

Ontology-based Information Standards Development

A thesis submitted for the degree of Doctor of Philosophy

By

Bahareh Rahmanzadeh Heravi

**Department of Information Systems and
Computing**

Brunel University

January 2012

Abstract

Standards may be argued to be important enablers for achieving interoperability as they aim to provide unambiguous specifications for error-free exchange of documents and information. By implication, therefore, it is important to model and represent the concept of a standard in a clear, precise and unambiguous way. Although standards development organisations usually provide guidelines for the process of developing and approving standards, they are usually more concerned with administrative aspect of the process. As a consequence, the state-of-the-art lacks practical support for developing the structure and content of a standard specification. In short, there is no systematic development method currently available: (a) For developing the conceptual model underpinning a standard; and/or (b) to guide a group of stakeholders to develop a standard specification.

Semantic interoperability is considered to be an essential factor for effective interoperation – the ability to achieve semantic interoperability effectively and efficiently being strongly equated with quality by some. Semantics require that the meaning of terms, their relationships and also the restrictions and rules in the standards should be clearly defined in the early stages of standard development and act as a basis for the latter stages. This research proposes that ontology can help standards developers and stakeholders to address the issues of improving conceptual models and providing a robust and shared understanding of the domain. This thesis presents OntoStanD, a comprehensive ontology-based standards development methodology, which utilises the best practices of the existing ontology creation methods.

The potential value of OntoStanD is in providing a comprehensive, clear and unambiguous method for developing robust information standards, which are more test friendly and of higher quality. OntoStanD also facilitates standards conformance testing and change management, impacts interoperability and also assists in improved communication among the standards development team. Last, OntoStanD provides an approach that is repeatable, teachable and potentially general enough for creating any kinds of information standard.

TABLE OF CONTENTS

ABSTRACT	II
TABLE OF CONTENTS	III
LIST OF TABLES	VI
LIST OF FIGURES	VII
ABBREVIATIONS	X
ACKNOWLEDGEMENTS	XII
LIST OF PUBLICATIONS	XIV
CHAPTER ONE: INTRODUCTION	15
1.1 MOTIVATION	15
1.2 AIM AND OBJECTIVES	17
1.3 RESEARCH METHOD	18
1.4 THESIS OVERVIEW	20
CHAPTER TWO: LITERATURE REVIEW	23
2.1 INTRODUCTION	23
2.2 INTEROPERABILITY	24
2.3 STANDARDS	25
2.3.1 Standards Typology	25
2.3.2 Structure of Standards	27
2.3.3 Standards Development Process	30
2.3.4 Quality of Standards	33
2.4 ONTOLOGIES	34
2.4.1 Ontology Engineering	35
2.4.2 Quality of Ontologies	43
2.5 ONTOLOGY FOR SEMANTIC INTEROPERABILITY	44
2.6 ONTOLOGIES AND STANDARDS IN THE LITERATURE	47
2.7 LITERATURE FINDINGS AND RESEARCH DIRECTION.....	50
CHAPTER THREE: RESEARCH METHOD	53
3.1 INTRODUCTION	53
3.2 RESEARCH PARADIGMS IN INFORMATION SYSTEMS	54
3.2.1 Design Science Research in IS	55
3.3 DESIGN SCIENCE RESEARCH METHODOLOGY	58
3.3.1 Problem Identification.....	59
3.3.2 Suggestion and Theorising	60
3.3.3 Development	60
3.3.4 Evaluation	61
3.3.5 Contribution and Communication.....	74
3.4 THE APPLICATION OF DESIGN RESEARCH	75

3.4.1	<i>Problem Identification</i>	76
3.4.2	<i>Suggestion & Theorising</i>	77
3.4.3	<i>Development</i>	77
3.4.4	<i>Evaluation</i>	79
3.1	SUMMARY	81
CHAPTER FOUR: TOWARDS ONTOLOGY-BASED STANDARDS		83
4.1	INTRODUCTION	83
4.2	SEMANTIC INTEROPERABILITY	84
4.3	STANDARDS DEVELOPMENT AND ONTOLOGY.....	85
4.4	COMMUNITY-BASED KNOWLEDGE MANAGEMENT	87
4.5	DOGMA	89
4.6	ONTOSTAND.....	91
4.6.1	<i>Formulate Vision and Feasibility Study</i>	94
4.6.2	<i>Technical Committee Formation</i>	94
4.6.3	<i>Preparation & Scoping</i>	95
4.6.4	<i>Domain Conceptualisation</i>	97
4.6.5	<i>Ontology Validation and Verification</i>	107
4.7	EVALUATION OF ITERATION 1	108
4.8	SUMMARY	111
CHAPTER FIVE: ONTOSTAND		113
5.1	INTRODUCTION	113
5.2	EXTENDED ONTOSTAND	113
5.2.1	<i>Axiomatisation</i>	115
5.2.2	<i>Specification Reification</i>	133
5.2.3	<i>Approval & Publication</i>	135
5.2.4	<i>Maintenance</i>	135
5.3	EVALUATION	136
5.3.1	<i>Consistency of the Model</i>	136
5.3.2	<i>Completeness of the Model</i>	137
5.3.3	<i>Usability of the Model</i>	137
5.3.4	CUSTOMISABILITY OF THE MODEL	139
5.3.5	<i>Practical adequacy of the Method</i>	140
5.4	SUMMARY	140
CHAPTER SIX: APPLICATION OF ONTOSTAND.....		142
6.1	INTRODUCTION	142
6.2	APPLICATION OF THE METHODOLOGY: GREEK E-GIF SCENARIO	143
6.2.1	<i>Formulate Vision & Feasibility Study in Greek e-GIF</i>	144
6.2.2	<i>Technical Committee Formation in Greek e-GIF</i>	145
6.2.3	<i>Preparation & Scoping in Greek e-GIF</i>	145
6.2.4	<i>Domain Conceptualisation for Greek e-GIF</i>	147
6.2.5	<i>Axiomatisation of Greek e-GIF</i>	151
6.2.6	<i>Ontology Validation and Verification of e-GIF</i>	153
6.2.7	<i>Specification Reification of e-GIF</i>	153
6.2.8	<i>Approval and Maintenance of e-GIF</i>	153

