the patterns which are not recommend in the ontologies and knowledge base.

While certain patterns should be used in an ontology as they guarantee good results, several anti-patterns are also important to consider. The presence of anti-patterns cues the presence of a problem in the ontology. Figure 10.3 displays a SPARQL query to detect an anti-pattern of subsuming nothing in the U Ontology. The query did not find the presence of such a pattern in the ontology.

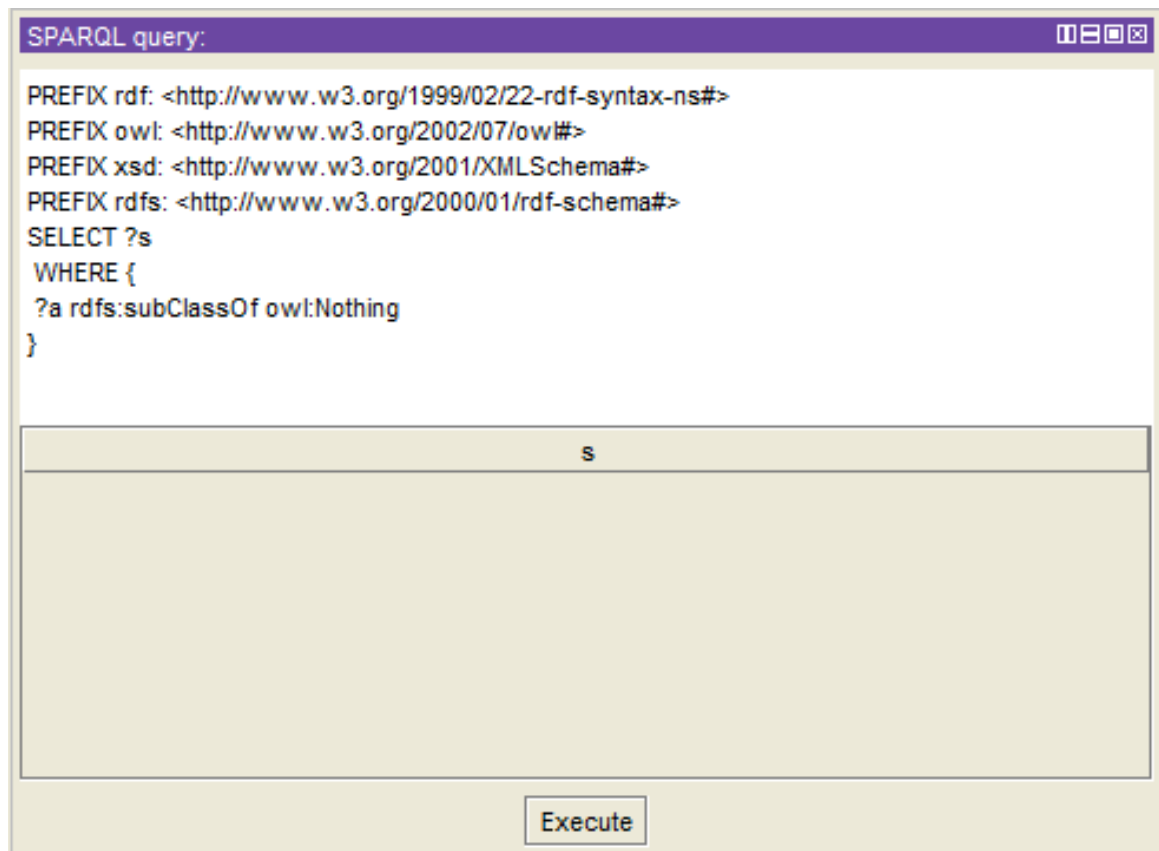


Figure 10.3: SPARQL query to identify an anti-pattern in the U Ontology

The other anti-patterns which were considered for detection in the ontology are:

```
select distinct ?A ?B1 ?B2 ?C1 where {
  ?B1 rdfs:subClassOf ?A .
  ?B2 rdfs:subClassOf ?A .
  ?C1 rdfs:subClassOf ?B1.
  ?C1 owl:disjointWith ?B2.
}
```

```
Select distinct ?ind
where{
  ?a owl:disjointWith ?b.
  ?ind ref:type ?a.
  ?ind rdf:type ?b.
}
```


None of the above anti-patterns were found in the U Ontology which ensures that the ontology does not have any issues associated with these anti-patterns.

Conclusion : Verified

10.5.3 Method 14 : OntoClean meta-property check

This method validates the ontology using OntoClean methodology.

Method 14 (OntoClean meta-property check)

An ontology can be tagged with the OntoClean meta-properties and then automatically checked for constraint violations. Since the tagging of classes is expensive, we provide an automatic tagging system AEON.

All constraint violations, i.e. inconsistencies in the meta-ontology, come from two possible sources:

- an incorrect meta-property tagging, or
- an incorrect subsumption.

The evaluator has to carefully consider each inconsistency, discover which type of error is discovered, and then either correct the tagging or redesign the subsumption hierarchy.

OntoClean ([Guarino and Welty, 2004](#)) is an ontology evaluation and validation methodology. It measures the adequacy of the ontology by analysing the taxonomic relationships present in the ontology. It makes use of philosophical notions such as rigidity, unity, dependency, and identity (known as OntoClean meta-properties) to formally analyse the classes and their subsumption hierarchies. While OntoClean has been well presented in conferences ([Guarino and Welty, 2002](#); [Olttramari et al., 2002](#)), documented and well-acknowledged by the community, it is still used infrequently due to its complexity in applications and limited support from tools (for annotation) ([Guarino and Welty, 2004](#)). There are a few tools in which OntoClean meta-properties are implemented such as WebODE ([Fernández-López and Gómez-Pérez, 2002](#)) and OntoEdit ([Sure et al., 2002b](#)).

Evaluation : For the validation of the U Ontology, OntoClean's meta-properties are not used since it takes a lot of effort to annotate all the contrast using meta-properties. The support of OntoClean in Protégé is minimal and the plug-in is not updated to the latest version of Protégé (i.e version 4.2). The steps suggested by

Protégé documentation² did not work as the required component was not available at the provided address, such as the PAL Constraint tab.

Conclusion : Not Applicable

10.6 U Ontology evaluation : Semantics aspect

The semantic aspect of an ontology measures the semantics of the ontology. In the previous aspect, the structure of the ontology was analysed by measuring the typological characteristics of the ontology without considering its semantics. In order to improve the overall quality of the ontology, it is important to measure the semantic aspects as well since the essence of ontologies is to carry and communicate the semantics of the domain of interest. Considering semantics along with the structure of the ontology allows taking RDFS/OWL semantics into consideration otherwise, generally ontology metrics consider the RDF graph model.

10.6.1 Method 15 : Ensuring a stable class hierarchy

This method checks the ontology hierarchies (incorporating the semantic aspect) to determine whether they are stable or not.

Method 15 (Ensuring a stable class hierarchy)

Calculate a normalized class depth measure, i.e. calculate the length of the longest subsumption path on the normalized version of the ontology $md(N(O))$. Now calculate the stable minimal depth of the ontology $md^{min}(O)$. If

$$md(N(O)) \neq md^{min}(O)$$

then the ontology hierarchy is not stable and may collapse.

In order to incorporate the semantic in metrics, the explicit model (structure) of the ontology should not be considered, rather all the models that are entailed from the ontology should be considered. This means that metrics need to be based on the implicit statements and not on the explicit statements, because the derived implicit statements represent the coverage of the domain knowledge conceptualized by the

²<http://protege.stanford.edu/ontologies/ontoClean/ontoCleanOntology.html>; retr. 12/1/2013

ontology. Therefore, reasoners are used to measure semantics and a normalization technique is used to obtain the stable metrics (Vrandečić and Sure, 2007).

Evaluation : In order to measure the stability of the class hierarchy in the U Ontology, a U Ontology Lite (UOT) is created to apply different normalization steps. The UOT contains two classes: `OntologyUsage` and `ConceptUsage` and properties. In the first normalization step, the anonymous classes are removed and in the case of UOT, there was no such anonymous class present. In the second step, anonymous individuals are removed. In UOT, no blank node is present and all the individuals were names with URI references. In the third step, subsumption hierarchies are materialised and to do this, RDFS entailments rules were used to generate the new statement materializing the subsumption relationship. In the fourth step of normalization, all the concepts and properties are instantiated. This means the instance of classes and properties are populated. In the fifth step which is similar to the third step, properties are materialized. The normalized class depth and the stable minimal path of OOT generated 2 and 2 respectively but since these measures are obtained on the smaller version of the ontology, they cannot be applied to the U Ontology. The measurement of these metrics and evaluation of this method will be considered in future work.

Conclusion : Deferred

10.6.2 Method 16 : Measuring language completeness

This method measures the language completeness of the ontology.

Method 16 (Measuring language completeness)

We define a function Υ_i with the index i being a language fragment (if none is given, the assertional fragment is assumed) from an ontology O to the set of all possible axioms over the signature of O given the language fragment i . We introduce C_i as language completeness over the language fragment i .

$$C_i(O) = \frac{|\{X \mid X \in \Upsilon(O), O \models X \vee O \models \neg X\}|}{|\Upsilon(O)|}$$

Given the set of names (URI references) in an ontology, language completeness measures the ratio between the knowledge that can be expressed and the knowledge that is stated.

Evaluation : When measuring language completeness, it is important to

understand the semantic aspects of the ontology, however, since the U Ontology does not have complex axioms (except a `rdfs:subClassOf`), an evaluation of this metric will not offer usable insight. Therefore, this metric is not considered in the evaluation of the U Ontology.

Conclusion : Not Applicable

10.7 U Ontology evaluation : Representation aspect

The representation aspect of an ontology analyses how the semantics of the ontology are structurally represented. This aspect helps to identify mistakes which may arise between the formal specification and the conceptualization. It is possible that the semantics of the ontology is structurally represented in more than one way and the need to obtain a normalized version arises to ensure the sub-model of the ontology has the same semantics. In (Lozano-Tello and Gomez-Perez, 2004; Vrandečić and Sure, 2007), the authors proposed metrics to measure the depth of the taxonomy and find the normalised model with the same semantics.

10.7.1 Method 17 : Explicitness of the subsumption hierarchy

This method identifies the relationships between the semantic and the structure of the ontology.

Method 17 (Explicitness of the subsumption hierarchy)

Calculate $ET(O)$.

- If $ET(O) = 1$ everything seems fine
- If $ET(O) < 1$ then some of the classes in the ontology have collapsed. Find the collapsed classes and repair the explicit class hierarchy
- If $ET(O) > 1$ part of the class hierarchy has not been explicated. Find that part and repair the class hierarchy

Using the ontology normalization (Vrandečić and Sure, 2007) functions, the maximum subsumption path length is computed and compared with the depth of the taxonomy. Note, here the taxonomy represents the normalized sub-model of the ontology offering the same semantics of the original sub-model (prior normalization). Vrandečić (2010) computes two metrics: maximum depth of the taxonomy (TD) of ontology O and maximum subsumption path length (SL) of the normalized version of

ontology O .

Evaluation : For this method $ET(O) = TD(O)/SL(O)$ is computed and the following measures are obtained

$$ET(UOntology) = 3/3 = 1$$

According to the definition of the metric, if both TD and SL are the same then it can be safely assumed that with the present structural representations of the ontology, there seems to be a balance in the taxonomy hierarchy and the semantics (shared conceptualization).

Conclusion : Verified

10.7.2 Method 18 : Explicit terminology ratio

This method identifies the explicit terminology ratio in the ontology.

Method 18 (Explicit terminology ratio)

Calculate $R_C(O)$ and $R_P(O)$.

- If $R_C(O) = R_P(O) = 1$, this indicates no problems with the coverage of elements with names in the ontology
- If $R_C(O) < 1$ or $R_P(O) < 1$ and the ontology does not include a mapping to an external vocabulary, this indicates possible problems since a number of names have collapsed to describe the same class
- If $R_C(O) < 1$ or $R_P(O) < 1$ and the ontology includes a mapping to an external vocabulary, we can remove all axioms providing the mapping and calculate $R_C(O')$ and $R_P(O')$ anew
- If $R_C(O) > 1$ or $R_P(O) > 1$, this indicates that not all interesting classes or properties have been given a name, i.e. the coverage of classes and properties with names may not be sufficient

This method is based on the measure labelled as M29 proposed by [Gangemi et al. \(2005b\)](#) called the Class / relations ratio which returns the ratio between classes and the relations in the ontology graph. For a given ontology, this means the number of nodes representing classes and the number of the nodes representing relations within the ontology graph.

Evaluation : As mentioned in the previous method (method 17), the $ET(UOntology) = 1$ therefore the value of $R_C(O) = |C_N(O)|/|C(O)| = 1$ (where $C_N(O) = 30$, and $C(O) = 30$) and $R_P(O) = |P_N(O)|/|P(O)| = 1$ (where $P_N(O) = 45$, and $P(O) = 45$) ratio between the normalized and not normalized ontology graph remains the same.

Conclusion : Verified

10.8 U Ontology evaluation : Context aspect

The context aspect in ontology evaluation refers to the identification and creation of the relevant artifact accompanying an ontology. The identified and created additional artifacts are then used by the evaluating tool to support the validation and verification process. These additional artifacts are the ones which specify the context of the ontology. One of the early approaches in providing the context is the use of competency questions (Uschold and Gruninger, 1996). Competency questions allow the evaluators to verify whether the developed ontology is able to answer all the issues raised in the competency question. To automate the verification process, competency questions need to be formally represented which is still not fully explored. Aside from this, certain constraints are imposed to verify the ontology. One of the latest approach in this regard is the use of a unit test in ontology evaluation (Vrandevcic, 2010).

10.8.1 Method 19 : Checking competency questions against results

This methods verifies the adequacy of an ontology using competency questions.

Method 19 (Checking competency questions against results)

Formalize the competency questions as a SPARQL query. Write down the expected answer as a SPARQL query result, either in XML or in JSON. Compare the actual and the expected results. Note that the order of results is often undefined.

Competency questions describe what kind of knowledge the resulting ontology is supposed to answer (Uschold and Gruninger, 1996). In order to automate the verification processes, the preferred approach is to formalize these competency questions instead of merely having them written down in natural language.

Evaluation : In order to identify the scope and requirements for the ontology usage domain, competency questions were presented in Section 8.4.2. The conceptual model of the U Ontology was developed based on these competency questions and formalized using the OWL language. The OUSAF framework is evaluated using the U Ontology in which SPARQL queries are generated, based on the competency questions presented in Section 8.4.2. The SPARQL queries are presented in Section 9.4, 9.5 and 9.6 for identification and empirical and quantitative analysis, respectively.

Conclusion : Verified

10.8.2 Method 20 : Checking competency questions with constraints

This method validated the ontology through competency questions with constraints.

Method 20 (Checking competency questions with constraints)

Formalize the competency questions for ontology O as a SPARQL CONSTRUCT query that formulates the result in RDF as an ontology R . Merge R with O and a possibly empty ontology containing further constraints C . Check the merged ontology for inconsistencies.

In the previous method, formalized competency questions were used in the form of SPARQL queries to verify the ontology. However, it is possible that when writing the competency questions, not all the requirements were either captured or known. Therefore, in order to verify whether the ontology will be able to accommodate the changes in future, competency questions with constraints are used to evaluate it.

Evaluation : For the evaluation of the U Ontology, this method is not considered. This method is helpful for ontologies which are dynamic in nature and require frequent changes. Changes are expected in the U Ontology but not on a regular basis, therefore this method is considered in future work.

Conclusion : Deferred

Method 21 : Unit testing with test ontologies This method validates the ontology using test ontologies.

Method 21 (Unit testing with test ontologies))

For each axiom A_i^+ in the positive test ontology T^+ test if the axiom is being inferred by the tested ontology O . For every axiom that is not being inferred, issue an error message. For each axiom A_i^- in the negative test ontology T^- test if the axiom is being inferred by the tested ontology O . For every axiom that is being inferred, issue an error message.

The concept of using a unit test is quite new in ontologies, however it has been extensively used in software development and testing. In the case of ontologies, a test ontology is used to verify if certain axioms can or cannot be derived from the ontology (Vrandečić and Gangemi, 2006).

Evaluation: As pointed out by Vrandevcic (2010), test ontologies are meant to be created and grown during the maintenance of the ontology. Every time an error is encountered in the usage of the ontology, the error is formalized and added to the appropriate ontology (as in the example above), therefore this method is not applicable with the current state of the ontology.

Conclusion : Not Applicable

10.8.3 Method 22 : Increasing expressivity

This model checks the consistency in expressiveness.

Method 22 (Increasing expressivity))

An ontology O can be accompanied by a highly axiomatized version of the ontology, C . The merged ontology of $O \cup C$ has to be consistent, otherwise the inconsistencies point to errors in O .