6.2.9	<i>Evaluation of the Greek e-GIF Ontological Model</i>	154
6.3	APPLICATION OF ONTOSTAND: THE CASE OF EBBP.....	155
6.3.1	<i>Formulate Vision & Feasibility Study in ebBP</i>	157
6.3.2	<i>Technical Committee Formation in ebBP</i>	157
6.3.3	<i>Preparation & Scoping in ebBP</i>	157
6.3.4	<i>Domain Conceptualisation of ebBP</i>	158
6.3.5	<i>Axiomatisation of ebBP</i>	164
6.3.6	<i>Ontology Validation and Verification of ebBP</i>	166
6.3.7	<i>Specification Reification of ebBP</i>	167
6.3.8	<i>Approval and Publication of ebBP</i>	167
6.3.9	<i>Maintenance of ebBP</i>	167
6.3.10	<i>Evaluation of the ebBP ontological Model</i>	167
6.4	EVALUATION OF THE METHOD	177
6.5	SUMMARY	178
CHAPTER SEVEN: CONCLUSION & FUTURE RESEARCH DIRECTIONS		179
7.1	RESEARCH SUMMARY	179
7.2	RESEARCH CONTRIBUTIONS.....	183
7.3	IMPLICATIONS OF THE WORK	185
7.4	LIMITATIONS AND CHALLENGES	187
7.5	FUTURE RESEARCH DIRECTIONS	189
REFERENCES		192
APPENDIX I: LETTER OF SUPPORT FROM FUJITSU		206
APPENDIX II: CERTIFICATION FRAMEWORK FOR PUBLIC ADMINISTRATION SITES AND PORTALS		208
APPENDIX III: THE OASIS TAG WIDGET ONTOLOGY AND RULES		216
APPENDIX IV: THE GREEK E-GIF ONTOLOGY		221
APPENDIX V: THE OASIS EBBP ONTOLOGY		233

LIST OF TABLES

TABLE 2-1. STANDARDS RESTRICTION KEYWORDS.....	28
TABLE 2-2. TEST ASSERTIONS DEFINITIONS	29
TABLE 2-3. SUMMARY OF VARIOUS STANDARD CREATION PROCESSES	32
TABLE 2-4. QUALITY MODEL FOR STANDARDS.....	34
TABLE 2-5. A SUMMATIVE COMPARISON ON ONTOLOGY ENGINEERING METHODS.....	42
TABLE 2-6. ONTOLOGY EVALUATION FACTORS.....	43
TABLE 3-1. THE OUTPUTS OF DESIGN RESEARCH.....	55
TABLE 3-2. COMPARISON OF ACTIVITIES/PHASES IN VARIOUS DESIGN RESEARCH METHODOLOGIES.....	58
TABLE 3-3. DESIGN SCIENCE RESEARCH EVALUATION METHODS	63
TABLE 3-4. EVALUATION CRITERIA ACCORDING TO MARCH AND SMITH.....	64
TABLE 3-5. HOW DIFFERENT DESIGN RESEARCH ARE EVALUATED IN THE LITERATURE	66
TABLE 3-6. SUMMARY OF RELEVANT EVALUATION FACTORS FOR EVALUATING ONTOLOGICAL CONCEPTUAL MODELS	69
TABLE 3-7. DSR ARTEFACTS PRODUCED IN THIS THESIS	78
TABLE 3-8 – SUMMARY OF EVALUATION APPROACHES TAKEN.....	80
TABLE 4-1. AN EMPTY NARRATOLOGICAL SCHEMA (NS) FORM.....	100
TABLE 4-2. A NARRATOLOGICAL SCHEMA FOR THE OASIS TAG WIDGET SPECIFICATION	102
TABLE 4-3: TAG WIDGET LEXON TABLE.....	104
TABLE 4-4. SUMMARY OF KEY OUTPUT OF EACH PHASE OF ONTOSTAND (V1.0).....	108
TABLE 4-5. ARTEFACT TO BE EVALUATED AT THE END OF ITERATION 1.....	108
TABLE 5-1. TAG WIDGET TEST ASSERTIONS	118
TABLE 5-2. OASIS TAG WIDGET TA FOR SECTION 100	127
TABLE 5-3. OASIS TAG WIDGET TA FOR SECTION 101	128
TABLE 5-4. OASIS TAG WIDGET TA FOR SECTION 102	130
TABLE 5-5. OASIS TAG WIDGET TA FOR SECTION 103	132
TABLE 5-6. OASIS TAG WIDGET TA FOR SECTION 104	133
TABLE 6-1. A NARRATOLOGICAL SCHEMA FOR THE GREEK E-GIF	148
TABLE 6-2 HIGHLIGHTED PHRASES FOR THE SEGMENTATION	148
TABLE 6-3. A LEXON TABLE FOR THE GREEK E-GIF	150
TABLE 6-4. A NARRATOLOGICAL SCHEMA FOR OASIS EBBP	159
TABLE 6-5. HIGHLIGHTED PHRASES FOR TOASIS EBBP	160
TABLE 6-6. A LEXON TABLE FOR OASIS EBBP.....	162
TABLE 7-1. HOW THE OBJECTIVES OF THE RESEARCH ARE ADDRESSED IN CHAPTERS.....	183
TABLE 7-1. DSR ARTEFACTS PRODUCED IN THIS THESIS.....	185