In the case of information systems, it is often required that the reasoner should provide the required information in less time with regard to ontologies. Ontologies which are lightweight in their design (this means they do not use constructs which make ontology reasoning undecidable) are the preferred choice for information systems and are often recommended for Web usage.

Evaluation: The U Ontology is evaluated for inconsistencies and disjoint violations using a reasoner. The FaCT++ (Tsarkov and Horrocks, 2006) reasoner and an RDFS-based reasoner implemented by the Virtuoso server (OpenLink Software,

2009) is used to verify all the axiomatic triples implemented in the U Ontology. During verification, no violation or inconsistency is reported. However, reasoner found the presence of unsupported datatypes³ which was fixed with a supported datatype i.e. `dateTime`.

Conclusion : Verified

10.8.4 Method 23 : Inconsistency checks with rules

This method check the presence of inconsistencies in an ontology with the help of rules.

Method 23 (Inconsistency checks with rules))

Translate the ontology to be evaluated and possible constraint ontologies to a logic program. This translation does not have to be complete. Formalize further constraints as rules or integrity constraints.

Concatenate the translated ontologies and the further constraints or integrity constraints. Run the resulting program. If it raises any integrity constraints, then the evaluated ontology contains errors.

The consistency checks in ontologies can be verified by making use of logical rules expressed using some formalism, for example, SWRL (Horrocks et al., 2004) and RDFS entailment rules (ter Horst, 2005). With the help of the expressivity of these languages, the context ontologies are not limited to OWL languages, therefore customised rules can be used for verification.

Evaluation : As mentioned earlier, the FaCT++ reasoner is used to validate the U Ontology and identify inconsistencies, if any.

Conclusion : Verified

10.9 Summary of U Ontology Evaluation

In the abovementioned sections, using the methods specified by Vrandevcic (2010), the U Ontology is evaluated from multiple aspects. For each method, based on the results obtained, an overall evaluation is concluded by using the metrics defined in Section

³ReasonerInternalException: Unsupported datatype "http://www.w3.org/2001/XMLSchema#dateTimeStamp"

10.2.3. In order to quantify the U Ontology evaluation and provide conclusively remarks about the results, the summarized values of these predefined metrics are shown in Figure 10.4.

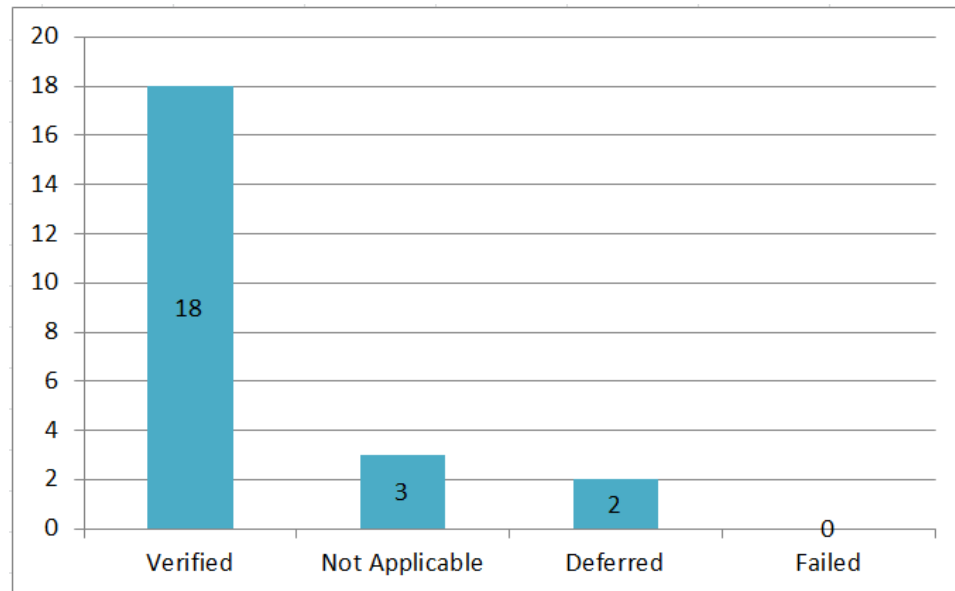


Figure 10.4: Summary of U Ontology Evaluation

As shown in Figure 10.4, of the 23 methods, 18 methods were Verified to meet the expectations of the methodology proposed by [Vrandevcic \(2010\)](#). Of the five methods which were not evaluated as Verified, three were Not Applicable because of a lack of technical support in applying these methods. For example, Method 14 (OntoClean meta-property check) requires the use of meta-properties defined by OntoClean to annotate the ontology to measure the rigidity, unity, dependency and identity, while manual annotation with the OntoClean meta-property is impractical, and built-in automatic annotation support in ontology tools is limited. Therefore, this method was marked as Not Applicable. Two of the five methods not verified are classified as Deferred which means that due to the complexity and extensive resource requirements needed to verify these methods, they are considered for future work. However, none of the methods failed in terms of not conforming to methodology expectations or not representing the required behaviour and characteristic.

Conclusively, based on the above summarized evaluation of the U Ontology, it can be stated that the ontology represents the required quality standards, and possesses the expected structural and semantic characteristics.

10.10 Conclusion

The objective of this chapter was to evaluate the U Ontology to ensure that the developed ontology is of an acceptable quality. Evaluation methodology, proposed by [Vrandevcic \(2010\)](#), is used for the U Ontology. This methodology is comprised of 23 methods (acting as the specification of the gold standard) which evaluate the ontology from six aspects. The U Ontology is analysed against each method and based on the obtained observations, a metric is assigned. Each metric is then summarized and plotted on a chart to provide a summary of the U Ontology evaluation.

In the next chapter, the work presented in this thesis is recapitulated to summarize the research problem, issues and the proposed solution.

Chapter 11 - Recapitulation and Future Work

11.1 Introduction

In the literature, a significant amount of work has been done on knowledge representation on the Web which includes ontology development (ontology engineering), formal languages to represent them, methodologies to evaluate and evolve them, and logic formalism for reasoning with them. As a result, numerous domain ontologies have been developed to describe information pertaining to different domains such as Health Care and Life Science (HCLS), the public sector, social spaces, libraries, entertainment, financial services and eCommerce. Consequently, we are witnessing a huge growth of structured data on the Web that is semantically described using domain ontologies. However, while there are billions of RDF triples and thousands of ontologies published on the Web, there is *no formal approach to evaluate, measure, and analyse the use of ontologies on the Web*. Such a study is very important to realize the following benefits:

- Make effective and efficient use of formalized knowledge (ontologies) available on the Web
- Provide a usage feedback loop to the ontology maintenance process for pragmatic conceptual model update
- Provide erudite insight on the state of semantic structured data based on the prevalent knowledge patterns for the consuming application

In order to realize the abovementioned benefits of ontologies, in this thesis, the OUSAF framework is proposed that provides pragmatic feedback and analysis of the

use of domain ontologies on the Web. In the next section, the issues that have been addressed in this thesis are recapitulated. In Section 12.3, the contribution made to the literature by successfully addressing the identified issues and proposing the OUSAF framework is discussed. In Section 10.4, areas for future work are identified and Section 10.5 concludes the chapter.

11.2 Recapitulation

The Semantic Web (also known as the Web of Data) is growing rapidly and becoming a decentralized social and knowledge platform for publishing and sharing information. Significant events such as the simplicity of Linked Data principles, the success of the Linked Open Data project and the recognition and consumption of Semantic Web data by search engines have given momentum to the widespread adoption of Semantic Web technologies, and therefore, to the use of ontologies. Ontologies are being used in many ways such as semantic annotation, information interoperability, data integration and knowledge assimilation by different type of users to achieve the potential benefits offered through the use of ontologies. Now, with the standardization of Semantic Web technologies and the increasing adoption and uptake of ontologies, a mechanism is required to understand how ontologies are being used in the real world setting. The mechanism needs to empirically analyse and quantitatively measure the use of domain ontologies in order to achieve the perceived benefits described in Section 10.1. As highlighted in Chapter 2, the current literature mainly focuses on evaluating ontologies without considering the usage aspect of those ontologies. Most ontology evaluation approaches analyse the correctness of the developed ontologies by evaluating their structural characteristics without considering how these ontologies are being adopted and utilized. Failure to consider this results in not having a feedback loop that is required for the effective utilization of Semantic Web data and for ontology development and maintenance. In this thesis, the problem of a lack of a formal approach to analysing the use of domain ontologies is addressed and divided into following sub-problems:

- 1) Define Ontology Usage Analysis as a focused (research) area in the ontology lifecycle, along with its role and utilization. The definition of Ontology Usage Analysis needs to specify the following details:

- 1.1 The relationships of ontology usage analysis with other ontology-related research areas such as ontology engineering, ontology evaluation, and ontology evolution.

-
- 1.2 The position (placement) of ontology usage analysis in the ontology (development) lifecycle model
 - 1.3 The anticipated role of ontology usage analysis and its utilization for different types of users such as ontology engineers, domain experts and application developers.
- 2) Propose an approach to carry out ontology usage analysis. The proposed approach needs to address the following requirements:
- 2.1 The analysis needs to be performed on empirical grounding to measure the usage based on ontology instantiation
 - 2.2 The approach needs to provide clear steps and the role of each step in analysing the usage of ontologies needs to be defined
 - 2.3 The proposed approach needs to analyse usage from different aspects to provide a comprehensive insight about the use of ontologies and their components
 - 2.4 The proposed approach should facilitate the utilization of usage analysis
- 3) Propose an approach to identify the ontologies from a given dataset or of a given application area (domain-specific). The proposed approach needs to cover the following requirements:
- 3.1 Provide steps and define the roles of each step involved in the identification of ontologies
 - 3.2 Provide flexibility in identifying the domain-specific ontologies and their relationships from a given dataset (or any input source)
 - 3.3 Capture the relationships between different ontologies emerging from their usage
 - 3.4 Measure the use of ontologies by different data sources (or data publishers) and analyse the publishing patterns and co-usability factors among different data sources
- 4) Propose an approach to empirically analyse the use of ontologies. The proposed approach needs to cover the following aspects:
-

-
- 4.1 Provide the necessary steps and define the roles of each step involved in empirically analysing the use of ontologies
 - 4.2 Cover the different aspect of usage analysis to obtain the required insight into their usage
 - 4.3 Analyse the use of different ontology components including concepts, relationships and attributes to quantify their usage
- 5) Propose an approach to quantitatively analyse the use of ontologies based on aspects identified by empirical analysis. The proposed approach needs to cover the following aspects:
- 5.1 Provide the necessary steps and define the roles of each step involved in the quantitative analysis of ontology usage.
 - 5.2 Measure ontology usage from different dimensions such as its structural, semantic and commercial aspects.
 - 5.3 Consolidate the quantified measures of the different dimensions to obtain a unified usage ranking.
- 6) Propose an approach to represent the output of usage analysis for further utilization. The proposed approach needs to address the following requirements:
- 6.1 Identify the key concepts for representing domain ontology usage analysis
 - 6.2 Develop and formalize a conceptual model representing the domain knowledge of ontology usage analysis
 - 6.3 Implement the conceptual model so it can be accessed by different types of users.
- 7) Evaluate the formal representation of the ontology usage analysis domain to realize the implementation of ontology usage analysis.

11.3 Contribution of the Thesis

The major contribution of this thesis to the existing literature is that it highlights the need, importance and proposes a methodological solution for ontology usage analysis. Ontology usage analysis aims at empirically and quantitatively measuring the use

of domain ontologies on the Web in order to understand and make effective use of information described through ontologies and provide a pragmatic feedback loop to the ontology development lifecycle. In order to position OUA and realize its benefits, a complete solution comprising of various definitions, methodologies, methods, metrics and a framework is presented in this thesis.

The definition, methodologies and framework proposed in this thesis are:

1. Conceptual definition of Ontology Usage Analysis (OUA) and its placement within the ontology development lifecycle model.
2. Methodological approach toward the implementation of semantic framework for measuring ontology usage analysis under the name of the Ontology Usage Analysis Framework (OUSAF).
3. Methodology for the identification of ontologies from a given dataset or application area
4. Methodology for empirically analysing the use of ontologies from different aspects
5. Methodology for quantitatively measuring the use of ontologies
6. Formalized conceptual model to represent the output of ontology usage analysis
7. Generation of the Web Schema based on the formalized output of ontology usage analysis.

The contributions made by each of the abovementioned points to the literature is briefly described as follows.

Contribution 1: Definition of Ontology Usage Analysis

Before defining ontology usage analysis, ontology engineering which encompasses development-related activities and the ontology lifecycle model is presented to provide the context to the problem being addressed in this thesis. Different ontology development methodologies and lifecycle models which are commonly referred to in the literature are presented to highlight the scope and functional detail of each respective approach.

The conceptual definition of ontology usage analysis is presented in Chapter 4 to describe the scope, role and primary function of OUA. The key terms that comprise

the definition of OUA are elaborated to provide a contextual description and avoid semantic ambiguity.

OUA is analysed and discussed with relevant areas such as ontology evaluation and ontology evolution to discuss the function and scope of OUA and highlight the subtle differences it has between other areas of ontology engineering. . The placement of OUA within the ontology lifecycle model and its relationships with other ontology engineering areas is described. The two main stages of the ontology lifecycle are described and OUA is placed in the runtime stage where ontologies are utilized.

To the best of my knowledge, ontology usage analysis as a research area and an important step in the ontology lifecycle is not discussed in the literature. The previous work in which ontologies specifically and Semantic Web data generally is analysed are from different perspectives. Ontologies have been analysed to study their structural and semantic characteristics however, as discussed in Chapter 2, they are not analysed from a usage perspective.

Contribution 2 : Methodological approach for the Implementation of OUA

The second contribution of this thesis is the development of a methodological approach to implement OUA. A semantic framework called the Ontology Usage Analysis Framework (OUSAF) is proposed and developed to measure and analyse the use of domain ontologies on the Web in Chapter 4. The methodological approach for OUSAF comprises four phases: identification, investigation, representation, and utilization for carrying out usage analysis. The objective of each phase along with their functional requirements is also explained.

To the best of my knowledge, there is no methodological approach proposed in the literature in which a framework is presented to measure the usage of ontologies in a real world setting. Crawled Semantic Web data (RDF data) has been used to analyse the presence of social networks, the quality of RDF data and the structural and semantic characteristics of RDF data, however, a framework to measure the use of ontologies on the Web has not been proposed.

Contribution 3 : Methodology to Identify Ontologies and their Co-usage

The third contribution of the thesis is the methodological approach for the identification phase of the OUSAF framework. A framework called the Ontology Usage Network Analysis Framework (OUN-AF) is developed for the implementation of the identification phase in Chapter 5. The OUN-AF framework comprises three

phases: the input phase, computation phase and analysis phase. The input phase is responsible for collecting the data for ontology identification. In the computation phase, the ontology usage network is constructed to provide the computational architecture for observing the relationships that ontologies have with data sources. In the analysis phase, different metrics are defined to observe the relationship between ontologies and the data sources using those ontologies.

In the literature, social network analysis (SNA) has been used to study the typological and structural characteristics of ontologies however, to the best of my knowledge, there is no approach proposed in which SNA is used to measure affiliation-based relationships based on their usage.

Contribution 4: Methodology to Empirically Analyse Ontology Usage

The fourth contribution of the thesis is the methodological approach for the investigation phase of the OUSAF framework. The investigation phase is implemented at two levels, firstly at an empirical level in which ontologies are empirically analysed, and secondly at a quantitative level, in which ontology usage is quantitatively analysed and measured. This contribution focuses on the first level, i.e performing an empirical analysis. An EMPirical Analysis Framework (EMP-AF) is proposed to empirically analyse the use of domain ontologies on the Web in Chapter 6. The EMP-AF framework comprises two phases: the data collection phase and the aspect analysis phase. In the data collection phase, data is collected from the Web and in the aspect analysis phase, usage is analysed from different aspects using the proposed metrics.

In the literature, several approaches have been proposed to empirically analyse RDF data, but their emphasis is more on measuring quality aspects but not from a usage aspect. Similarly for ontologies, empirical work has been done on evaluating ontologies but not from a usage perspective.

Contribution 5 : Methodology to Quantitatively Analyse Ontology Usage

The fifth contribution pertains to the development of the QUAntitative Analysis Framework (QUA-AF) for the quantitative level investigation phase of the OUSAF framework, presented in Chapter 7. In the QUA-AF framework, ontology usage is quantitatively measured from three dimensions to comprehensive analyse the use of ontologies on the Web. QUA-AF comprises three phases: the data collection phase, the computation phase, and the application phase. In the data collection phase, the

data is collected by crawling the Web. In the computation phase, for each dimension, a different set of metrics is defined and their results are then consolidated to obtain a unified rank of the usage. In the application phase, the QUA-AF framework is evaluated by using a use case scenario.

In the literature, different metrics have been proposed to quantify the ontologies' structural and typological characteristics but those purely consider ontologies only and not their usage.