LIST OF FIGURES

FIGURE 1-1. OVERVIEW OF THE THESIS	22
FIGURE 2-1. VARIOUS LEVELS OF INTEROPERABILITY.....	24
FIGURE 2-2. GENERAL ANATOMY OF A TEST ASSERTION	28
FIGURE 2-3. QUALITY MODEL FOR STANDARDS BASED ON ISO 9126	33
FIGURE 2-4. USCHOLD AND KING'S ONTOLOGY CREATION PROCESS	35
FIGURE 2-5. GRÜNINGER AND FOX'S ONTOLOGY DESIGN AND EVALUATION PROCESS.....	37
FIGURE 2-6. METHONTOLOGY DEVELOPMENT PROCESS AND LIFECYCLE	37
FIGURE 2-7. METHONTOLOGY CONCEPTUALISATION TASK	38
FIGURE 2-8. OTKM KNOWLEDGE METAPROCESS	39
FIGURE 2-9. DILIGENT METHODOLOGY PROCESS STAGES, ACTIONS AND STRUCTURES	39
FIGURE 2-10. DOGMA METHODOLOGY	40
FIGURE 2-11. ONTOENG METHODOLOGY.....	41
FIGURE 2-12A. ONTOLOGICAL COMPLETENESS	43
FIGURE 2-12B. ONTOLOGICAL INCOMPLETENESS OR CONSTRUCT DEFICIT	43
FIGURE 2-13. CONSTRUCT OVERLOAD	44
FIGURE 2-14. CONSTRUCT REDUNDANCY	44
FIGURE 2-15. CONSTRUCT EXCESS	44
FIGURE 3-1. INFORMATION SYSTEMS RESEARCH FRAMEWORK	54
FIGURE 3-2. AN ACTIVITY FRAMEWORK FOR DESIGN SCIENCE RESEARCH	57
FIGURE 3-3. COMPONENTS OF UTILITY THEORIES AND HYPOTHESES	57
FIGURE 3-4. ADOPTED DESIGN SCIENCE RESEARCH METHODOLOGY	59
FIGURE 3-5. STRATEGIC DSR EVALUATION FRAMEWORK	62
FIGURE 3-6. RESEARCH DESIGN	76
FIGURE 4-1. BUSINESS SEMANTIC MANAGEMENT PARADIGM	88
FIGURE 4-2. THE DOGMA DOUBLE ARTICULATION	91
FIGURE 4-3. ONTOSTAND V1.0	93
FIGURE 4-4. ONTOSTAND PREPARATION & SCOPING PHASE	96
FIGURE 4-5. ONTOSTAND DOMAIN CONCEPTUALISATION PHASE	99
FIGURE 4-6. TAG WIDGET BASELINE TAXONOMY	105
FIGURE 4-7. THREE OF TAG WIDGET RELATIONSHIPS	105
FIGURE 4-8. OASIS TAG WIDGET SPECIFICATION EXTRACT	106
FIGURE 5-1. AN OVERVIEW OF THE ONTOSTAND METHODOLOGY	115
FIGURE 5-2. HOW TEST ASSERTIONS RELATE TO SPECIFICATION AND TESTING	117
FIGURE 5-3. FORMAL CONSISTENCY QUESTION FOR COMPETENCY QUESTION 1	138
FIGURE 5-4. FORMAL CONSISTENCY QUESTION FOR COMPETENCY QUESTION 2	138
FIGURE 5-5. AN ALTERNATIVE DL QUERY FOR COMPETENCY QUESTION 2	138
FIGURE 5-6. FORMAL CONSISTENCY QUESTION FOR COMPETENCY QUESTION 3	139
FIGURE 5-7. FORMAL CONSISTENCY QUESTION FOR COMPETENCY QUESTION 4	139
FIGURE 6-1. A TYPE HIERARCHY FOR THE TERMS IN THE GREEK E-GIF STANDARD	150
FIGURE 6-2. MEREOLOGICAL RELATIONS FOR THE CONCEPTS IN TABLE 6-3	151
FIGURE 6-3. AN ONTOLOGY FOR THE SELECTED PART OF THE GREEK E-GIF STANDARDS	151
FIGURE 6-4. A MANDATORY CONSTRAINT MODELLED IN ORM	152
FIGURE 6-5. A TYPE HIERARCHY FOR SELECTED PART OF EBBP SPECIFICATION	163
FIGURE 6-6. A PART OF EBBP ONTOLOGY	164
FIGURE 6-7. AN INCONSISTENCY IN THE STANDARDS SPECIFICATION	166

FIGURE 6-8. SCHEMATIC OF CORE BUSINESS TRANSACTION SEMANTICS169
FIGURE 6-9. SIMPLE ORDERING PROCESS (UBL)171
FIGURE 7-1. OVERVIEW OF ONTOSTAND METHODOLOGY184

LIST OF LISTINGS

LISTING 4-1. THE OASIS TAG WIDGET SPECIFICATION EXTRACT (OASIS TAG TC 2011)	101
LISTING 6-1. TWO INDIVIDUALS OF ORDERING PROCESS IN THE EBBP ONTOLOGY.....	172
LISTING 6-2. DL QUERY FOR COMPETENCY QUESTION 1.....	173
LISTING 6-3. DL QUERY FOR COMPETENCY QUESTION 2	173
LISTING 6-4. DL QUERY FOR COMPETENCY QUESTION 3.....	174
LISTING 6-5. OPTIMISED DL QUERY FOR COMPETENCY QUESTION 4.....	174
LISTING 6-6. NOT OPTIMISED DL QUERY FOR COMPETENCY QUESTION 4.....	175
LISTING 6-7. DL QUERY FOR COMPETENCY QUESTION 5.....	176

ABBREVIATIONS

The following abbreviations are used throughout the thesis:

AI	Artificial Intelligence
AIAG	Automotive Industry Action Group
B2B	Business to Business
BSI	British Standards Institute
CCTS	Core Components Technical Specification
CEN	European Committee for Standardisation/Comité Européen de Normalisation
CPA	Collaboration Protocol Agreement
CPP	Collaboration Protocol Profile
CWA	Closed World Assumption
DILLIGENT	Distributed, Loosely controlled and Evolving Engineering of oNTologies
DL	Description Logic
DOGMA	Developing Ontology-Grounded Methods and Applications
DSR	Design Science Research
DSRM	Design Science Research Methodology
ebBP	ebXML Business Process
ebXML	e-Business XML
e-GIF	e-Government Interoperability Framework
ETSI	European Telecommunications Standards Institute
EU	European Union
F2F	Face to Face
HTML	Hyper Text Markup Language
IC	Integrity Constraint
ICT	Information and Communication Technology
ICV	Integrity Constraint Validator
IEC	International Electro-technical Commission
IEEE	Institute of Electrical and Electronics Engineers
IS	Information Systems
ISBN	International Standard Book Number
ISO	International Organization for Standardisation
IT	Information Technology
Naf	Negation as Failure
NLP	Natural Language Processing
NS	Narratological Schema
NTUA	National Technical University of Athens
OAGIS	Open Applications Group Integration Specification
OASIS	Organisation for Advancing open Standards for the Information Society
OntoEng	Ontology Engineering
OntoStand	Ontology-based Standards Development