Contribution 6 : Formal model for Representing Ontology Usage Analysis domain knowledge

The sixth contribution of the thesis is the formalization of ontology usage analysis domain knowledge for the representation phase of the OUSAF framework which is presented in Chapter 8. In order to make the ontology usage analysis results accessible to different types of users, a conceptual model is built to represent domain knowledge. The conceptual model is then formalized on UML which is a standard modelling language, and is implemented using a formal ontology language to generate the ontology artifact for population and further utilization.

To the best of my knowledge, no ontology or conceptual model is proposed in the literature which represents and conceptualises the domain knowledge of ontology usage analysis.

Contribution 7 : Generation of Web Schema based on the output of Ontology Usage Analysis

In order to demonstrate the output of the OUSAF framework and the benefits of OUA, a use case scenario is implemented. The use case scenario requires the generation of a Web Schema comprising the terminological knowledge representing the concepts describing the specific application area. The Web Schema representing the eCommerce domain is presented in Chapter 7 and 8 which is constructed based on the usage analysis performed on the collected dataset. The Web Schema enables data publishers to obtain a consolidated view of the currently used vocabularies with reasonable usage and consider them for their semantic annotation.

11.4 Future work

In this thesis, ontology usage analysis is introduced as an important area of work for measuring and evaluating the use of ontologies on the Web. Based on the methodological approach, the OUSAF framework is presented to realize the implementation of OUA. However, during the course of the work presented in this thesis, several important future directions have emerged. Considering these directions as future work would further strengthen the proposed methodology and framework and help integrate OUA within the ontology lifecycle model. Following are the high level areas for future work which have been identified for future work.

1. Expand the dataset for OUA that extends to other application domains
2. Publish the ontology usage analysis in the form of an Ontology Usage Catalogue
3. Explore other dimensions and aspects required for measuring ontology usage and provide support for reasoning over collected dataset.
4. Explore and incorporate other approaches for measuring incentives.
5. Explore further ontology evaluation methods to validate U Ontology.

Each of the possible directions for future work is briefly described in the following sub-sections.

11.4.1 Expand the dataset and Extend to other Application domains in order to provide (near to) real time Usage Analysis

Since more data provides more accurate results, the first possible direction is to expand the dataset. There are primarily two dimensions in which the dataset can be expanded, vertically and horizontally. With respect to vertical expansion, the dataset needs to be expanded to include a larger set of RDF data that can be reasonably representative of the actual instantiation of ontologies on the Web. Regarding horizontal expansion, the dataset needs to include the RDF data published in different application areas (domains) to provide a cross domain Semantic Web data corpus for usage analysis. Horizontal expansion will not only help in identifying the use of ontologies in different given application areas but will also provide an opportunity to observe the ontology usage patterns across different domains.

Aside from these expansions, the dataset can also be collected at different intervals of time to build a longitudinal dataset. A longitudinal dataset will help in determining whether the status quo remains unchanged pertaining to ontology adoption and usage patterns and, if not, how implementation develops with increased maturity.

The ideal situation for future work would be to build a streaming mechanism to provide near-to-real time data feeds to the ontology usage analysis framework to measure the up-to-date usage patterns to provide more accurate usage measures. The provision of such Semantic Web data streaming will not only refresh the dataset with new instances of ontology usage but will also improve the applicability of usage analysis because of their temporal status.

11.4.2 Publish the Ontology Usage Analysis in the form of Ontology Usage Catalogue

Any user, whether a publisher or consumer of Semantic Web data in general, would like to know which ontologies and what in those ontologies describe the entities relevant to the domain in focus. In order to allow data publishers or other ontology users to access the latest state of an ontology's adoption and usage, developing an ontology usage catalogue is proposed as future work. This includes creating a profile for each Web ontology to classify the set of entities it describes and the relationship of different entities to represent the overlapping domains. It would be desirable to build an Ontology Usage Catalogue representing the key entities of the particular domain in focus, and the usage level of ontology components.

The ontology usage catalogue for each domain ontology will help to provide consolidated terminological knowledge, representing the key entities constituting the application domain. Additionally, it can provide quantitative measures for the semantic (RDF) repositories which can use it to evaluate the axioms which need to be supported for reasoning, based on the actual usage data.

11.4.3 Explore other Dimensions and Aspects required for Measuring Ontology Usage and provide support for Reasoning over the collected dataset

In order to undertake a more specialized empirical and quantitative analysis of ontology usage, other aspects and dimensions need to be explored for their

consideration. It would be interesting to measure the use of RDFS and OWL standards in ontologies and instance data, and how the ontologies and instance data are interlinked and mapped. The defined metrics both in EMP-AF and QUA-AF can be further extended by incorporating the use of axiomatic triples. For example, while measuring the *RelationshipValue* (Chapter 6) of an object property, the concepts in the domain and range of property are calculated. However, it is possible that the object property has a sub-property axiom and the domain range value of sub-properties can be considered for measuring the richness value.

Similarly, the provision of subsumption axioms can be considered to explore the reasoning possible on the knowledge patterns reported by the usage analysis. Also, it would be interesting to know what kind of reasoning the defined semantics enables and how much can be obtained in the form of implicit knowledge from the explicit knowledge through reasoning.

11.4.4 Explore and Incorporate other approaches for Measuring Incentives

In Chapter 7, three dimensions are used to quantitatively analyse the use of ontologies. To incorporate the business dimension, the commercial incentives available to the particular ontologies are measured. Due to the lack of any formal study in the literature that quantifies the exact commercial benefits available to publishers, an Incentive measure which measures the benefits to the ontology (in the form of search visibility) in search engines was proposed. However, in future work, it needs to be extended to incorporate the other forms of incentives available to data publishers. One possible model could be the Financial Incentive Model to measure the financial benefits attained by implementers through the use of Web ontologies.

Similarly, a survey of data publishers can be conducted to learn about the factors which motivate them to use ontologies on the Web and based on the survey findings, an adoptive incentive model for measuring ontology usage can be developed.

11.4.5 Explore further ontology evaluation methods to validate U Ontology

In Chapter 10, U Ontology was evaluated using 23 methods. Two out of the 23 methods (see Figure 10.4) were not verified and classified as deferred due to their complexity and extensive resource requirements. However, as part of my future work these two methods will be implemented to evaluate U Ontology from these two aspects.

11.5 Conclusion

In this chapter, the work that has been undertaken and documented to address the identified research issues in the thesis has been recapitulated. The different contributions made to the literature through this thesis are presented. This was followed by a brief description of the future work that is intended to be considered in order to extend the approaches developed in this thesis. The work that was undertaken in this thesis has been published extensively as a part of the proceedings in peer reviewed international journals and conferences. This work (Ashraf, 2012) also received the **Best PhD Symposium Paper Award** at the Extended Semantic Web Conference, 2012 (ESWC2012)¹ of the work presented in this thesis are attached in Appendix C.

¹<http://2012.eswc-conferences.org/> (retr; 6/01/2013)

References

- Alani, H., Brewster, C., and Shadbolt, N. (2006). Ranking Ontologies with AKTiveRank. In *Proceedings of the 5th International conference on the Semantic Web (ISWS)*, volume 4273 of *Lecture Notes in Computer Science*, pages 1–15, Athens, Georgia. Springer-Verlag.
- Albert, R. and Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74:47–97.
- Almpanidis, G., Kotropoulos, C., and Pitas, I. (2007). Combining text and link analysis for focused crawling: An application for vertical search engines. *Information Systems*, 32(6):886–908.
- Amardeilh, F. (2006). OntoPop or how to annotate documents and populate ontologies from texts. In *Proceedings of the Workshop on Mastering the Gap: From Information Extraction to Semantic Representation (ESWC 2006)*, Budva, Montenegro.
- Aoyama, M. et al. (1998). New age of software development: How component-based software engineering changes the way of software development. In *1998 International Workshop on CBSE*.
- Ashburner, M., Ball, C., Blake, J., Botstein, D., Butler, H., Cherry, M., Davis, A., Dolinski, K., Dwight, S., and Eppig, J. (2000). Gene Ontology: Tool for the Unification of Biology. *Nature Genetics*, 25(1):25–29.
- Ashraf, J. (2011). List of datasources included in GRDS2 dataset (google document). <https://docs.google.com/spreadsheet/ccc?key=0AqjAK1TTtaSZdGpIMkVQUTRNenlrTGctR2J1bk16WEE> [Last access: 14/11/2012].
- Ashraf, J. (2012). A Framework for Ontology Usage Analysis. In Simperl, E., Cimiano, P., Polleres, A., Corcho, O., and Presutti, V., editors, *The Semantic Web: Research and Applications*, volume 7295 of *Lecture Notes in Computer Science*, pages 813–817. Springer Berlin Heidelberg.
-

- Ashraf, J., Cyganiak, R., O’Riain, S., and Hadzic., M. (2011). Open eBusiness Ontology Usage: Investigating Community Implementation of GoodRelations. In *Proceedings of Linked Data on the Web Workshop (LDOW) at WWW2011*, volume 813 of *CEUR Workshop Proceedings*, pages 1–11, Hyderabad, India.
- Asunción Gómez-Pérez, A., Fernández-López, M., and DE VINCENTE, A. (1996). Towards a method to conceptualize domain ontologies. In *ECAI-96 Workshop on Ontological Engineering*, pages 41–52, Budapest, Hungary.
- Atzeni, P., Mecca, G., and Merialdo, P. (1997). Semistructured and structured data in the Web: Going back and forth. *SIGMOD Record*, 26(4):16–23.
- Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. *The Semantic Web*, pages 722–735.
- Auer, S. and Lehmann, J. (2010). Creating knowledge out of interlinked data. *Semantic Web Journal*, 1(1):97–104.
- Baker, T. and Herman, I. (2009). Semantic web case studies and use cases. <http://www.w3.org/2001/sw/sweo/public/UseCases/>. (Last accessed 12/5/2012).
- Barabási, A. and Albert, R. (1999). Emergence of scaling in random networks. *science*, 286(5439):509–512.
- Barabási, A.-L., Albert, R., and Jeong, H. (2000). Scale-free characteristics of random networks: the topology of the world-wide web. *Physica A: Statistical Mechanics and its Applications*, 281(1):69–77.
- Batsakis, S., Petrakis, E., and Milios, E. (2009). Improving the performance of focused web crawlers. *Data & Knowledge Engineering*, 68(10):1001–1013.
- Beckett, D. (2004). RDF/XML Syntax Specification (Revised).
- Benjamin, P. C., Menzel, C. P., Mayer, R. J., and et. al, F. F. (1994). IDEF5 Ontology Description Capture Method Report. Knowledge based systems, Inc.
- Bergman, M. (2011). Making connections real . <http://www.mkbergman.com/941/making-connections-real/> [Last access: 14/11/2012].
- Berners-Lee, T. (1998a). Semantic Web Road map. *Design Issues for the World Wide Web*, 2008 (September 1998):1–10.
- Berners-Lee, T. (1998b). Web architecture from 50,000 feet. <http://www.w3.org/DesignIssues/Architecture.html> [Last access: 2002-08-20].
-

-
- Berners-Lee, T. (2006). Linked data - design issue. <http://www.w3.org/DesignIssues/LinkedData.html> [Last accessed, 12/1/2013].
- Berners-Lee, T., Fielding, R. T., and Masinter, L. (1998). Uniform Resource Identifiers (URI): Generic Syntax. Internet RFC 2396.
- Berners-Lee, T., Fielding, R. T., and Masinter, L. (2005). Uniform resource identifier (uri): Generic syntax. *Network Working Group*, 66(3986):1–61.
- Berners-Lee, T. and Fischetti, M. (1999). Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor. , San Francisco.
- Berrueta, D. and Phipps, J. (2008). Best Practice Recipes for Publishing RDF Vocabularies. W3C Working Group Note.
- Bishop, B., Kiryakov, A., Ognyanov, D., Peikov, I., Tashev, Z., and Velkov, R. (2011). FactForge: A fast track to the Web of data. *Semantic Web*, 2(2):157–166.
- Bizer, C., Cyganiak, R., and Heath, T. (2008). How to Publish Linked Data on the Web, <http://www4.wiwiss.fu-berlin.de/bizer/pub/LinkedDataTutorial/>.
- Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22.
- Bizer, C., Jentzsch, A., and Cyganiak, R. (2011). State of the Linked Open Data (LOD) Cloud. Technical Report. <http://www4.wiwiss.fu-berlin.de/lodcloud/state/>.
- Blomqvist, E., Gangemi, A., and Presutti, V. (2009). Experiments on pattern-based ontology design. In *Proceedings of the fifth international conference on Knowledge capture*, K-CAP '09, NY, USA., pages 41–48.
- Bock, J., Haase, P., Ji, Q., and Volz, R. (2008). Benchmarking OWL Reasoners. In van Harmelen, F., Herzig, A., Hitzler, P., Lin, Z., Piskac, R., and Qi, G., editors, *Proceedings of the ARea 2008 Workshop*, volume 350,
- Boehm, B. (1988). A spiral model of software development and enhancement. *Computer*, 21(5):61–72.
- Boehm, B. W. (1987). A spiral model of software development and enhancement. *Software Engineering Project Management*, pages 128–142.
- Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD international conference on Management of data*, SIGMOD '08, pages 1247–1250, NY, USA.
-

- Bollacker, K. D., Cook, R. P., and Tufts, P. (2007). Freebase: A shared database of structured general human knowledge. In *AAAI*, pages 1962–1963. AAAI Press.
- Borgatti, S. (2009). 2-Mode Concepts in Social Network Analysis. In Meyers, R., editor, *Encyclopedia of Complexity and Systems Science*. Springer.
- Borgatti, S. and Halgin, D. (2011). Analyzing Affiliation Networks. *The Sage handbook of social network analysis*, pages 417–433.
- Bozsak, E., Ehrig, M., Handschuh, S., Hotho, A., Maedche, A., Motik, B., Oberle, D., Schmitz, C., Staab, S., Stojanovic, L., Stojanovic, N., Studer, R., Stumme, G., Sure, Y., Tane, J., Volz, R., and Zacharias, V. (2002). Kaon - towards a large scale semantic web. In Bauknecht, K., Tjoa, A. M., and Quirchmayr, G., editors, *Proceedings of the Third International Conference on E-Commerce and Web Technologies EC-Web 2002*, volume 2455 of *Lecture Notes in Computer Science*, pages 304–313. Springer.
- Brank, J., Grobelnik, M., and Mladenić, D. (2005). A Survey of Ontology Evaluation Techniques. In *Proceedings of the Conference on Data Mining and Data Warehouses (SiKDD)*, pages 166–170, Ljubljana, Slovenia.
- Brass, D. J. (1985). Men’s and women’s networks: A study of interaction patterns and influence in an organization. *Academy of Management Journal*, 28(2):327–343.
- Braun, S., Schmidt, A., Walter, A., Nagypal, G., and Zacharias, V. (2007). Ontology maturing: a collaborative web 2.0 approach to ontology engineering. In Noy, N., Alani, H., Stumme, G., Mika, P., Sure, Y., and Vrandečić, D., editors, *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge (CKC 2007) at the 16th International World Wide Web Conference, Banff, Canada*, volume 273 of *CEUR Workshop Proceedings*.
- Breslin, J. G., Decker, S., Harth, A., and Bojars, U. (2006). SIOC: An Approach to Connect Web-Based Communities. *International Journal of Web Based Communities*, 2:133–142.
- Breslin, J. G., O’Sullivan, D., Passant, A., and Vasiliu, L. (2010). Semantic web computing in industry. *Computers in Industry*, 61(8):729 – 741.
- Brewster, C., Alani, H., Dasmahapatra, S., and Wilks, Y. (2004). Data Driven Ontology Evaluation. In *International Conference on Language Resources and Evaluation*, Lisbon, Portugal.
- Brickley, D. and Guha, R. V. (2004). Rdf vocabulary description language 1.0: Rdf schema. *W3C Recommendation*, <http://www.w3.org/TR/rdf-schema>, [Last accessed, 12/1/2013]
-

-
- Brickley, D. and Miller, L. (2004). Foaf vocabulary specification. Namespace Document, FOAF Project. <http://xmlns.com/foaf/0.1/>.
- Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A., and Wiener, J. (2000). Graph structure in the web. *Computer networks*, 33(1):309–320.
- Broekstra, J., Kampman, A., and van Harmelen, F. (2002). Sesame: A generic architecture for storing and querying rdf and rdf schema. In Horrocks, I. and Hendler, J., editors, *The Semantic Web : International Semantic Web Conference (ISWC)*, volume 2342 of *Lecture Notes in Computer Science*, pages 54–68. Springer Berlin / Heidelberg.
- Buitelaar, P., Eigner, T., and Declerck, T. (2004). OntoSelect: A Dynamic Ontology Library with Support for Ontology Selection. In *Proceedings of the Demo Session at the International Semantic Web Conference (ISWC) Demo Track*, Hiroshima, Japan.
- Caire, P. and van der Torre, L. (2010). Convivial Ambient Technologies: Requirements, Ontology and Design. *The Computer Journal*, 53(8):1229–1256.
- Carrol, J. and McBride, B. (2001). The Jena Semantic Web Toolkit. Public API, HP-Labs, Bristol. See <http://www.hpl.hp.com/semweb/jena-top.html>.
- Carroll, J. J., Bizer, C., Hayes, P., and Stickler, P. (2005). Named graphs, provenance and trust. In *Proceedings of the 14th international conference on World Wide Web, WWW '05*, NY, USA., pages 613–622
- Celjuska, D. and Vargas-vera, D. M. (2004). Ontosophie: A Semi-Automatic System for Ontology Population from Text. In: *International Conference on Natural Language Processing (ICON)*.
- Chakrabarti, S., Van den Berg, M., and Dom, B. (1999). Focused crawling: a new approach to topic-specific web resource discovery. *Computer Networks*, 31(11):1623–1640.
- Changrui, Y. and Yan, L. (2012). Comparative research on methodologies for domain ontology development. In Huang, D.-S., Gan, Y., Gupta, P., and Gromiha, M., editors, *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, volume 6839 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg., pages 349–356
- Chaudhri, V. K., Farquhar, A., Fikes, R., Karp, P. D., and Rice, J. (1998). Okbc: A programmatic foundation for knowledge base interoperability. In Mostow, J. and Rich, C., editors, *AAAI/IAAI*, pages 600–607. AAAI Press / The MIT Press.
-

- Cheng, G., Gong, S., and Qu, Y. (2011). An Empirical Study of Vocabulary Relatedness and Its Application to Recommender Systems. In *International Semantic Web Conference (ISWC)*, volume 7031 of *Lecture Notes in Computer Science*, Bonn, Germany. Springer., pages 98–113.
- Cheng, G. and Qu, Y. (2008). Term Dependence on the Semantic Web. In *Proceedings of 7th International Semantic Web Conference (ISWC)*, volume 5318 of *Lecture Notes in Computer Science*, Springer, Karlsruhe, Germany., pages 665–680
- Claudio, M., Nicola, G., Alessandro, O., and Luc, S. (2005). The wonderweb library of foundational ontologies. Wonderweb deliverable d18. <http://wonderweb.semanticweb.org/deliverables/documents/D18.pdf> [Last accessed, 12/12/2012]
- Clauset, A., Shalizi, C. R., and Newman, M. E. J. (2009). Power-law distributions in empirical data. *Society for Industrial and Applied Mathematics (SIAM) Review*, 51(4): pages 661–703.
- Cooley, R., Mobasher, B., and Srivastava, J. (1997). Web mining: Information and pattern discovery on the World Wide Web. In : *Proceedings of the 9th International Conference on Tools with Artificial Intelligence (ICTAI '97)*, Washington, DC, USA. IEEE Computer Society. pages 558 - 567
- Corcho, O., Fernández, M., Gómez-Pérez, A., and López-Cima, A. (2005). Building Legal Ontologies with METHONTOLOGY and WebODE. In Benjamins, R., Casanovas, P., Breuker, J., and Gangemi, A., editors, *Law and the Semantic Web*, LNAI, , Heidelberg, DE. Springer., pages 142–157
- Corcho, O., Fernández-López, M., and Gómez-Pérez, A. (2003). Methodologies, tools and languages for building ontologies: Where is their meeting point? *Data & Knowledge Engineering*, 46(1):41 – 64.
- Corcho, O., Fernandez-Lopez, M., and Gomez-Perez, A. (2007). Ontological engineering: What are ontologies and how can we build them? In Cardoso, J., editor, *Semantic Web Services: Theory, Tools and Applications*, chapter 03, pages 44–70. IGI Global.
- Corlosquet, S., Delbru, R., Clark, T., Polleres, A., and Decker, S. (2009). Produce and consume linked data with drupal! In Bernstein, A., Karger, D., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., and Thirunarayan, K., editors, *The Semantic Web - International Semantic Web Conference (ISWC)*, volume 5823 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg. , pages 763–778
-

- Cranefield, S. and Purvis, M. K. (1999). UML as an ontology modelling language. In *Intelligent Information Integration*, volume 23 of *CEUR Workshop Proceedings*.
- d'Aquin, M., Baldassarre, C., Gridinoc, L., Angeletou, S., Sabou, M., and Motta, E. (2007). Characterizing Knowledge on the Semantic Web with Watson. In *Proceedings of the 5th International Workshop on Evaluation of Ontologies and Ontology-based Tools(EON) at ISWC2007*, volume 329 of *CEUR Workshop Proceedings*, pages 1–10, Busan, Korea.
- d'Aquin, M. and Lewen, H. (2009). Cupboard - A Place to Expose Your Ontologies to Applications and the Community. In *Proceedings of the 6th European Semantic Web Conference (ESWC) on The Semantic Web: Research and Applications*, volume 5554 of *Lecture Notes in Computer Science*, , Heraklion, Crete, Greece. Springer Berlin / Heidelberg., pages 913–918
- d'Aquin, M. and Motta, E. (2011). Watson, more than a Semantic Web search engine. *Semantic Web*, 2(1):55–63.
- d'Aquin, M. and Noy, N. F. (2012). Where to Publish and Find Ontologies? A Survey of Ontology Libraries. *Web Semantics: Science, Services and Agents on the World Wide Web*, 11(0):96 – 111.
- Dasgupta, S., Dinakarpanthian, D., and Lee, Y. (2007). A Panoramic Approach to Integrated Evaluation of Ontologies in the Semantic Web. In *Proceedings of the 5th International Workshop on Evaluation of Ontologies and Ontology-based Tools(EON) at ISWC2007*, volume 329 of *CEUR Workshop Proceedings*, pages 31–40, Busan, Korea.
- David, J. and Euzenat, J. (2008). Comparison between Ontology Distances (Preliminary Results). In Sheth, A., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., and Thirunarayan, K., editors, *The Semantic Web - International Semantic Web Conferences (ISWC) 2008*, volume 5318 of *Lecture Notes in Computer Science*, pages 245–260. Springer Berlin / Heidelberg.
- Davies, S., Hatfield, J., Donaher, C., and Zeitz, J. (2010). User interface design considerations for Linked Data authoring environments. In *Proceedings of Linked Data on the Web Workshop (LDOW) at WWW2010*, volume 628 of *CEUR Workshop Proceedings*, Raleigh, USA.
- De Bruijn, J., Lara, R., Polleres, A., and Fensel, D. (2005). OWL DL vs. OWL flight: conceptual modeling and reasoning for the semantic Web. In *Proceedings of the 14th international conference on World Wide Web*, pages 623–632.
-

- Dean, M., Connolly, D., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., and Stein, L. A. (2002). OWL:Web Ontology Language 1.0 reference. Technical report, W3C Working Draft. <http://www.w3.org/TR/2002/WD-owl-ref-20020729/> [Last accessed, 17/12/2012]
- Decker, S., Erdmann, M., Fensel, D., and Studer, R. (1999). Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In *Proceedings of the IFIP TC2/WG2. 6 Eighth Working Conference on Database Semantics-Semantic Issues in Multimedia Systems*, Kluwer, BV. pages 351–369.
- Ding, L., DiFranzo, D., Graves, A., Michaelis, J., Li, X., McGuinness, D. L., and Hendler, J. (2010a). Data-gov wiki: Towards linking government data. In *AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*.
- Ding, L. and Finin, T. (2006). Characterizing the Semantic Web on the Web. In *International Semantic Web Conference*, volume 4273 of , Athens, GA, USA., *Lecture Notes in Computer Science*, pages 242–257.
- Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., Reddivari, P., Doshi, V., and Sachs, J. (2004). Swoogle: a Search and Metadata Engine for the Semantic Web. In *Proceedings of the 13th ACM international conference on Information and knowledge management*, NY. USA , pages 652–659
- Ding, L., Finin, T., Shinavier, J., and McGuinness, D. L. (2010b). owl:sameAs and linked data: An empirical study.
- Ding, L., Zhou, L., Finin, T., and Joshi, A. (2005). How the Semantic Web is Being Used: An Analysis of FOAF Documents. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, volume 4, Washington, DC, USA. , pages 113–120.
- Ding, Y. (2007). A simple picture of Web evolution. <http://www.zdnet.com/blog/web2explorer/a-simple-picture-of-web-evolution/408>. [Last retrieved 27/01/2013.]
- Ding, Y. and Fensel, D. (2001). Ontology library systems the key to successful ontology re-use. In *Proceedings of the 1st International Semantic Web Working Symposium (SWWS 2001)*, pages 93–112.
- Dodds, L. (2006). Slug: A semantic web crawler. In *Proceedings of Jena User Conference*, volume 2006.
-

- Dodds, L. and Davis, I. (2010). *Linked Data Patterns.*, <http://patterns.dataincubator.org/book/> [Last accessed, 31/01/2013]
- Doms, A. and Schroeder, M. (2005). Gopubmed: exploring pubmed with the gene ontology. *Nucleic acids research*, 33(suppl 2): pages 783-W786.
- Dong, H., Hussain, F., and Chang, E. (2008). A survey in semantic web technologies-inspired focused crawlers. In *Third International Conference on Digital Information Management, ICDIM 2008* , pages 934–936.
- Drucker, P. (1958). *The practice of management*. Allied Publishers.
- Ehrig, M. and Maedche, A. (2003). Ontology-focused crawling of web documents. In *Proceedings of the ACM symposium on Applied computing*, pages 1174–1178. ACM.
- Elkan, C. and Greiner, R. (1993). Building large knowledge-based systems: Representation and inference in the CYC project. *Artif. Intell.*, 61(1):41–52.
- Ell, B., Vrandečić, D., and Simperl, E. P. B. (2011). Labels in the Web of Data. In Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N. F., and Blomqvist, E., editors, *International Semantic Web Conference (1)*, volume 7031 of *Lecture Notes in Computer Science*, pages 162–176. Springer.
- Erétéo, G. (2011). *Semantic Social Network Analysis*. PhD thesis, Orange Labs Telecom ParisTech INRIA Sophia Antipolis à Méditerranée Karlsruhe Institute of Technology. Available Online at: <http://www.emilio.ferrara.name/wp-content/uploads/2011/06/thesis.pdf> (Last accessed 21/7/2012).
- Euzenat, J. (1995). Building consensual knowledge bases: Context and architecture. In Mars, N., editor, *Towards Very Large Knowledge Bases - Proceedings of the KB&KS '95 Conference*, pages 143–155. IOS Press.
- Euzenat, J. (1996). Corporate memory through cooperative creation of knowledge bases and hyper-documents, *Proceedings of 10th KAW*,(36), pages 1–18.
- Euzenat, J. (1997). A protocol for building consensual and consistent repositories. Technical Report RR-3260, INRIA., <http://hal.inria.fr/inria-00073429/> [Last accessed, 14/12/2012]
- Fellbaum, C., editor (1998). *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.
-

- Fensel, D., Bussler, C., Ding, Y., Kartseva, V., Klein, M., Korotkiy, M., Omelayenko, B., and Siebes, R. (2002). Semantic web application areas. In *Applications of Natural Language to Data Bases (NLDB) Workshop*.
- Fernández-López, M. and Gómez-Pérez, A. (2002). The integration of ontoclean in webode. CEUR Workshop Proceedings.
- Fernández-López, M., Gómez-Pérez, A., et al. (2002). Overview and analysis of methodologies for building ontologies. *The Knowledge Engineering Review*, 17(2):129–156.
- Fernandez-Lopez, M., Gomez-Perez, A., Euzenat, J., Gangemi, A., Kalfoglou, Y., Pisanelli, D., Schorlemmer, M., Steve, G., Stojanovic, L., Stumme, G., and Sure, Y. (2002). A survey on methodologies for developing, maintaining, integrating, evaluating and reengineering ontologies. OntoWeb deliverable 1.4, Universidad Politecnica de Madrid.
- Fernández-López, M., Gómez-Pérez, A., and Juristo, N. (1997). Methontology: from ontological art towards ontological engineering. In *Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering*.
- Fershtman, M. (1997). Cohesive group detection in a social network by the segregation matrix index. *Social Networks*, 19(3):193 – 207.
- Formica, A. and Missikoff, M. (2002). Concept Similarity in SymOntos: An Enterprise Ontology Management tool. *The Computer Journal*, 45(6):583–594.
- Fox, M. and Gruninger, M. (1998). Enterprise modeling. *AI magazine*, 19(3):109.
- Franceschet, M. (2011). Collaboration in computer science: A network science approach. *Journal of the American Society for Information Science and Technology*, 62(10):1992–2012.
- Freeman, L. C. (2003). Finding social groups: A meta-analysis of the southern women data. , *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*, National Academies Press.
- Fürber, C. and Hepp, M. (2010). Using SPARQL and SPIN for Data Quality Management on the Semantic Web. In Abramowicz, W. and Tolksdorf, R., editors, volume 47 of *Lecture Notes in Business Information Processing (BIS)*, pages 35–46. Springer.
-

- Fürber, C. and Hepp, M. (2011). Towards a vocabulary for data quality management in semantic web architectures. In *Proceedings of the 1st International Workshop on Linked Web Data Management*, pages 1–8.
- Galliers, R. D. (1992). *Information systems research: Issues, methods, and practical guidelines*, Alfred Waller, Henley-on-Thames, Oxfordshire.
- Gangemi, A., Catenacci, C., Ciaramita, M., and Lehmann, J. (2005a). A Theoretical Framework for Ontology Evaluation and Validation. In *Proceedings of 2nd Italian Semantic Web Workshop Semantic Web Application and Perspectives(SWAP)*, CEUR Workshop Proceedings, Trento, Italy.
- Gangemi, A., Catenacci, C., Ciaramita, M., and Lehmann, J. (2005b). Ontology evaluation and validation: An integrated formal model for the quality diagnostic task. Technical report, Laboratory of Applied Ontologies – CNR, Rome, Italy. http://www.loa-cnr.it/Files/OntoEval4OntoDev_Final.pdf. [Last accessed, 17/01/2013]
- Gangemi, A., Pisanelli, D. M., and Steve, G. (1999a). An overview of the onions project: Applying ontologies to the integration of medical terminologies. *Data and Knowledge Engineering*, 31(2):183 – 220.
- Gangemi, A., Pisanelli, D. M., and Steve, G. (1999b). An overview of the onions project: Applying ontologies to the integration of medical terminologies. *Data Knowl. Eng.*, 31(2):183–220.
- Gangemi, A., Steve, G., and Giacomelli, F. (1996). Onions: An ontological methodology for taxonomic knowledge integration, *ECAI-96 Workshop on Ontological Engineering*
- Garcia, A., O'Neill, K., Garcia, L. J., Lord, P., Stevens, R., Corcho, O., and Gibson, F. (2010). Developing ontologies within decentralised settings. In Chen, H., Wang, Y., Cheung, K.-H., Sharda, R., and VoÃ, S., editors, *Semantic e-Science*, volume 11 of *Annals of Information Systems*, Springer US. pages 99–139.
- Garcia, R. and Gil, R. (2009). Publishing XBRL as Linked Open Data. In Bizer, C., Heath, T., Berners-Lee, T., and Hausenblas, M., editors, *Proceedings of the Linked Data on the Web Workshop (LDOW) at WWW2009*, volume 538 of *CEUR Workshop Proceedings*, Madrid, Spain.
- Garton, L., Haythornthwaite, C., and Wellman, B. (1997). Studying online social networks. *Journal of Computer-Mediated Communication*, 3(1):1-8.
-

- Geleijnse, G. and Korst, J. (2005). Automatic Ontology Population by Googling. In *Proceedings of the Seventeenth Belgium-Netherlands Conference on Artificial Intelligence (BNAIC 2005)*, pages 120 – 126, Brussels, Belgium.
- Goh, C. H., Bressan, S., Madnick, S., Siegel, M., Hian, C., Stephane, G., Stuart, B., and Siegel, M. M. (1999). Context interchange: New features and formalisms for the intelligent integration of information. *ACM Transactions on Information Systems*, 17:270–293.
- Gomez-Perez, A. and Corcho, O. (2002). Ontology languages for the semantic web. *Intelligent Systems, IEEE*, 17(1):54 – 60.
- Gómez-Pérez, A., Fernández-López, M., and Corcho, O. (2004). *Ontological Engineering*. Springer-Verlag Heidelberg, Berlin.
- Gomez-Perez, A. and Suarez-Figueroa, M. C. (2009). Scenarios for building ontology networks within the neon methodology. In *Proceedings of the Fifth International Conference on Knowledge Capture (K-CAP 2009)*. pages183-184
- Gomm, R. (2004). *Social research methodology*. Palgrave Macmillan, New York.
- González, R. G. (2005). *A Semantic Web approach to Digital Rights Management*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spanien.
- Gradmann, S. (2005). rdfs:frbr – Towards an Implementation Model for Library Catalogs Using Semantic Web Technology. *Cataloging and Classification Quarterly*, 39(3-4):63–75.
- Graham, I., O’Callaghan, A., and Wills, A. (2001). *Object-oriented methods: principles & practice*, volume 6. Addison-Wesley.
- Graham, W. and Graham, W. (2012). Facebook developer tools. In *Beginning Facebook Game Apps Development*, Apress. pages 201–229.
- Grant, J., Beckett, D., and McBride, B. (2002). Rdf test cases: N-triples. *World Wide Web Consortium (W3C) working draf*, <http://www.w3.org/TR/rdftestcases/#ntriples>.
- Grover, V. and Kettinger, W. (2000). *Process Think: Winning Perspectives for Business Change in the Information Age*. IGI Global.
- Gruber, T. R. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2):199–220.
-