ORM	Object Role Modelling
ORM-ML	Object Role Modelling Markup Language
OTKM	On-To-Knowledge Methodology
OWA	Open World Assumption
OWL	Web Ontology Language
QUOMOS	Quantities and Units of Measure Ontology Standard
RDF	Resource Description Framework
SDO	Standards Development Organisation
SET	Semantic Support for Electronic Business Document Interoperability TC
SPARQL	SPARQL Protocol and RDF Query Language
SSO	Standards Setting Organisation
STAR	Standards for Technology in Automotive Retail
TAB	Technical Advisory Board
TAG	Test Assertion Guidelines
TC	Technical Committee
UBL	Universal Business Language
UN/CEFACT	United Nations Centre for Trade Facilitation and Electronic Business
UNA	Unique Name Assumption
W3C	World Wide Web Consortium
WS-BPEL	Web Services Business Process Execution Language
XML	Extensible Markup Language
XPDL	XML Process Definition Language

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to those who have helped in various ways along my path for conducting my PhD research:

- Professor Mark Lycett for his supervision, support and encouragement during all stages of my research.
- Dr. David Bell for his generous support as my second supervisor.
- Dr. Sergio de Cesare for his useful insights and support in various stages of my work.
- Fujitsu Laboratories of Europe Ltd., who provided a significant amount of funding for my PhD research, without which my PhD would not have been possible. I would especially like to thank Dr. David F. Snelling, who provided me with great support and encouragement during the span of my PhD.
- My colleagues from the Greek Interoperability Centre of the National Technical University of Athens, where I was granted a three months research visit secondment. I would like to specifically thank Mrs. Ioanna Makaroni, Dr. Yannis Charalabidis, Professor John Psarras and professor Dimitris Askounis who helped me in various ways, academic and non-academic, during my stay in Greece and made it an unforgettable experience for me.
- Dr. Yan Tang from the VUB Star lab of the Vrije Universiteit Brussel, whom I was lucky to work with during my stay in Athens. Yan provided me with a great insight to DOGMA framework, which is developed in their VUB Star lab.

- Stephen D. Green of the OASIS, from Document Engineering Services, for his endless supports with respect to ebXML specifications and the associated matters in the relevant TCs. He also helped me in designing the competency questions for the ebBP scenario.
- OASIS ebCore TC and its members for their sporadic support with respect to my work.
- Dr Jacque Durand of the OASIS (member of the OASIS Technology Advisory Board, TAG TC and various OASIS ebXML TCs, OASIS Interoperability and Conformance (IIC) TC, etc.), from Fujitsu Software America, for his continues and supportive help with respect to my work.
- Google, for awarding me the Google Anitaborg Memorial Scholarship, which was a great inspiration as well as a financial support.
- My great friends Malika, Roshanak and Panos who helped me proof read my thesis.
- My friends and colleagues at Brunel DISC who made an open, fun and at the same time intellectually stimulating environment for my research at Brunel.

LIST OF PUBLICATIONS

- [1] Heravi, B. R., Lycett, M., *Semantically Enriched e-Business Standards Development: The Case of ebXML Business Process Specification Schema*, to be published in Handbook of Research on E-Business Standards and Protocols: Documents, Data and Advanced Web Technologies, 2012.
- [2] Heravi, B. R., Green S. D., Bell D., Lycett, M., Towards an Ontology for Automating Collaborative Business Processes, 5th International Workshop on Vocabularies, Ontologies and Rules for The Enterprise (VORTE) at the 14th IEEE International EDOC Conference, October 2010, Vitória, Brazil.
- [3] Heravi, B. R., Green S. D., Snelling, D., Bell D., Lycett, M., *Towards Ontology Based e-Business Standards*, International Conference on e-Business (ICE-B 210), July 2010, Athens, Greece.
- [4] Bell, D., Heravi, B. R., Lycett, M., *Sensory Semantic User Interfaces (SenSUI)*, 2nd International Workshop on Semantic Sensor Networks (SSN09), Washington DC, USA, October 2009.

In Progress:

- [5] Heravi, B. R., Lycett, M., *Ontology-based Standards Development for Automating Collaborative Business Processes*.
- [6] Heravi, B. R., Yan Tang, Lycett, M., Dimitris Batis, *OntoStand: an Ontology-based Standards Development*.

CHAPTER ONE: Introduction

1.1 Motivation

Effective inter-organisational collaborations are vital means of gaining competitive advantage in today's global business. Interoperability is considered to be one of the most important aspects of successful collaborative inter-organisational business, achieving which requires a common understanding and agreement between parties involved. Standards are traditionally used for achieving shared understanding in a domain with the ultimate aim of interoperability. Standards, however, mainly focus on the syntactic aspects of interoperation and little attention is expended on the semantics of the terms and concepts in a domain and their relationships in standards based interoperability. Moreover, most standards are developed by (geographically spread) groups of stakeholders, in a relatively long span of time. These would, inevitably, cause misunderstandings and thus ambiguity and inconsistencies in the standards specifications, having an adverse affect on interoperability.

Lack of semantics support and also ambiguity eliminate the power of standards for achieving effective semantic interoperability. Ontologies, on the other hand, focus on the clear representation of any domain and the semantic aspects of interoperability, which makes them a promising means for achieving semantic interoperability (Rebstock et al. 2008). Thus, the convergence of standards and ontologies is considered to be a promising approach for achieving semantic interoperability. This research proposes that ontologies contribute to the

development, extension and improvement of standards specifications in the following ways:

- *Formalising concepts within existing standards*; resulting in a more stable definition of semantics in a standard and allowing the writing of expressions based on clear, unambiguous terms and categories. Specifications serve as a foundation for developed ontologies and therefore the ontologies can be evaluated against the standards specifications and/or their XML Schemas (The Ontolog Community 2010).
- *Reengineering of existing standards based on ontological analysis*; identifying their potential problems and semantic ambiguities with the aim of improving their quality (The Ontolog Community 2010). The prerequisite for ontological reengineering of a standard is developing a formal, ontological representation of the standard, as noted above.
- *Facilitation of integration between different standards or multiple implementations of the same standard*; possible when standards are already defined in an ontological manner, preferably using a methodological approach, such as the one proposed in this thesis.
- *Development of standards*, wherein ontologies are used throughout the standard development phases, from start to finish. This approach can be taken where appropriate for developing new standards or new versions of existing standards (Heravi et al. 2010b).

This research targets the above points, using ontologies in conjunction with standards to provide an approach for ontology-based standards development - OntoStanD. The ultimate aim of the OntoStanD methodology is achieving higher quality standards and thus a higher level of semantic interoperability as a consequence. OntoStanD is a comprehensive ontology-based standards

development methodology, which utilises the best practices of the existing ontology and standards creation methods seeing the following commonalities between them:

- *Interoperability*; a standard records a statement of sharable items for conformant parties, while an ontology contains formal agreements (the conceptualisation) made by domain experts within a community of interests.
- *Community*; a standard is created by a group of stakeholders while an ontology is created by a community of domain experts.

Ontology-based standards development potentially brings all the components of a standard specification under one single umbrella. It also has the potential to unambiguously formalise the concepts and relationships of a domain under standardisation, enable logic based reasoning, automate consistency and conformance checking, facilitate change management and reduce errors in the specifications and their conceptual models. These characteristics of ontology-based standards development potentially lead to higher quality standards and a better degree of semantic interoperation. OntoStanD may be used for developing new standards specifications from scratch or new versions of existing standards, re-engineering existing standards and also formalising concept in existing standards as explained earlier.

1.2 Aim and Objectives

The aim of this research is to provide a *methodological approach for developing ontology-based standards*, with the aim of achieving higher quality standards and a higher degree of semantic interoperability as an outcome. The objectives of this research are as follows:

- **Objective 1:** Investigate the structure of standards and the existing standards development processes, with the aim of finding the associated gaps in their development processes which are believed to eliminate semantic interoperability.
- **Objective 2:** Investigate existing ontology engineering methods with the aim of learning about their strengths, weaknesses and best practices for utilising them in standards development process.
- **Objective 3:** Identify the requirements for a robust standards development methodology in order to improve the state-of-the-art (taking into account the findings of Objective 1 and Objective 2).
- **Objective 4:** Develop a methodological approach for ontology-based standards development (which takes into account and covers the findings of Objective 3) that provides semantic clarity and coherence.
- **Objective 5:** Evaluate and demonstrate the practical adequacy of the proposed method via its application on three standards specifications.

The research objectives in this research look to provide new solutions to existing problems in the domain of standards-based interoperability. The objectives are fulfilled by building a methodology for designing standards and providing practical application of the method by its application on three existing standards.

1.3 Research Method

The Design Science Research (DSR) paradigm (March, Smith 1995, Peffers, Tuunanen et al. 2007, Vaishnavi, Kuechler 2007) is the research method adopted in this thesis. DSR is a problem solving research paradigm, which is aimed at designing artefacts as a solution to research problems. A design

research artefact is typically of the form of constructs, models, methods, instantiations and arguably design/utility theories.

A number of DSR methodologies exist in the literature. This research studies and synthesises the existing DSR methodologies and constructs a suitable DSR methodology to best suit this research. The final artefact of this research is a ‘method’ which is evaluated for its practical adequacy by its application on three existing standards (1) The OASIS TAG Widget, (2) the Certification Framework for Public Administration Sites and Portals specification of the Greek e-Government Interoperability Framework (e-GIF) standards and (3) the OASIS ebXML Business Process Specifications (ebBP). The reasons behind choosing these three standards are that: (a) They cover various aspects of a standard’s specification, (b) the standards’ specifications and also standards developers were accessible by the researcher and (c) they are three different types of standards, which demonstrate the practical adequacy of OntoStanD in various contexts.

The application of OntoStanD results in a set of ontological models, which are artefacts of type ‘model’ in DSR term. The models resulting from the application of OntoStanD on each of the selected scenarios are evaluated for their consistency, completeness, usability and expandability. These factors for evaluating ontological models of standards are sketched out based on an extensive analysis on various aspects of the quality for such models.

The evaluation in this thesis is an ex-post descriptive approach, utilising both scenarios and informed arguments and is further positioned as artificial in the sense that the users are not real, and the methodology is a prototype scenario – the scenarios which OntoStanD are applied to are however real.

1.4 Thesis Overview

In achieving its objectives, the remainder of the thesis is structured as depicted in Figure 1-1 and explained in the following:

Chapter 2 critically reviews three intersecting fields of study necessary for this research – semantic interoperability, standards and ontology. This literature review is organised in four main sections. The first section presents a brief overview of interoperability and its various levels. The second section provides an in-depth review on standards, their typology, structure, development process and quality factors. The third part provides a chronological overview of the ontology engineering methodologies and the fourth section shows how the three facets above are used in conjunction in the literature so far. The aim of this literature review is to gain an understanding of the state of the art in the above domains and further learn about the ways in which ontologies may facilitate standards based semantic interoperability. This review assists in identifying the gaps in the domain primarily for achieving greater standards-based semantic interoperability.

Chapter 3 investigates and presents the design science research methodology as the adopted research method in this thesis for designing an ontology-based standards development methodology. Design research is fundamentally a problem solving paradigm which addresses “important unsolved problems in unique or innovative ways or solved problems in more effective or efficient way” (Hevner et al. 2004). It involves a rigorous process to design artefacts, solve observed problems, make research contributions, evaluate the designs, and communicate the results to appropriate audiences (Hevner et al. 2004). This chapter further critically studies and analyses the evaluation of artefacts in DSR and provides an evaluation framework to be used in this research. Finally it discusses the research design of this thesis.

Chapter 4 discusses ontology-based standards and the reasons why they are considered to be a desirable approach for standards development. It further introduces the OntoStanD methodology, which provides a methodological approach for designing and developing ontology-based standards. This chapter provides a set of guidelines for defining the conceptual model of a standard in an ontological manner. OntoStanD V1.0 introduced in this chapter is applied to a simple pseudo specification of OASIS TAG TC, the Widget specification, as a working example, in order to ground the key aspects of OntoStanD that are discussed in detail.

Chapter 5 refines and extends the outcomes of the first iteration of the research by providing guidelines for modelling normative statements of a standard, while applying them to the OASIS TAG Widget specification and its ontology base layer created in Chapter 4. This chapter also discusses the potential problems associated with the Open World Assumption of some ontology languages such as OWL and provides a set of guidelines for overcoming the problems when not desirable, i.e. when Closed World reasoning is required.

Chapter 6 evaluates OntoStanD for its practical adequacy by its application on a subsection of two other existing standards: (a) The Greek e-Government Interoperability Standards, as a non-technical standard and (b) the ebXML Business Process Specification Schema as a technical, XML-based standard. These two scenarios, in combination with the OASIS TAG Widget standard, prove that not only OntoStanD is capable of capturing the domain knowledge and turning them into a robust ontology-based model, but also it matches very well with the standards communities' best practices and activities. The three scenarios also demonstrate that OntoStanD could be used for various types of standards and in different contexts.