- Gruninger, M. and Fox, M. S. (1994a). The design and evaluation of ontologies for enterprise engineering. *Workshop on Implemented Ontologies, European Conference on Artificial Intelligence*.
- Gruninger, M. and Fox, M. S. (1994b). The role of competency questions in enterprise engineering. In *Proceedings of the International Federation for Information Processing (IFIP), WG5.7 Workshop on Benchmarking - Theory and Practice*, volume 7, pages 212–221.
- Grüninger, M. and Fox, M. S. (1995). Methodology for the Design and Evaluation of Ontologies. In *International Joint Conference on Artificial Intelligence (IJCAI95), Workshop on Basic Ontological Issues in Knowledge Sharing*.
- Guarino, N. and Welty, C. (2002). Evaluating ontological decisions with ontoclean. *Communications of the ACM*, 45(2):61–65.
- Guarino, N. and Welty, C. (2004). An Overview of OntoClean. In *Handbook on Ontologies*, Germany. Springer, pages 151–159
- Guebitz, B., Schnedl, H., and Khinast, J. (2012). A risk management ontology for quality-by-design based on a new development approach according gamp 5.0. *Expert Systems with Applications*. 39(8), pages 7291-7301.
- Guéret, C., Groth, P. T., van Harmelen, F., and Schlobach, S. (2010). Finding the achilles heel of the web of data: Using network analysis for link-recommendation. In Patel-Schneider, P. F., Pan, Y., Hitzler, P., Mika, P., 0007, L. Z., Pan, J. Z., Horrocks, I., and Glimm, B., editors, *International Semantic Web Conference (ISWC)*, volume 6496 of *Lecture Notes in Computer Science*, pages 289–304.
- Guizzardi, G. (2006). On ontology, ontologies, conceptualizations, modeling languages, and (meta)models. In Vasilecas, O., Eder, J., and Caplinskias, A., editors, *DB&IS*, volume 155 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, pages 18–39.
- Haase, P. and Stojanovic, L. (2005). Consistent Evolution of OWL Ontologies. In Gomez-Perez, A. and Euzenat, J., editors, *Proceedings of the Second European Semantic Web Conference*, volume 3532, Springer., pages 182–197, Heraklion, Crete, Greece.
- Haase, P., Volz, R., Erdmann, M., and Studer, R. (2007). Ontology engineering and plugin development with the neon toolkit. In *International Semantic Web Conference /Asian Semantic Web Conference (ISWC/ASWC) 2007 tutorial*.
-

-
- Harary, F. (1991). *Graph theory*. Addison-Wesley.
- Harth, A., Umbrich, J., and Decker, S. (2006). Multicrawler: A pipelined architecture for crawling and indexing semantic web data. *The Semantic Web-ISWC 2006*, pages 258–271.
- Hartmann, J., Sure, Y., Haase, P., Palma, R., and del Carmen Suarez-Figueroa, M. (2005). Omv – ontology metadata vocabulary. In Welty, C., editor, *ISWC 2005 - In Ontology Patterns for the Semantic Web.*, Galway, Ireland.
- Hassanzadeh, O., Kementsietsidis, A., Lim, L., Miller, R. J., and Wang, M. (2009). LinkedCT: A linked data space for clinical trials. *CoRR*, arXiv:0908.0567
- Hausenblas, M., Halb, W., Raimond, Y., and Heath, T. (2008). What is the Size of the Semantic Web? In *Proceedings of the International Conference on Semantic Systems (ISemantics)*, Universal Computer Science, pages 6–16, Graz, Austria.
- Hayes, P. (2004). RDF Semantics. Technical report. <http://www.w3.org/TR/rdf-mt/>.
- Heath, T. and Bizer, C. (2011). *Linked Data: Evolving the Web into a Global Data Space*. Synthesis Lectures on the Semantic Web. Morgan & Claypool.
- Hepp, M. (2007). Possible Ontologies: How Reality Constrains the Development of Relevant Ontologies. *IEEE Internet Computing*, 11(1):90–96.
- Hepp, M. (2008). GoodRelations: An Ontology for Describing Products and Services Offers on the Web. In *Proceedings of the 16th International conference on Knowledge Engineering: Practice and Patterns (EKAW)*, volume 5268 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg. , Sicily, Italy. , pages 329–346
- Herman, I. (2011). LDOW2011 workshop (blog post). <http://ivan-herman.name/2011/03/29/ldow2011-workshop> [Last access: 11/11/2012].
- Heymann, P. and Garcia-Molina, H. (2006). Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical Report, Stanford InfoLab. , <http://ilpubs.stanford.edu:8090/775/> [Last accessed. 31/1/2013]
- Hitzler, P. and van Harmelen, F. (2010). A Reasonable Semantic Web. *Semantic Web Journal*, 1(1–2):39–44.
- Hogan, A. (2011a). *Exploiting RDFS and OWL for Integrating Heterogeneous, Large-Scale, Linked Data Corpora*. PhD thesis, National University of Ireland, Galway, Ireland.
-

- Hogan, A. (2011b). *OWL for Integrating Heterogeneous, Large-Scale, Linked Data Corpora*. PhD thesis, Digital Enterprise Research Institute, National University of Ireland, Galway, available from <http://aidanhogan.com/docs/thesis>.
- Hogan, A., Harth, A., Passant, A., Decker, S., and Polleres, A. (2010). Weaving the Pedantic Web. In *Proceedings of Linked Data on the Web Workshop (LDOW) at WWW2010*, volume 628 of *CEUR Workshop Proceedings*, Raleigh, USA.
- Hogan, A., Harth, A., Umbrich, J., Kinsella, S., Polleres, A., and Decker, S. (2011). Searching and browsing Linked Data with SWSE: The Semantic Web Search Engine. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(4):365–401.
- Hogan, A., Zimmermann, A., Umbrich, J., Polleres, A., and Decker, S. (2012). Scalable and distributed methods for entity matching, consolidation and disambiguation over linked data corpora. *Journal of Web Semantics.*, 10:76–110.
- Hogg, T., Wilkinson, D. M., Szabó, G., and Brzozowski, M. J. (2008). Multiple Relationship Types in Online Communities and Social Networks. In *AAAI Spring Symposium: Social Information Processing*, pages 30–35. AAAI.
- Hori, M., Euzenat, J., and Patel-Schneider, P. F. (2003). OWL Web Ontology Language XML presentation syntax. W3C Note, www.w3.org/TR/owl-xmlsyntax/ [Last accessed, 23/11/2012].
- Horridge, M., Drummond, N., Goodwin, J., Rector, A., Stevens, R., and Wang, H. H. (2006). The manchester owl syntax. In Grau, B. C., Hitzler, P., Shankey, C., and Wallace, E., editors, *Proceedings of OWL: Experiences and Directions (OWLED'06)*, Athens, Georgia, USA,.
- Horrocks, I. (2005). Ist project 2001-33052 wonderweb: Ontology infrastructure for the semantic web. d29: Final report. Technical report. wonderweb.semanticweb.org/deliverables/documents/D29.pdf [Last accessed, 14/07/2012]
- Horrocks, I., Patel-Schneider, P., Boley, H., Tabet, S., Grosz, B., Dean, M., et al. (2004). SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member submission*, <http://www.w3.org/Submission/SWRL/> [Last accessed, 23/12/2012]
- Horrocks, I., Patel-Schneider, P. F., and van Harmelen, F. (2003). From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics*, 1(1):7-26
-

- Hoser, B., Hotho, A., Jäschke, R., Schmitz, C., and Stumme, G. (2006). Semantic Network Analysis of Ontologies. In Sure, Y. and Domingue, J., editors, *The Semantic Web: Research and Applications*, volume 4011 of *Lecture Notes in Computer Science*, Springer, Berlin/Heidelberg, pages 514–529. .
- Iqbal, A., Ureche, O., Hausenblas, M., and Tummarello, G. (2009). Ld2sd: Linked data driven software development. In *Software Engineering and Knowledge Engineering (SEKE 09)*, Boston, USA, pages 240–245.
- Isele, R., Umbrich, J., Bizer, C., and Harth, A. (2010). LDspider: An Open-source Crawling Framework for the Web of Linked Data. In *Proceedings of the International Semantic Web Conference (ISWC) Posters & Demonstrations Track*, volume 658 of *CEUR Workshop Proceedings*, pages 29–32, Shanghai, China.
- Jain, P., Hitzler, P., Yeh, P., Verma, K., and Sheth, A. (2010). Linked Data is Merely More Data. In *Proceedings of the AAAI Spring Symposium, Linked AI: Linked Data Meets Artificial Intelligence*, pages 82–86, Menlo Park, CA, USA. AAAI.
- Jarrar, M. and Meersman, R. (2002). Formal ontology engineering in the dogma approach. In Meersman, R. and Tari, Z., editors, *On-The-Move Conferences; CoopIS/DOA/ODBASE*, volume 2519 of *Lecture Notes in Computer Science*, pages 1238–1254.
- Jones, D., Bench-Capon, T., and Visser, P. (1998). Methodologies for ontology development. In *Proceedings of IT&KNOWS Conference of the 15 th IFIP World Computer Congress*, pages 62–75. Chapman and Hall Ltd.
- Klyne, G. and Carroll, J. J. (2004). Resource description framework (rdf): Concepts and abstract syntax. World Wide Web Consortium, Recommendation REC-rdf-concepts-20040210, www.w3.org/TR/rdf-concepts/ [Last accessed, 14/9/2012]
- Knight, K., Chander, I., Haines, M., Hatzivassiloglou, V., Hovy, E. H., Iida, M., Luk, S. K., Whitney, R., and Yamada, K. (1995). Filling knowledge gaps in a broad-coverage machine translation system., *Proceedings of the 14th international joint conference on Artificial intelligence*, Montreal, Quebec, Canada, volume 2: pages 1390–1396.
- Knight, K. and Luk, S. K. (1994). Building a large-scale knowledge base for machine translation. In *National Conference on Artificial Intelligence, AAAI*, pages 773–778.
- Knoke, D., Yang, S., and Kuklinski, J. (2008). *Social network analysis*, volume 2. Sage Publications Los Angeles, CA.
-

- Knublauch, H., Fergerson, R., Noy, N., and Musen, M. (2004). The protégé owl plugin: An open development environment for semantic web applications. *The Semantic Web-ISWC 2004*, pages 229–243.
- Kobilarov, G., Bizer, C., Auer, S., and Lehmann, J. (2009a). Dbpedia - a linked data hub and data source for web applications and enterprises. In *Proceedings of Developers Track of 18th International World Wide Web Conference (WWW 2009), April 20th-24th, Madrid, Spain*.
- Kobilarov, G., Scott, T., Raimond, Y., Oliver, S., Sizemore, C., Smethurst, M., Bizer, C., and Lee, R. (2009b). Media meets Semantic Web—how the BBC uses DBpedia and Linked Data to make connections. *The Semantic Web: Research and Applications*, pages 723–737.
- Kotis, K. (2008). On supporting hcome-3o ontology argumentation using semantic wiki technology. In Kotis, K., editor, *OnTheMove (OTM) 2008 Conference, Community-Based Evolution of Knowledge-Intensive Systems (COMBEK'08) Workshop*, LNCS 5333, Springer-Verlag Berlin Heidelberg. , pages 193–199
- Kotis, K. and Vouros, G. (2006). Human-centered ontology engineering: the hcome methodology. *International Journal of Knowledge and Information Systems (KAIS)*, 10:109–131.
- Kotis, K., Vouros, G., and Alonso, J. P. (2004). Hcome: tool-supported methodology for collaboratively devising living ontologies. In Kotis, K., editor, *Semantic Web and Databases, Workshop in VLDB'04 Conference*, Toronto ,Canada.
- Lattanzi, S. and Sivakumar, D. (2009). Affiliation networks. In Mitzenmacher, M., editor, *Proceedings of the 41st annual ACM symposium on Theory of computing*, STOC 2009, NY, USA., pages 427–434.
- Lenat, D. B., Guha, R. V., Pittman, K., Pratt, D., and Shepherd, M. (1990). Cyc: Toward programs with common sense. *Communication of ACM*, 33(8):30–49.
- Leung, N. K. Y., Lau, S. K., Fan, J., and Tsang, N. (2011). An integration-oriented ontology development methodology to reuse existing ontologies in an ontology development process. In *Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services*, iiWAS '11, NY, USA., pages 174–181
- Lewis, R. (2007). Dereferencing http uris. Draft Tag Finding, <http://www.w3.org/2001/tag/doc/httpRange-14/2007-05-31/HttpRange-14.html>. [Last accessed, 25/7/2012]
-

- López, M. F., Gómez-Pérez, A., Sierra, J. P., and Sierra, A. P. (1999). Building a chemical ontology using Methontology and the OntologyDesign Environment. *IEEE Intelligent Systems and Their Applications*, 14(1):37–46.
- Lozano-Tello, A. and Gomez-Perez, A. (2004). ONTOMETRIC: A Method to Choose the Appropriate Ontology. *Journal of Database Management*, 15(2):1–18.
- Madnick, S. E. and Lupu, M. (2008). Implementing the content interchange (coin) approach through use of semantic web tools. *Semantic Web, Ontologies and Databases*, Lecture Notes in Computer Science Volume 5005, 2008, pp 77-97
- Madnick, S. E., Wernerfelt, B., and Firat, A. (2003). Information integration using contextual knowledge and ontology merging. Technical report. www.mit.edu/bgrosop/paps/phd-thesis-aykut-firat.pdf [Last accessed, 23/12/2012]
- Maedche, A. and Staab, S. (2002). Measuring Similarity between Ontologies. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web (EKAW)*, pages 251–263, London, UK. Springer-Verlag.
- Maedche, A. and Zacharias, V. (2002). Clustering Ontology-Based Metadata in the Semantic Web. In *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD)*, volume 2431 of *Lecture Notes in Computer Science*, pages 383–408, Helsinki, Finland. Springer Berlin / Heidelberg.
- Manaf, N. A. A., Bechhofer, S., and Stevens, R. (2010). A Survey of Identifiers and Labels in OWL Ontologies. In Sirin, E. and Clark, K., editors, *OWLED*, volume 614 of *CEUR Workshop Proceedings*.
- Mariolis, P. (1975). Interlocking directorates and control of corporations: The theory of bank control. *Social Science Quarterly*, 56(3):425–439.
- Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIlraith, S., Narayanan, S., Paolucci, M., Parsia, B., Payne, T. R., Sirin, E., Srinivasan, N., and Sycara, K. (2004). OWL-S: Semantic markup for web services. W3C Member submission 22 : 2007-04.
- Martínez-Prieto, M., Arias Gallego, M., and Fernández, J. (2012). Exchange and Consumption of Huge RDF Data. In Simperl, E., Cimiano, P., Polleres, A., Corcho, O., and Presutti, V., editors, *Lecture Notes in Computer Science, The Semantic Web: Research and Applications*, volume 7295, Springer Berlin Heidelberg., pages 437–452
-

- Masolo, C., Borgo, S., Gangemi, A., Guarino, N., and Oltramari, A. (2003). Wonderweb deliverable d18 ontology library (final). Technical report, IST Project 2001-33052 trac.assembla.com/soray/export/97/user/Henry/.../D18.pdf [Last accessed, 17/9/2012]
- McBride, B. (2002). Jena: A Semantic Web Toolkit. *IEEE Internet Computing*, 6(6):55–59.
- McGuinness, D. L. and van Harmelen, F. (2004). OWL Web Ontology Language Overview. Technical report, W3C - World Wide Web Consortium. <http://www.w3.org/TR/owl-features/> (Last Accessed 15 July 2012).
- McIlraith, S., Son, T., and Zeng, H. (2001). Semantic web services. *Intelligent Systems, IEEE*, 16(2):46 – 53.
- Medicale, A. (1995). Issues in the structuring and acquisition of an ontology for medical language understanding. *Meth Inform Med*, 34:15–24.
- Meditkos, G. and Bassiliades, N. (2010). Dlejena: A practical forward-chaining owl 2 r reasoner combining jena and pellet. *Web Semantics*, 8(1):89–94.
- Mika, P. (2005). Ontologies Are Us: A Unified Model of Social Networks and Semantics. In *Proceedings of 4th International Semantic Web Conference (ISWC)*, volume 3729 of *Lecture Notes in Computer Science*, pages 522–536, Galway, Ireland. Springer Berlin / Heidelberg.
- Mika, P., Meij, E., and Zaragoza, H. (2009). Investigating the semantic gap through query log analysis. In Bernstein, A., Karger, D. R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., and Thirunarayan, K., editors, *International Semantic Web Conference*, volume 5823 of *Lecture Notes in Computer Science*, pages 441–455. Springer.
- Mizoguchi, R. and Ikeda, M. (1996). Towards ontology engineering. Technical Report AI-TR-96-1, The Institute of Scientific and Industrial Research, Osaka University.
- Möller, K. (2012). Lifecycle Models of Data-centric Systems and Domains. *Semantic Web journal*. http://semantic-web-journal.org/sites/default/files/swj125_0.pdf [Last accessed, 15/12/2012]
- Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., and Swartout, W. (1991). Enabling technology for knowledge sharing. *AI Magazine*, 12(3):36–56.
- Newman, M. (2008). The mathematics of networks. *The New Palgrave Encyclopedia of Economics*, 2.
-

- Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45(2):167–256.
- Newman, M. E. J. (2004). Coauthorship networks and patterns of scientific collaboration. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5200–5205.
- Newman, M. E. J. (2010). *Networks: An Introduction*. Oxford University Press.
- Newman, M. E. J., Strogatz, S. H., and Watts, D. J. (2001). Random graphs with arbitrary degree distributions and their applications. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 64(2):026118.
- Nikolov, A., Uren, V., and Motta, E. (2010). Data linking: capturing and utilising implicit schema-level relations. In *Proceedings of the Linked Data on the Web (LDOW 2010) at 19th International World Wide Web Conference (WWW 2010)*, Raleigh, USA. CEUR Workshop Proceedings Vol.628. pages 1–11,
- Noy, N. F. and Klein, M. (2004). Ontology evolution: Not the same as schema evolution. *Knowledge and Information Systems*, 6(4):428–440.
- Noy, N. F. and McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology. Technical report, *Stanford Knowledge Systems Laboratory and Stanford Medical Informatics*.
- Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A. D., Chute, C. G., and Musen, M. A. (2009). BioPortal: Ontologies and Integrated Data Resources at the Click of a Mouse. *Nucleic Acids Research*, 37(Web-Server-Issue):170–173.
- Oberle, D., Staab, S., and Volz, R. (2005). Three dimensions of knowledge representation in WonderWeb. *KI*, 19(1):31-35.
- Obrst, L., Ceusters, W., Mani, I., Ray, S., and Smith, B. (2007). The evaluation of ontologies. In Baker, C. J. O. and Cheung, K.-H., editors, *Semantic Web*, Springer US. , pages 139–158.
- Okamoto, K., Chen, W., and Li, X.-Y. (2008). Ranking of closeness centrality for large-scale social networks. In Preparata, F., Wu, X., and Yin, J., editors, *Frontiers in Algorithmics*, volume 5059 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg. pages 186–195.
- Oliveira, M. and Gama, J. a. (2012). An overview of social network analysis. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(2):99–115.
-

- Oltramari, A., Gangemi, A., Guarino, N., and Masolo, C. (2002). Restructuring wordnet's top-level: The OntoClean approach. *LREC2002, Las Palmas, Spain*, 68.
- OpenLink Software (2009). Virtuoso Open-Source Edition., virtuoso.openlinksw.com/dataspace/dav/wiki/Main/ [Last accessed, 17/12/2012]
- O'Reilly, T. (2005). What is web 2.0: Design patterns and business models for the next generation of software. *Communications strategies 1 (2007)*: 17
- Oren, E., Möller, K., Scerri, S., Handschuh, S., and Sintek, M. (2006). What are semantic annotations? Technical report, DERI Galway.
- Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab.
- Palma, R., Hartmann, J., and Haase, P. (2008). Omv-ontology metadata vocabulary for the semantic web. Technical report, Technical report, Universidad Politecnica de Madrid, University of Karlsruhe, 2008. Version 2.4. Available at <http://omv.ontoware.org>.
- Pant, G. and Srinivasan, P. (2005). Learning to crawl: Comparing classification schemes. *ACM Transactions on Information Systems (TOIS)*, 23(4):430–462.
- Pant, G., Srinivasan, P., and Menczer, F. (2004). Crawling the web. *Web Dynamics*, 2004:153–178.
- Patel-Schneider, P. F., Hayes, P., and Horrocks, I. (2004). OWL:Web Ontology Language Semantics and Abstract Syntax. *W3C Recommendation.*, www.w3.org/TR/owl-semantics/ [Last accessed, 25/12/2012]
- Porzel, R. and Malaka, R. (2004). A task-based approach for ontology evaluation. In *Proceedings of the 16th European Conference on Artificial Intelligence, Valencia, Spain*.
- Presutti, V., Gangemi, A., David, S., de Cea, G., Surez-Figueroa, M., Montiel-Ponsoda, E., and Poveda, M. (2008). Neon deliverable d2. 5.1. a library of ontology design patterns: reusable solutions for collaborative design of networked ontologies. *NeOn Project*. <http://www.neon-project.org>.
- Radatz, J., Geraci, A., and Katki, F. (1990). IEEE standard glossary of Software Engineering terminology. *IEEE Standard*, standards.ieee.org/findstds/standard/610.12-1990.html [Last accessed, 18/12/2012]
-

- Raimond, Y., Abdallah, S. A., Sandler, M. B., and Giasson, F. (2007). The Music Ontology. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria. pages 417–422,
- Rector, A. L., Drummond, N., Horridge, M., Rogers, J., Knublauch, H., Stevens, R., Wang, H., and Wroe, C. (2004). OWL Pizzas: Practical Experience of Teaching OWL-DL: Common Errors and Common Patterns. In *Proceedings of 14th International Conference on Knowledge Engineering and Knowledge Management (EKAW)*, volume 3257 of *Lecture Notes in Computer Science*, Whittlebury Hall, UK. Springer Berlin / Heidelberg. pages 63–81,
- Richardson, M., Prakash, A., and Brill, E. (2006). Beyond pagerank: machine learning for static ranking. In *Proceedings of the 15th international conference on World Wide Web*, pages 707–715.
- Ruttenberg, A., Clark, T., Bug, W. J., Samwald, M., Bodenreider, O., Chen, H., Doherty, D., Forsberg, K., Gao, Y., Kashyap, V., Kinoshita, J., Luciano, J., Marshall, M. S., Ogbuji, C., Rees, J., Stephens, S., Wong, G. T., Wu, E., Zaccagnini, D., Hongsermeier, T., Neumann, E., Herman, I., and Cheung, K.-H. (2007). Advancing translational research with the semantic web. *BMC Bioinformatics*, 8(S-3).
- Sabou, M., Lopez, V., Motta, E., and Uren, V. (2006). Ontology Selection: Ontology Evaluation on the Real Semantic Web. In *Proceedings of the Evaluation of Ontologies on the Web Workshop, held in conjunction with WWW'2006*, pages 1–8, Edinburgh, Scotland.
- Sandhaus, E. (2010). Abstract: Semantic technology at the new york times: Lessons learned and future directions. In Patel-Schneider, P. F., Pan, Y., Hitzler, P., Mika, P., 0007, L. Z., Pan, J. Z., Horrocks, I., and Glimm, B., editors, *International Semantic Web Conference (2)*, volume 6497 of *Lecture Notes in Computer Science*, 55. Springer.
- Sauermann, L. and Cyganiak, R. (2008). *Cool URIs for the Semantic Web*. W3C Interest Group Note, W3C. www.w3.org/TR/2007/WD-cooluris-20071217/ [Last accessed, 17/08/2012]
- Schreiber, A. T. and Terpstra, P. (1996). Sisyphus-vt: A CommonKADs solution. *Int. J. Hum.-Comput. Stud.*, 44(3-4):373–402.
- Schreiber, G., Wielinga, B., and Jansweijer, W. (1995). The KACTUS view on the 'o' word. In Skuce, D., editor, *The 1995 International Joint Conference on AI: Montreal, Quebec, Canada: 1995, August, 20-25*, Workshop on Basic Ontological Issues in Knowledge Sharing, pages 15(1):1-10.
-

- Schwaber, K. et al. (1995). Scrum development process. In *Proceedings of the Workshop on Business Object Design and Implementation at the 10th Annual Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA'95)*.
- Shadbolt, N., Hall, W., and Berners-Lee, T. (2006). The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3):96–101.
- Simmons, E. (2005). The usage model: a structure for richly describing product usage during design and development. In *Proceedings of 13th IEEE International Conference on Requirements Engineering*, pages 403 – 407, Paris, France. IEEE.
- Simperl, E. (2009). Reusing ontologies on the Semantic Web: A feasibility study. *Data and Knowledge Engineering*, 68(10):905–925.
- Sirin, E. and Parsia, B. (2004). Pellet: An OWL DL Reasoner. *Web Semantics: science, services and agents on the World Wide Web 5.2 (2007)*: 51-53.
- Sleeman, J. and Finin, T. (2010). Learning Co-reference Relations for FOAF Instances. In *Proceedings of the International Semantic Web Conference (ISWC) Posters & Demonstrations Track*, volume 658 of *CEUR Workshop Proceedings*, Shanghai, China.
- Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L., Eilbeck, K., Ireland, A., Mungall, C., et al. (2007). The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251–1255.
- Staab, S. and Studer, R., editors (2009). *Handbook on Ontologies*. Springer, Berlin, 2. edition.
- Staab, S., Studer, R., Schnurr, H.-P., and Sure, Y. (2001). Knowledge processes and ontologies. *IEEE Intelligent Systems*, 16(1):26–34.
- Steiner, R. T. T. and Hausenblas, M. (2010). How Google is using Linked Data Today and Vision For Tomorrow. In *Proceedings of the Workshop on Linked Data in the Future Internet at the Future Internet Assembly*, volume 700 of *CEUR Workshop Proceedings*, Ghent, Belgium.
- Steve, G., Gangemi, A., and Pisanelli, D. M. (1997). Integrating medical terminologies with onions methodology. In *Information Modelling and Knowledge Bases VIII*. IOS Press.
- Stojanovic, L. (2004). *Methods and tools for ontology evolution*. PhD thesis, Karlsruhe Institute of Technology. <http://d-nb.info/1001606787>.
-

- Stojanovic, L., Maedche, A., Motik, B., and Stojanovic, N. (2002). User-Driven Ontology Evolution Management. In *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, volume 2473 of *Lecture Notes in Computer Science*, Siguenza, Spain. Springer Berlin / Heidelberg, pages 133–140,
- Stuckenschmidt, H. (2003). Modularization of ontologies. WonderWeb Deliverable D21, available at: <http://wonderweb.semanticweb.org/deliverables/D21.shtml>. [Last accessed, 25/8/2012]
- Studer, R., Benjamins, V. R., and Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1-2):161–197.
- Suárez-Figueroa, M. C., Gómez-Pérez, A., and Fernández-López, M. (2012). The NeOn Methodology for Ontology Engineering. In Suárez-Figueroa, M. C., Gómez-Pérez, A., Motta, E., and Gangemi, A., editors, *Ontology Engineering in a Networked World*, Springer Berlin Heidelberg, pages 9–34.
- Sure, Y. (2002). On-to-knowledge: Ontology based knowledge management tools and their application. *Kunstliche Intelligenz*, pages 35–37.
- Sure, Y., Akkermans, H., Broekstra, J., Davies, J., Ding, Y., Duke, A., Engels, R., Fensel, D., Horrocks, I., Iosif, V., (2003). On-to-knowledge: Semantic web enabled knowledge management. *Web Intelligence*, 35:277–300.
- Sure, Y., Erdmann, M., Angele, J., Staab, S., Studer, R., and Wenke, D. (2002a). Ontoedit: Collaborative ontology development for the semantic web. In Horrocks, I. and Hendler, J. A., editors, *International Semantic Web Conference*, volume 2342 of *Lecture Notes in Computer Science*, Springer, pages 221–235.
- Sure, Y., Erdmann, M., Angele, J., Staab, S., Studer, R., and Wenke, D. (2002b). OntoEdit: Collaborative Ontology Development for the Semantic Web. In Horrocks, I. and Hendler, J. A., editors, *International Semantic Web Conference*, volume 2342 of *Lecture Notes in Computer Science*, Springer, pages 221–235.
- Sure, Y., Staab, S., and Studer, R. (2004). On-To-Knowledge Methodology (OTKM). In Staab, S. and Studer, R., editors, *Handbook on Ontologies: International Handbook on Information Systems*, Springer, pages 117–132.
- Swartout, B., Patil, R., Knight, K., and Russ, T. (1997). Toward distributed use of large-scale ontologies. In *Ontological Engineering, AAAI-97 Spring Symposium Series*, pages 138–148.
-

- T. B. Lee, J. H. and Lassila., O. (2001). The Semantic Web. T. B. Lee, J. Hendler, and O. Lassila.. In *Scientific America*, May 2001. 284.5 (2001): 28-37.
- Tao, C., Embley, D. W., and Liddle, S. W. (2009a). FOCIH: Form-based ontology creation and information harvesting. In Laender, A. H. F., Castano, S., Dayal, U., Casati, F., and de Oliveira, J. P. M., editors, *Conceptual Modeling-*, volume 5829 of *Lecture Notes in Computer Science*, Springer, pages 346–359.
- Tao, J., Ding, L., and McGuinness, D. L. (2009b). Instance Data Evaluation for Semantic Web-Based Knowledge Management Systems. In *Proceedings of the 42nd Hawaii International Conference on Systems Science (HICSS)*, , Big Island, HI, USA. IEEE Computer Society, pages 1–10
- Tartir, S. and Arpinar, I. B. (2007). Ontology evaluation and ranking using ontoqa. *International Conference on Semantic Computing*, pages185–192.
- Tartir, S., Arpinar, I. B., Moore, M., Sheth, A. P., and Aleman-Meza, B. (2005). OntoQA: Metric-Based Ontology Quality Analysis. In *Proceedings of IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources*, volume 9, Houston, Texas. IEEE Computer Society Press, pages 45–53,
- ter Horst, H. J. (2005). 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(2-3):79 – 115.
- Thelwall, M. and Stuart, D. (2006). Web crawling ethics revisited: Cost, privacy, and denial of service. *Journal of the American Society for Information Science and Technology*, 57(13):1771–1779.
- Theoharis, Y., Tzitzikas, Y., Kotzinos, D., and Christophides, V. (2008). On Graph Features of Semantic Web Schemas. *IEEE Transactions on Knowledge and Data Engineering* , 20(5):692 –702.
- Tokosumi, A., Matsumoto, N., Tomioka, M., and Voss, K. (2006). Academic knowledge ontologies and a systems solution. In *Proceedings of the 5th International Conference of the Cognitive Science*, pages 211–212.
- Tong, A., Drees, B., Nardelli, G., Bader, G., Brannetti, B., Castagnoli, L., Evangelista, M., Ferracuti, S., Nelson, B., Paoluzi, S., et al. (2002). A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules. *Science Signalling*, 295(5553):321.
-

- TopQuadrant (2011). TopQuadrant | Products | TopBraid Composer. www.topquadrant.com/products/TB_Composer.html [Last accessed, 12/5/2012]
- Tran, T., Haase, P., Lewen, H., Óscar Muñoz García, Gómez-Pérez, A., and Studer, R. (2008). Lifecycle-support in architectures for ontology-based information systems. pages 508–522.
- Tsarkov, D. and Horrocks, I. (2006). Fact++ description logic reasoner: System description. *Automated Reasoning*, pages 292–297.
- Tummarello, G., Cyganiak, R., Catasta, M., Danielczyk, S., Delbru, R., and Decker, S. (2010). Sig.ma: Live views on the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4):355–364.
- Tummarello, G., Oren, E., and Delbru, R. (2007). Sindice.com: Weaving the Open Linked Data. In *Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference (ISWC/ASWC), Busan, South Korea*, volume 4825 of *LNCS*, , Berlin, Heidelberg. Springer Verlag, pages 547–560
- Tutoky, G. (2011). Collaboration Social Networksâ Information Sciences and Technologies. In *Bulletin of the ACM Slovakia - ISSN 1338-1237*. ACM.
- Uschold, M. (1996). Building ontologies: Towards a unified methodology. In *Proceedings of Expert Systems '96, the 16th Annual Conference of the British Computer Society Specialist Group on Expert Systems*, Cambridge, UK.
- Uschold, M., Bateman, J., Davis, M., and Sowa, J. (2011). Ontology summit 2011 communique : Making the case for ontology. http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2011_Communique [Last accessed, 18/7/2012]
- Uschold, M. and Gruninger, M. (1996). Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, 11(02):93–136.
- Uschold, M. and Jasper, R. (1999). A framework for understanding and classifying ontology applications. In *Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods (KRR5): Stockholm, Sweden: 1999*.
- Uschold, M. and King, M. (1995). Towards a methodology for building ontologies. In *Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*, Montreal, Canada.
-