Chapter 7 presents a summary of the thesis and articulates the research values and contributions of this research. It further discusses the implications, challenges and limitations associated with this research. It finally provides an exposition of the research limitations and new lines of research and recommendations proposed for further research.

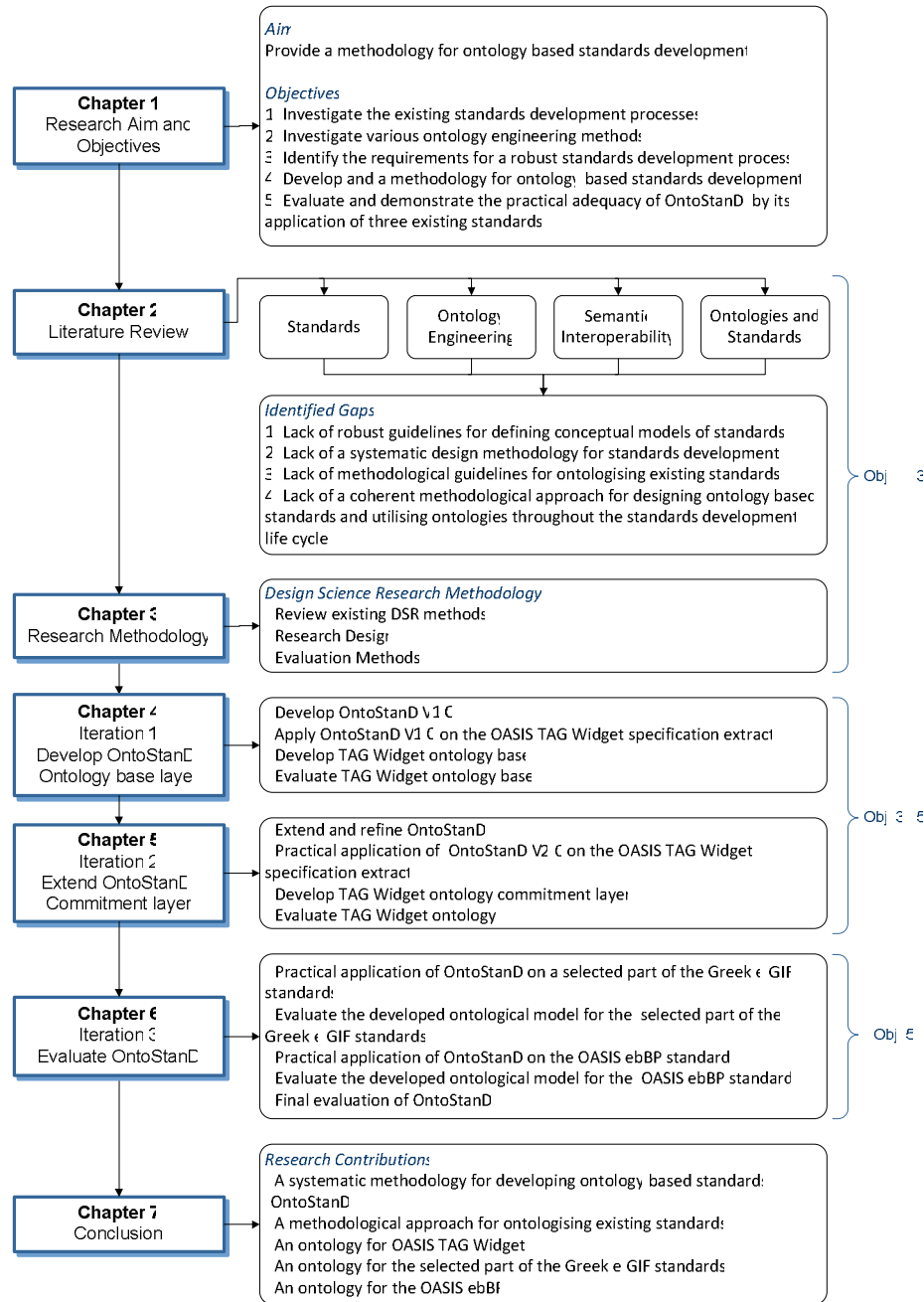


Figure 1-1. Overview of the thesis

CHAPTER TWO: Literature Review

2.1 Introduction

This chapter critically reviews three intersecting fields of study that are necessary for this research: semantic interoperability, standards and ontology, while discussing the limitations of the semantic interoperability and standards development. The aim of this literature review is to: (1) Provide an understanding of the state of the art in semantic interoperability, standards and ontology; (2) discusses how the Semantic Web and ontology facilitate semantic interoperability and (3) how they have been used in conjunction with standards for achieving semantic clarity and interoperability thus far. This literature review assists in identifying the gaps in the domain, primarily for achieving greater standards-based semantic interoperability. It also facilitates selection of a suitable research methodology for addressing the identified gaps.

The chapter is organised as follows: Section 2.2 briefly discusses the literature review process taken in this thesis. Section 2.3 briefly reviews various aspects of interoperability. Section 2.4 studies the structure of the standards and their development processes in general and also provides an extensive review on the standards development process of six influential standardisation bodies. Section 2.5 introduces ontology and reviews and synthesises various existing ontology engineering methods and approaches. Section 2.6 provides a literature review on the current research on using standards for semantic interoperability followed by Section 2.7 which reviews the research on the convergence of

standards and ontologies. Section 2.8 articulates the research findings in this chapter and identifies a set of research gaps on that basis.

2.2 Interoperability

No single definition of the term ‘interoperability’ exists in the literature, but it is taken here as the “ability of two or more systems or components to exchange data and use the information” (Van der Veer, Wiles 2008 pp. 5). According to ETSI (ETSI 2010), there are four levels of interoperability as depicted in Figure 2-1 and explained in the following.