- Valente, A., Russ, T., MacGregor, R., and Swartout, W. (1999). Building and (re)using an ontology of air campaign planning. *Intelligent Systems and their Applications, IEEE*, 14(1):27–36.
- Van Damme, C., Hepp, M., and Siorpaes, K. (2007). Folksontology: An integrated approach for turning folksonomies into ontologies. In *Bridging the Gap between Semantic Web and Web 2.0.*, 2, 57-70.
- Vega, J. C. A., Corcho, O., Fernández-López, M., and Gómez-Pérez, A. (2001). WebODE: A Scalable workbench for Ontological Engineering. In Proceedings of the 1st international conference on Knowledge capture papers. 6-13.
- Völker, J., Vrandečić, D., and Sure, Y. (2005). Automatic Evaluation of Ontologies (AEON). In Gil, Y., Motta, E., Benjamins, V. R., and Musen, M. A., editors, *Proceedings of the 4th International Semantic Web Conference (ISWC'05)*, volume 3729 of *LNCS*, Galway, Ireland. Springer. pages 716–731
- Volz, R. (2001). ONTOSERVER –Infrastructure for the Semantic Web. SWWS01 (2001): 96.
- Volz, R., Oberle, D., Motik, B., Staab, S., and Studer, R. (2002). Kaon server architecture. Technical Report D5, WonderWeb project deliverable. Also published as AIFB technical report.
- Vossen, G. and Hagemann, S. (2007). *Unleashing Web 2.0: From Concepts to Creativity*. Morgan Kaufmann.
- Vrandečić, D. and Gangemi, A. (2006). Unit Tests for Ontologies. In Meersman, R., Tari, Z., and Herrero, P., editors, *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, volume 4278 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, , pages 1012–1020.
- Vrandečić, D., Pinto, S., Tempich, C., and Sure, Y. (2005). The DILIGENT knowledge processes. *Journal of Knowledge Management*, 9(5):85–96.
- Vrandečić, D. and Sure, Y. (2007). How to design better ontology metrics. *The Semantic Web: Research and Applications*, pages 311–325.
- Vrandečić, D. (2010). *Ontology Evaluation*. PhD thesis, Karlsruhe Institute of Technology. Available Online at: <http://www.aifb.kit.edu/images/b/b5/\OntologyEvaluation.pdf>. [Last accessed, 30/1/2013]
- Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1st Edition.
-

-
- Weisstein, E. (2005). Normal distribution. From MathWorld-A Wolfram Web Resource. <http://mathworld.wolfram.com/NormalDistribution.html>. [Last accessed, 7/8/2012]
- Weisstein, E. and Polynomials, M. (2004). Mathworld-A wolfram web resource. From MathWorld,A Wolfram Web Resource. <http://mathworld.wolfram.com/BellNumber.html>.
- Wielinga, B., Akkermans, J. M., and Schreiber, A. T. (1995). A formal analysis of parametric design problem solving. In *Proceedings of the 9th Banff Knowledge Acquisition Workshop (KAW-95)*, pages 37–1.
- Zablith, F. (2009). Ontology Evolution: A Practical Approach. In *Proceedings of Workshop on Matching and Meaning at Artificial Intelligence and Simulation of Behaviour (AISB), Edinburgh, UK, 2009*, volume 1.
- Zablith, F. (2011). *Harvesting Online Ontologies for Ontology Evolution*. PhD thesis, The Knowledge Media Institute (KMi), The Open University, Milton Keynes, The United Kingdom, available from <http://fouad.zablith.org>.
- Zachary, W. (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473.
- Zhang, H. (2008). The scale-free nature of semantic web ontology. In *Proceeding of the 17th international conference on World Wide Web*, pages 1047–1048.
- Zhang, X., Li, H., and Qu, Y. (2006). Finding Important Vocabulary Within Ontology. In Mizoguchi, R., Shi, Z., and Giunchiglia, F., editors, *Asian Semantic Web Conference (ASWC)*, volume 4185 of *Lecture Notes in Computer Science*, Springer., pages 106–112.
- Zhang, Z., Huang, Z., and Zhang, X. (2010). Knowledge summarization for scalable semantic data processing. *Journal of Computational Information Systems*, 6(12):3893–3902.
- Zhou, D., Huang, J., and Schölkopf, B. (2006). Learning with hypergraphs: Clustering, classification, and embedding. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *NIPS*, MIT Press., pages 1601–1608.
- Zimmermann, A. (2010). Ontology Recommendation for the Data Publishers. In *Proceedings of 1st Workshop on Ontology Repositories*, volume 596 of *CEUR Workshop Proceedings*, Aachen, Germany.
-

-
- Zweigenbaum, P. (1994). Menelas: an access system for medical records using natural language. *Computer methods and programs in Biomedicine*, 45(1-2):117–120.
- Zweigenbaum, P., Jacquemart, P., Grabar, N., and Habert, B. (2001). Building a text corpus for representing the variety of medical language. *Studies in health technology and informatics*, (1):290–294.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear from any copyright owner who has been omitted or incorrectly acknowledged.

Appendix A - U Ontology Listing

For readability, the ontology listing for U Ontology only contains the definitions of its components (i.e. concepts, object properties, datatype properties, and axioms) without the instance data and in-line documentation.

```
1 <?xml version="1.0"?>
3
5 <!DOCTYPE rdf:RDF [
7   <!ENTITY uo "http://oua.uontology.org/v1#" >
9   <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
11  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
13  <!ENTITY skos "http://www.w3.org/2004/02/skos/core#" >
15  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
17  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
19 ]>
21
23 <rdf:RDF xmlns="http://oua.uontology.org/v1#"
25   xmlns:base="http://oua.uontology.org/v1"
27   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
29   xmlns:owl="http://www.w3.org/2002/07/owl#"
31   xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
33   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
35   xmlns:skos="http://www.w3.org/2004/02/skos/core#"
37   xmlns:uo="http://oua.uontology.org/v1#">
39   <owl:Ontology rdf:about="http://oua.uontology.org/v1#">
41     <owl:imports rdf:resource="http://omv.ontoware.org/2005/05/ontology"/>
43     <owl:Ontology>
45
47       <!--
49       //////////////////////////////////////
51       //
53       // Annotation properties
55       //
57       //////////////////////////////////////
59       -->
61
63       <owl:AnnotationProperty rdf:about="&skos;prefLabel">
65         <rdfs:subPropertyOf rdf:resource="&rdfs;label"/>
67       </owl:AnnotationProperty>
69
71
73       <!--
75       //////////////////////////////////////
77       //
79       // Datatypes
81       //
83       //////////////////////////////////////
85       -->
87
89
91
93
95
97
99
101
103
105
107
109
111
113
115
117
119
121
123
125
127
129
131
133
135
137
139
141
143
145
```

```

47 //////////////////////////////////////////////////////////////////////////////////////////////////////////////////
48 -->
49
50
51
52
53 <!--
54 //////////////////////////////////////////////////////////////////////////////////////////////////////////////////
55 //
56 // Object Properties
57 //
58 //////////////////////////////////////////////////////////////////////////////////////////////////////////////////
59 -->
60
61
62
63 <!-- http://oua.ontology.org/v1#analysesOntology -->
64
65 <owl:ObjectProperty rdf:about="&u;analysesOntology">
66   <rdfs:range rdf:resource="&u;OntologyUsage"/>
67   <rdfs:domain rdf:resource="&u;OntologyUsageAnalysis"/>
68 </owl:ObjectProperty>
69
70
71
72
73 <!-- http://oua.ontology.org/v1#analysisPerformedOn -->
74
75 <owl:ObjectProperty rdf:about="&u;analysisPerformedOn">
76   <rdfs:domain rdf:resource="&u;OntologyUsageAnalysis"/>
77   <rdfs:range rdf:resource="&u;Source"/>
78 </owl:ObjectProperty>
79
80
81
82
83 <!-- http://oua.ontology.org/v1#attributeValue -->
84
85 <owl:ObjectProperty rdf:about="&u;attributeValue">
86   <rdfs:domain rdf:resource="&u;AttributeUsage"/>
87   <rdfs:domain rdf:resource="&u;Attribute"/>
88   <rdfs:range rdf:resource="&u;AttributeValue"/>
89   <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
90 </owl:ObjectProperty>
91
92
93 <!-- http://oua.ontology.org/v1#hasAttribute -->
94
95 <owl:ObjectProperty rdf:about="&u;hasAttribute">
96   <rdfs:comment xml:lang="en">This property allows to represent relationships
97     present between concepts based on their usage on real world implementations
98   </rdfs:comment>
99   <rdfs:range rdf:resource="&u;Attribute"/>
100   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
101 </owl:ObjectProperty>
102
103
104
105
106
107 <!-- http://oua.ontology.org/v1#hasAttributeUsage -->
108
109 <owl:ObjectProperty rdf:about="&u;hasAttributeUsage">
110   <rdfs:range rdf:resource="&u;AttributeUsage"/>
111   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
112   <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
113 </owl:ObjectProperty>
114
115
116
117 <!-- http://oua.ontology.org/v1#hasConceptUsage -->

```

```
119 <owl:ObjectProperty rdf:about="&u;hasConceptUsage">
120   <rdfs:range rdf:resource="&u;ConceptUsage"/>
121   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
122   <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
123 </owl:ObjectProperty>
124
125
126
127 <!-- http://oua.ontology.org/v1#hasDomainLabel -->
128
129 <owl:ObjectProperty rdf:about="&u;hasDomainLabel">
130   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
131   <rdfs:range rdf:resource="&u;DomainLabel"/>
132 </owl:ObjectProperty>
133
134
135
136
137 <!-- http://oua.ontology.org/v1#hasFormalLabel -->
138
139 <owl:ObjectProperty rdf:about="&u;hasFormalLabel">
140   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
141   <rdfs:range rdf:resource="&u;FormalLabel"/>
142 </owl:ObjectProperty>
143
144
145
146
147 <!-- http://oua.ontology.org/v1#hasIncentiveDim -->
148
149 <owl:ObjectProperty rdf:about="&u;hasIncentiveDim">
150   <rdfs:range rdf:resource="&u;IncentiveDim"/>
151   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
152 </owl:ObjectProperty>
153
154
155
156
157 <!-- http://oua.ontology.org/v1#hasKnowledgePattern -->
158
159 <owl:ObjectProperty rdf:about="&u;hasKnowledgePattern">
160   <rdfs:range rdf:resource="&u;KnowledgePattern"/>
161   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
162 </owl:ObjectProperty>
163
164
165
166
167 <!-- http://oua.ontology.org/v1#hasLabelUsage -->
168
169 <owl:ObjectProperty rdf:about="&u;hasLabelUsage">
170   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
171   <rdfs:range rdf:resource="&u;LabelUsage"/>
172 </owl:ObjectProperty>
173
174
175
176
177 <!-- http://oua.ontology.org/v1#hasMeasures -->
178
179 <owl:ObjectProperty rdf:about="&u;hasMeasures">
180   <rdfs:range rdf:resource="&u;Measure"/>
181   <rdfs:domain rdf:resource="&u;OntologyUsageAnalysis"/>
182 </owl:ObjectProperty>
183
184
185
186
187 <!-- http://oua.ontology.org/v1#hasObjectInPath -->
188
189 <owl:ObjectProperty rdf:about="&u;hasObjectInPath">
190   <rdfs:range rdf:resource="&u;PathConcept"/>
191   <rdfs:domain rdf:resource="&u;PathStep"/>
192 </owl:ObjectProperty>
193
194
195
196
197 <!-- http://oua.ontology.org/v1#hasPath -->
```

```
191 <owl:ObjectProperty rdf:about="&u;hasPath">
193   <rdfs:domain rdf:resource="&u;KnowledgePattern"/>
195   <rdfs:range rdf:resource="&u;Path"/>
197 </owl:ObjectProperty>
199 <!-- http://oua.ontology.org/v1#hasPathStep -->
201 <owl:ObjectProperty rdf:about="&u;hasPathStep">
203   <rdfs:domain rdf:resource="&u;Path"/>
205   <rdfs:range rdf:resource="&u;PathStep"/>
207 </owl:ObjectProperty>
209 <!-- http://oua.ontology.org/v1#hasPropertyInPath -->
211 <owl:ObjectProperty rdf:about="&u;hasPropertyInPath">
213   <rdfs:range rdf:resource="&u;PathProperty"/>
215   <rdfs:domain rdf:resource="&u;PathStep"/>
217 </owl:ObjectProperty>
219 <!-- http://oua.ontology.org/v1#hasRelation -->
221 <owl:ObjectProperty rdf:about="&u;hasRelation">
223   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
225   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
227   <rdfs:range rdf:resource="&u;Relationship"/>
229 </owl:ObjectProperty>
231 <!-- http://oua.ontology.org/v1#hasRelationshipUsage -->
233 <owl:ObjectProperty rdf:about="&u;hasRelationshipUsage">
235   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
237   <rdfs:range rdf:resource="&u;RelationshipUsage"/>
239   <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
241 </owl:ObjectProperty>
243 <!-- http://oua.ontology.org/v1#hasRichness -->
245 <owl:ObjectProperty rdf:about="&u;hasRichness">
247   <rdfs:range rdf:resource="&u;ConceptRichness"/>
249   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
251   <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
253 </owl:ObjectProperty>
255 <!-- http://oua.ontology.org/v1#hasRichnessDim -->
257 <owl:ObjectProperty rdf:about="&u;hasRichnessDim">
259   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
261   <rdfs:range rdf:resource="&u;RichnessDim"/>
</owl:ObjectProperty>
<!-- http://oua.ontology.org/v1#hasSubjectInPath -->
<owl:ObjectProperty rdf:about="&u;hasSubjectInPath">
  <rdfs:range rdf:resource="&u;PathConcept"/>
  <rdfs:domain rdf:resource="&u;PathStep"/>
</owl:ObjectProperty>
```



```
263
265 <!-- http://oua.ontology.org/v1#hasTerm -->
267 <owl:ObjectProperty rdf:about="&u;hasTerm">
269   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
271   <rdfs:range rdf:resource="&u;Term"/>
273 </owl:ObjectProperty>
275
277 <!-- http://oua.ontology.org/v1#hasUsageDim -->
279 <owl:ObjectProperty rdf:about="&u;hasUsageDim">
281   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
283   <rdfs:range rdf:resource="&u;UsageDim"/>
285 </owl:ObjectProperty>
287
289 <!-- http://oua.ontology.org/v1#hasVocab -->
291 <owl:ObjectProperty rdf:about="&u;hasVocab">
293   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
295   <rdfs:range rdf:resource="&u;Vocab"/>
297 </owl:ObjectProperty>
299
301 <!-- http://oua.ontology.org/v1#isComponentOf -->
303 <owl:ObjectProperty rdf:about="&u;isComponentOf">
305   <rdfs:domain rdf:resource="&u;Attribute"/>
307   <rdfs:domain rdf:resource="&u;DomainLabel"/>
309   <rdfs:domain rdf:resource="&u;FormalLabel"/>
311   <rdfs:domain rdf:resource="&u;Relationship"/>
313   <rdfs:range rdf:resource="&u;Vocab"/>
315 </owl:ObjectProperty>
317
319 <!-- http://oua.ontology.org/v1#isCoused -->
321 <owl:ObjectProperty rdf:about="&u;isCoused">
323   <rdf:type rdf:resource="&owl;ReflexiveProperty"/>
325   <rdfs:range rdf:resource="&u;OntologyUsage"/>
327   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
329 </owl:ObjectProperty>
331
333 <!-- http://oua.ontology.org/v1#isIncentivizedBy -->
335 <owl:ObjectProperty rdf:about="&u;isIncentivizedBy">
337   <rdfs:domain rdf:resource="&u;AttributeUsage"/>
339   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
341   <rdfs:domain rdf:resource="&u;RelationshipUsage"/>
343   <rdfs:range rdf:resource="&u;SearchEngine"/>
345   <rdfs:domain rdf:resource="&u;Term"/>
347 </owl:ObjectProperty>
349
351 <!-- http://oua.ontology.org/v1#isSupportedBy -->
353 <owl:ObjectProperty rdf:about="&u;isSupportedBy">
355   <rdfs:range rdf:resource="&u;SoftwareSupport"/>
357   <rdfs:domain rdf:resource="&u;Term"/>
359 </owl:ObjectProperty>
```