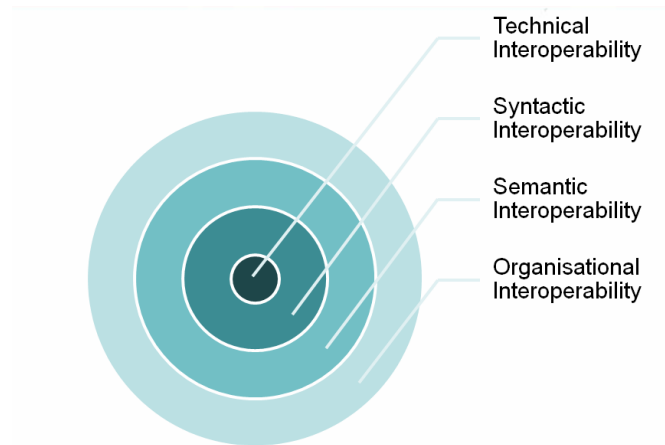


Figure 2-1. Information Systems Research Framework (Source: Hevner et al. 2004)

- *Technical Interoperability*, normally associated with hardware/software components and centred on communication protocols and infrastructure.
- *Syntactic Interoperability*, usually concerned with data formats and exchange. The data should be machine readable at this level.
- *Semantic Interoperability*, concerned with the meaning of content. The exchanged data and information should be machine readable, processable and understandable to some extent.

- *Organisational Interoperability*, concerned with the ability of two organisations to effectively communicate and transfer data and information even though using different information systems. Organisational interoperability depends on successful technical, syntactic and semantic interoperability.

Since the aim of this research is primarily addressing the standards-based semantic interoperability, the focus of this thesis is mainly on the semantic interoperability level, which eventually assists in achieving (inter) organisational interoperability.

2.3 Standards

Standards are important facilitators for achieving interoperability. A standard is a technical specification approved by a recognised standardisation body, which is designed to be used consistently, as a rule, a guideline, or a definition across particular communities of interest (ETSI 2010). The aim of a standard is to provide unambiguous specifications for error-free exchange of documents and information for achieving mutual benefits.

2.3.1 Standards Typology

Standards may be categorised from various aspects, such as their requirements, development process and users. This research targets interoperability standards, which are also called information standards in this thesis. These standards are considered to be one of the most important types of standards in IT and IS (De Vries 2006), which are aimed at providing a shared understanding between various systems for seamless (in most cases inter-organisational) interoperation.

With respect to the development process and developing entities, standards can be classified according to a number of perspectives. From a geographic

perspective, there are international, regional, and national standards bodies (De Vries 2006). From a technology or industry perspective, there are standards developing organisations (SDOs) and also standards setting organisations (SSOs) also known as consortia.

Standards are usually developed by standardisation bodies (SDO or SSO) and although there are a significant number of standardisation bodies in the information systems domain, relevant standards can be grouped in a number of categories. The most important of these are:

- *Formal standards*, also known as *de jure* standards, which are normative documents that have passed through a full and open consensus process. Formal standards normally have a legal basis and can be made mandatory to conform to for a certain group of users (W3C COPRAS 2007).
- *Technical or industry specifications*, which are based on consensus among members of standards bodies, consortia or trade organisations. Compared to formal standards, they require less time to produce and do not have a formal character or legal basis. However, when widely accepted and used they can become *de facto* standards (W3C COPRAS 2007). Technical specifications can subsequently become formal standards if passed through a formal approval process.

The openness of standards is another factor in the process of standards development. The purpose of open standards is to support common agreements that enable communications between parties who conform to the standard (Krechmer 2005). Bird (1998 pp. 76) defines an open standard as "a publicly available specification that is developed and maintained by an open, public consensus process and that is consistent with international standards, where relevant". Berners-Lee (2010 pp. 3) further defines open standards as standards

that can have “any committed expert involved in the design, that have been widely reviewed as acceptable, that are available for free on the Web, and that are royalty-free (no need to pay) for developers and users”. Thus, three main characteristics can be used to distinguish open standards from closed standards: (1) They are publically available; (2) no entity owns the standard; (3) the standard development process involves public participation (Shah, Kesan 2008). Both formal standards and technical specifications that are developed in an open process can be regarded as open standards (W3C COPRAS 2007).

2.3.2 Structure of Standards

A standard specification is generally composed of descriptive text, a set of normative statements, often with a conformance clause (all inline in the text), and optionally associated test assertions. The descriptive text provides background information, description and examples and provides contextual information (OASIS 2007a). Normative statements define the prescriptive requirements of a specification: They form its core and are normally distinguished by the use of the reserved restriction keywords as explained in Table 2-1. The keywords in Table 2-1 are mainly based on the OASIS guidelines and may be slightly different in other standards bodies.

In the standardisation terminology, conformance refers to the fulfilment of specified requirements by an implementation of the standard. This is verified with the use of conformance clauses that must, directly or indirectly, reference one or more normative statements and may also refer to another conformance clause (OASIS TAG TC 2011). A test assertion is an independent, complete, testable or measurable statement for evaluating the adherence of part of an implementation to a normative statement in a specification (OASIS TAG TC 2011, Durand et al. 2009). Test assertions are a means for formalising the

normative statements, but are not bound to any conceptual model defined for a standard. Figure 2-2 depicts the structure of a test assertion.

Restriction Keyword	Description
MUST	The requirement is an absolute requirement of the specification.
MUST NOT	The requirement is an absolute prohibition of the specification
REQUIRED	See MUST
SHALL	See MUST
SHALL NOT	See MUST NOT
SHOULD	There may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
SHOULD NOT	There may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.
RECOMMENDED	See SHOULD.
MAY	The item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation that does not include a particular option MUST be prepared to interoperate with another implementation that does include the option, though perhaps with reduced functionality. In the same vein an implementation, which does include a particular option MUST be prepared to interoperate with another implementation that does not include the option (except, of course, for the feature the option provides).

Table 2-1. Standards Restriction Keywords, adopted from (OASIS 2007a).

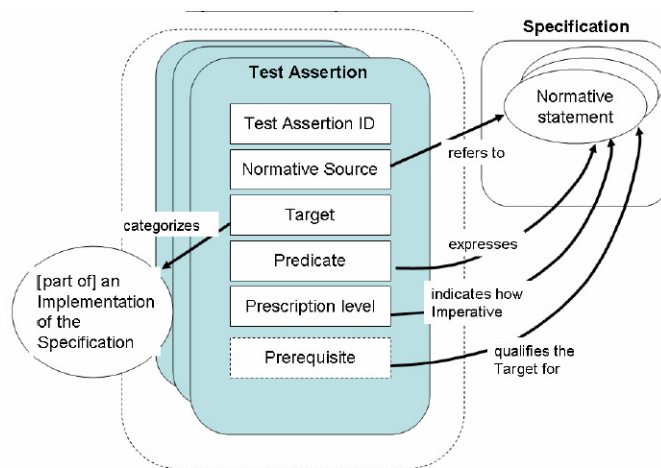


Figure 2-2. General anatomy of a test assertion, adopted from (OASIS TAG TC 2011)

As presented in Figure 2-2, a test assertion is composed of an identifier, normative source, target, predicate and optional description, prerequisite, prescription level, tags and variables as explained in Table 2-2.