333

```
335 <!-- http://oua.ontology.org/v1#isUsedBy -->
337 <owl:ObjectProperty rdf:about="&u;isUsedBy">
339   <rdfs:range rdf:resource="&u;DataSource"/>
   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
341 </owl:ObjectProperty>
343
345 <!-- http://oua.ontology.org/v1#relationshipValue -->
347 <owl:ObjectProperty rdf:about="&u;relationshipValue">
   <rdfs:domain rdf:resource="&u;Relationship"/>
   <rdfs:domain rdf:resource="&u;RelationshipUsage"/>
349   <rdfs:range rdf:resource="&u;RelationshipValue"/>
   <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
351 </owl:ObjectProperty>
353
355 <!--
357 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
   //
   // Data properties
359 //
   ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
361 -->
363
365 <!-- http://oua.ontology.org/v1#URI -->
367 <owl:DatatypeProperty rdf:about="&u;URI">
369   <rdfs:domain rdf:resource="&u;AttributeUsage"/>
   <rdfs:domain rdf:resource="&u;Attribute"/>
371   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
   <rdfs:domain rdf:resource="&u;DataSource"/>
373   <rdfs:domain rdf:resource="&u;DomainLabel"/>
   <rdfs:domain rdf:resource="&u;FormalLabel"/>
375   <rdfs:domain rdf:resource="&u;LabelUsage"/>
   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
377   <rdfs:domain rdf:resource="&u;PathConcept"/>
   <rdfs:domain rdf:resource="&u;PathProperty"/>
379   <rdfs:domain rdf:resource="&u;Relationship"/>
   <rdfs:domain rdf:resource="&u;RelationshipUsage"/>
381   <rdfs:domain rdf:resource="&u;Term"/>
   <rdfs:domain rdf:resource="&u;Vocab"/>
383   <rdfs:range rdf:resource="&xsd:anyURI"/>
385 </owl:DatatypeProperty>
387
389 <!-- http://oua.ontology.org/v1#URL -->
391 <owl:DatatypeProperty rdf:about="&u;URL">
   <rdfs:domain rdf:resource="&u;DataSource"/>
   <rdfs:domain rdf:resource="&u;SearchEngine"/>
393   <rdfs:range rdf:resource="&xsd:anyURI"/>
395 </owl:DatatypeProperty>
397
399 <!-- http://oua.ontology.org/v1#analysisTimestamp -->
401 <owl:DatatypeProperty rdf:about="&u;analysisTimestamp">
   <rdfs:domain rdf:resource="&u;OntologyUsageAnalysis"/>
   <rdfs:range rdf:resource="&xsd:dateTimeStamp"/>
403 </owl:DatatypeProperty>
405
```

```

407 <!-- http://oua.ontology.org/v1#description -->
409 <owl:DatatypeProperty rdf:about="&u;description">
411   <rdfs:domain rdf:resource="&u;AttributeUsage"/>
411   <rdfs:domain rdf:resource="&u;Attribute"/>
413   <rdfs:domain rdf:resource="&u;AttributeValue"/>
413   <rdfs:domain rdf:resource="&u;ConceptRichness"/>
415   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
415   <rdfs:domain rdf:resource="&u;DataSource"/>
417   <rdfs:domain rdf:resource="&u;Dataset"/>
417   <rdfs:domain rdf:resource="&u;DomainLabel"/>
419   <rdfs:domain rdf:resource="&u;FormalLabel"/>
419   <rdfs:domain rdf:resource="&u;KnowledgePattern"/>
421   <rdfs:domain rdf:resource="&u;LabelUsage"/>
421   <rdfs:domain rdf:resource="&u;Measure"/>
423   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
423   <rdfs:domain rdf:resource="&u;OntologyUsageAnalysis"/>
425   <rdfs:domain rdf:resource="&u;Path"/>
425   <rdfs:domain rdf:resource="&u;PathConcept"/>
427   <rdfs:domain rdf:resource="&u;PathProperty"/>
427   <rdfs:domain rdf:resource="&u;PathStep"/>
429   <rdfs:domain rdf:resource="&u;Relationship"/>
429   <rdfs:domain rdf:resource="&u;RelationshipUsage"/>
431   <rdfs:domain rdf:resource="&u;RelationshipValue"/>
431   <rdfs:domain rdf:resource="&u;SearchEngine"/>
433   <rdfs:domain rdf:resource="&u;SoftwareSupport"/>
433   <rdfs:domain rdf:resource="&u;Source"/>
435   <rdfs:domain rdf:resource="&u;Streaming"/>
435   <rdfs:domain rdf:resource="&u;Term"/>
437   <rdfs:domain rdf:resource="&u;Vocab"/>
437   <rdfs:range rdf:resource="&rdfs;Literal"/>
439 </owl:DatatypeProperty>
441
443 <!-- http://oua.ontology.org/v1#docURI -->
445 <owl:DatatypeProperty rdf:about="&u;docURI">
445   <rdfs:domain rdf:resource="&u;OntologyUsageAnalysis"/>
447   <rdfs:range rdf:resource="&xsd:anyURI"/>
447 </owl:DatatypeProperty>
449
451 <!-- http://oua.ontology.org/v1#hasCousedValue -->
453 <owl:DatatypeProperty rdf:about="&u;hasCousedValue">
455   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
455   <rdfs:range rdf:resource="&xsd:int"/>
457 </owl:DatatypeProperty>
459
461 <!-- http://oua.ontology.org/v1#hasInstantiation -->
463 <owl:DatatypeProperty rdf:about="&u;hasInstantiation">
463   <rdfs:domain rdf:resource="&u;AttributeUsage"/>
465   <rdfs:domain rdf:resource="&u;ConceptUsage"/>
465   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
467   <rdfs:domain rdf:resource="&u;RelationshipUsage"/>
467   <rdfs:range rdf:resource="&xsd:int"/>
469 </owl:DatatypeProperty>
471
473 <!-- http://oua.ontology.org/v1#hasUsers -->
475 <owl:DatatypeProperty rdf:about="&u;hasUsers">
475   <rdfs:domain rdf:resource="&u;OntologyUsage"/>
477   <rdfs:range rdf:resource="&xsd:int"/>
477 </owl:DatatypeProperty>

```

```
479
481 <!-- http://oua.ontology.org/v1#incentiveValue -->
483 <owl:DatatypeProperty rdf:about="&uoincentiveValue">
485   <rdfs:domain rdf:resource="&uoincSearchEngine"/>
487   <rdfs:range rdf:resource="&xsdstring"/>
489 </owl:DatatypeProperty>
491
493 <!-- http://oua.ontology.org/v1#industry -->
495 <owl:DatatypeProperty rdf:about="&uoindustry">
497   <rdfs:domain rdf:resource="&uoincDataSource"/>
499   <rdfs:range rdf:resource="&rdfsLiteral"/>
501 </owl:DatatypeProperty>
503
505 <!-- http://oua.ontology.org/v1#name -->
507 <owl:DatatypeProperty rdf:about="&uoiname">
509   <rdfs:domain rdf:resource="&uoincAttributeUsage"/>
511   <rdfs:domain rdf:resource="&uoincAttribute"/>
513   <rdfs:domain rdf:resource="&uoincAttributeValue"/>
515   <rdfs:domain rdf:resource="&uoincConceptRichness"/>
517   <rdfs:domain rdf:resource="&uoincConceptUsage"/>
519   <rdfs:domain rdf:resource="&uoincDataSource"/>
521   <rdfs:domain rdf:resource="&uoincDataset"/>
523   <rdfs:domain rdf:resource="&uoincDomainLabel"/>
525   <rdfs:domain rdf:resource="&uoincFormalLabel"/>
527   <rdfs:domain rdf:resource="&uoincKnowledgePattern"/>
529   <rdfs:domain rdf:resource="&uoincLabelUsage"/>
531   <rdfs:domain rdf:resource="&uoincMeasure"/>
533   <rdfs:domain rdf:resource="&uoincOntologyUsage"/>
535   <rdfs:domain rdf:resource="&uoincOntologyUsageAnalysis"/>
537   <rdfs:domain rdf:resource="&uoincPath"/>
539   <rdfs:domain rdf:resource="&uoincPathConcept"/>
541   <rdfs:domain rdf:resource="&uoincPathProperty"/>
543   <rdfs:domain rdf:resource="&uoincPathStep"/>
545   <rdfs:domain rdf:resource="&uoincRelationship"/>
547   <rdfs:domain rdf:resource="&uoincRelationshipUsage"/>
549   <rdfs:domain rdf:resource="&uoincRelationshipValue"/>
551   <rdfs:domain rdf:resource="&uoincSearchEngine"/>
553   <rdfs:domain rdf:resource="&uoincSoftwareSupport"/>
555   <rdfs:domain rdf:resource="&uoincSource"/>
557   <rdfs:domain rdf:resource="&uoincStreaming"/>
559   <rdfs:domain rdf:resource="&uoincTerm"/>
561   <rdfs:domain rdf:resource="&uoincVocab"/>
563   <rdfs:range rdf:resource="&rdfsLiteral"/>
565 </owl:DatatypeProperty>
567
569 <!-- http://oua.ontology.org/v1#prefix -->
571 <owl:DatatypeProperty rdf:about="&uoincprefix">
573   <rdfs:domain rdf:resource="&uoincAttributeUsage"/>
575   <rdfs:domain rdf:resource="&uoincAttribute"/>
577   <rdfs:domain rdf:resource="&uoincConceptUsage"/>
579   <rdfs:domain rdf:resource="&uoincDomainLabel"/>
581   <rdfs:domain rdf:resource="&uoincFormalLabel"/>
583   <rdfs:domain rdf:resource="&uoincLabelUsage"/>
585   <rdfs:domain rdf:resource="&uoincOntologyUsage"/>
587   <rdfs:domain rdf:resource="&uoincRelationshipUsage"/>
589   <rdfs:domain rdf:resource="&uoincTerm"/>
591   <rdfs:domain rdf:resource="&uoincVocab"/>
593   <rdfs:range rdf:resource="&xsdstring"/>
595 </owl:DatatypeProperty>
```

```

551 <!-- http://oua.ontology.org/v1#searchEngineName -->
553
555 <owl:DatatypeProperty rdf:about="&u0;searchEngineName">
557   <rdfs:domain rdf:resource="&u0;SearchEngine"/>
559   <rdfs:range rdf:resource="&rdfs;Literal"/>
561 </owl:DatatypeProperty>
563
565 <!-- http://oua.ontology.org/v1#usageValue -->
567
569 <owl:DatatypeProperty rdf:about="&u0;usageValue">
571   <rdfs:domain rdf:resource="&u0;AttributeUsage"/>
573   <rdfs:domain rdf:resource="&u0;ConceptUsage"/>
575   <rdfs:domain rdf:resource="&u0;RelationshipUsage"/>
577   <rdfs:range rdf:resource="&xsd;int"/>
579 </owl:DatatypeProperty>
581
583 <!-- ////////////////////////////////////////////////////
585 //
587 // Classes
589 //
591 ////////////////////////////////////////////////////
593 -->
595
597 <!-- http://oua.ontology.org/v1#AttributeUsage -->
599
601 <owl:Class rdf:about="&u0;AttributeUsage">
603   <rdfs:subClassOf rdf:resource="&u0;Term"/>
605 </owl:Class>
607
609 <!-- http://oua.ontology.org/v1#Attribute -->
611
613 <owl:Class rdf:about="&u0;Attribute"/>
615
617 <!-- http://oua.ontology.org/v1#AttributeValue -->
619
621 <owl:Class rdf:about="&u0;AttributeValue"/>
623
625 <!-- http://oua.ontology.org/v1#ConceptRichness -->
627
629 <owl:Class rdf:about="&u0;ConceptRichness"/>
631
633 <!-- http://oua.ontology.org/v1#ConceptUsage -->
635
637 <owl:Class rdf:about="&u0;ConceptUsage">
639   <rdfs:subClassOf rdf:resource="&u0;Term"/>
641 </owl:Class>
643
645 <!-- http://oua.ontology.org/v1#DataSource -->
647
649 <owl:Class rdf:about="&u0;DataSource"/>
651
653 </pre>

```

```
623 <!-- http://oua.ontology.org/v1#Dataset -->
625 <owl:Class rdf:about="&u;Dataset">
627 <rdfs:subClassOf rdf:resource="&u;Source"/>
629 </owl:Class>
631 <!-- http://oua.ontology.org/v1#DomainLabel -->
633 <owl:Class rdf:about="&u;DomainLabel">
635 <rdfs:subClassOf rdf:resource="&u;LabelUsage"/>
637 </owl:Class>
639 <!-- http://oua.ontology.org/v1#FormalLabel -->
641 <owl:Class rdf:about="&u;FormalLabel">
643 <rdfs:subClassOf rdf:resource="&u;LabelUsage"/>
645 </owl:Class>
647 <!-- http://oua.ontology.org/v1#IncentiveDim -->
649 <owl:Class rdf:about="&u;IncentiveDim">
651 <rdfs:subClassOf rdf:resource="&u;Measure"/>
653 </owl:Class>
655 <!-- http://oua.ontology.org/v1#KnowledgePattern -->
657 <owl:Class rdf:about="&u;KnowledgePattern"/>
659 </owl:Class>
661 <!-- http://oua.ontology.org/v1#LabelUsage -->
663 <owl:Class rdf:about="&u;LabelUsage"/>
665 </owl:Class>
667 <!-- http://oua.ontology.org/v1#Measure -->
669 <owl:Class rdf:about="&u;Measure"/>
671 </owl:Class>
673 <!-- http://oua.ontology.org/v1#OntologyUsage -->
675 <owl:Class rdf:about="&u;OntologyUsage">
677 <rdfs:isDefinedBy rdf:datatype="&xsd:string">Jamshaid Ashraf</rdfs:isDefinedBy>
679 <skos:prefLabel rdf:datatype="&xsd:string">OntologyUsage</skos:prefLabel>
681 <rdfs:comment rdf:datatype="&xsd:string">This Concept is the core
683 concept of U Ontology. This represents the ontology which is analysed
685 by the OUSAF framework </rdfs:comment>
687 </owl:Class>
689 <!-- http://oua.ontology.org/v1#OntologyUsageAnalysis -->
691 <owl:Class rdf:about="&u;OntologyUsageAnalysis"/>
693 </owl:Class>
695 <!-- http://oua.ontology.org/v1#Path -->
697 <owl:Class rdf:about="&u;Path"/>
699 </owl:Class>
```

```
695
697 <!-- http://oua.ontology.org/v1#PathConcept -->
699 <owl:Class rdf:about="&u0;PathConcept"/>
701
703 <!-- http://oua.ontology.org/v1#PathProperty -->
705 <owl:Class rdf:about="&u0;PathProperty"/>
707
709 <!-- http://oua.ontology.org/v1#PathStep -->
711 <owl:Class rdf:about="&u0;PathStep"/>
713
715 <!-- http://oua.ontology.org/v1#Relationship -->
717 <owl:Class rdf:about="&u0;Relationship"/>
719
721 <!-- http://oua.ontology.org/v1#RelationshipUsage -->
723 <owl:Class rdf:about="&u0;RelationshipUsage">
725   <rdfs:subClassOf rdf:resource="&u0;Term"/>
727 </owl:Class>
729
731 <!-- http://oua.ontology.org/v1#RelationshipValue -->
733 <owl:Class rdf:about="&u0;RelationshipValue"/>
735
737 <!-- http://oua.ontology.org/v1#RichnessDim -->
739 <owl:Class rdf:about="&u0;RichnessDim">
741   <rdfs:subClassOf rdf:resource="&u0;Measure"/>
743 </owl:Class>
745
747 <!-- http://oua.ontology.org/v1#SearchEngine -->
749 <owl:Class rdf:about="&u0;SearchEngine"/>
751
753 <!-- http://oua.ontology.org/v1#SoftwareSupport -->
755 <owl:Class rdf:about="&u0;SoftwareSupport"/>
757
759 <!-- http://oua.ontology.org/v1#Source -->
761 <owl:Class rdf:about="&u0;Source"/>
763
765 <!-- http://oua.ontology.org/v1#Streaming -->
767 <owl:Class rdf:about="&u0;Streaming">
769   <rdfs:subClassOf rdf:resource="&u0;Source"/>
771 </owl:Class>
```

```
767
769 <!-- http://oua.ontology.org/v1#Term -->
771 <owl:Class rdf:about="&u;Term"/>
773
775 <!-- http://oua.ontology.org/v1#UsageDim -->
777 <owl:Class rdf:about="&u;UsageDim">
779   <rdfs:subClassOf rdf:resource="&u;Measure"/>
781 </owl:Class>
783
785 <!-- http://oua.ontology.org/v1#Vocab -->
787 <owl:Class rdf:about="&u;Vocab"/>
789 </rdf:RDF>
<!-- Generated by the OWL API (version 3.3.1957) http://owlapi.sourceforge.net -->
```


**Appendix B - Best PhD Symposium
Paper Award**



Figure B.1: Scanned copy of Best PhD Symposium Paper Award

Appendix C - Selected Publications