TA Parts	Description
Identifier	A unique identifier of the test assertion, which facilitates the mapping of assertions to specification statements.
Normative Source	These refer to the precise specification requirements or normative statements that the test assertion addresses.
Target	The target categorises an implementation or a part of an implementation of the referred specification, that is the main object of the test assertion and of its Normative Sources.
Predicate	A predicate asserts, in the form of an expression, the feature (a behaviour or a property) described in the specification statement(s) referred by the Normative Sources. If the predicate is an expression which evaluates to "true" over the test assertion target, this means that the target exhibits this feature. "False" means the target does not exhibit this feature.
Description	An informal definition of the role of the test assertion, with some optional details on some of its parts.
Prescription level	A keyword that indicates how imperative it is that the Normative Statement referred to in the Normative Source, be met. See possible keyword values in the Table 2-1.
Prerequisite	A test assertion Prerequisite is a logical expression (similar to a Predicate) which further qualifies the Target for undergoing the core test (expressed by the Predicate) that addresses the Normative Statement. It may include references to the outcome of other test assertions. If the Prerequisite evaluates to "false" then the Target instance is not qualified for evaluation by the Predicate.
Tags	Test assertions may be assigned 'tags' or 'keywords', which may in turn be given values. These tags provide an opportunity to categorise the test assertions.
Variables	Test assertions may also include variables for convenience in storing values for reuse and shared use, as well as for parameterisation.

Table 2-2. Test assertions definitions, adopted from (OASIS TAG TC 2011)

A number of conformance profiles may be developed for any standard. A conformance profile is a special type of specification that constrains a specification for a specific functional requirement identified within a domain or context. In other words a conformance profile could be considered as a customised version of a standard targeted to a specific task. For example AS4 is a conformance profile of the OASIS ebXML Messaging Services (ebMS),

which defines a simplified conformance profile of the main standards (ebMS v3.0 specification) for the secure exchange of B2B messages on the Cloud using Web Services (Drummond Group Inc. 2011).

2.3.3 Standards Development Process

Standards bodies usually provide a set of guidelines for the process of creating standards. This process, however, is not the same in all standards development organisations. For the purpose of this research, and to assure the proposed method is suitable for most standards organisations, it is important to understand the standards development processes taken in various standardisation bodies. Therefore, the standards creation processes of six relevant influential standards development organisations are extensively studied and analysed. Table 2-3 provides an overview of the standards creation processes of these six SDOs.

Analysing the processes of different standards bodies, it is noted that although not all standardisation processes follow exactly the same development process, a set of commonalities can be identified in their processes, which may be considered as a typical standards creation process, as follows:

1. *Identifying needs for a new standard:* Standardisation efforts are initiated when there is a need for an idea or concept to be standardised (IEEE 2011). A demand for a new standard or standardisation effort has to be identified and expressed by an interested party and a set of requirements for the new standard has to be drafted to be published as the proposal.
2. *TC Formation:* No standard is developed by one person; therefore group collaboration and consensus is required(IEEE 2011). A Technical Committee (TC), also called Working Group, in a standardisation body

is normally responsible for developing standards. Standards organisations have different rules for becoming a TC member: some leave it open to any interested party and some require membership of the standards body. TC members may be individuals or representatives of public or private organisations, who are interested in the proposal and have knowledge and expertise in the technology/concept being standardised. Consequently, any party who is interested in the published proposal may participate in a TC.

3. *Specification Drafting:* The specification is drafted by the members of the TC. This phase forms the main content and structure of a standard.
4. *Approval and Publication:* A formal approval process is conducted when the specification draft is ready. This process is usually composed of one or more rounds of technical committee reviews followed by public review, which may vary depending on the type of standard. At the end of this step the standard specification may be submitted to be considered for achieving ‘standard’ status. Acquiring ‘standard’ status may take several years - up to 4 years for formal standards (W3C COPRAS 2007). During this time a specification may be implemented if it receives sufficient public review and achieves a certain level of approval. The specification’s use may be widespread even without full standardisation. Eventually a specification would be published as a ‘standard’ if accepted, if not it might be published as a technical specification.
5. *Maintenance:* This phase is concerned with the validity and deprecation of standards and also revisions, amendments, modifications, reaffirmation or withdrawal.

OASIS (OASIS 2010)	ISO (ISO 2011)	ETSI (ETSI 2010)	CEN (CEN 2009)	IEEE (IEEE 2011)	BSI (BSI 2011)
TC discussion	Proposal stage	Identifying needs for standardisation	Proposal	Initiate the project	Proposal and assessment of new work
TC formation	Preparatory stage	Defining the technical committee	Acceptance	Mobilising the working group	
Drafting	Committee stage	Identification, definition, approval and adoption of work items	Drafting	Drafting the standard	Preparation of draft
Approval of a committee draft	Enquiry stage	Drafting, editing and publication.	CEN Enquiry	Ballot the draft	Public enquiry
Public review of a committee draft	Approval stage				
Approval of a committee specification			Adoption by weighted vote	Gaining final approval	Preparation of draft standard for formal vote
Approval of an OASIS standard	Publication		Publication	Publication	Formal vote Publication
Errata	N/A	N/A	Review	Maintenance	N/A

Table 2-3. Summary of various standard creation processes

The details of each of the above stages may vary in each standardisation body depending on the type and scope of the standard. Although the standards creation process, as seen in Table 2-3, is usually defined by the standards bodies, the technical aspect of developing/drafting the standards themselves, Step 3 of the typical process above, has not received much attention either by the standards bodies or in the literature. It is believed here that a robust standards development process should provide more clear and coherent

guidelines for this step, which would potentially bring significant values for standards developers and users.

2.3.4 Quality of Standards

Although quality of standards is repeatedly considered as an important factor for achieving interoperability, there is little information found in the literature which actually articulates the characteristics of a high quality standard. Based on the ISO standard for software engineering quality, ISO 9126 (ISO/IEC. 2001), Folmer & Putner (2011) introduce a quality model for information standards as depicted in Figure 2-3.

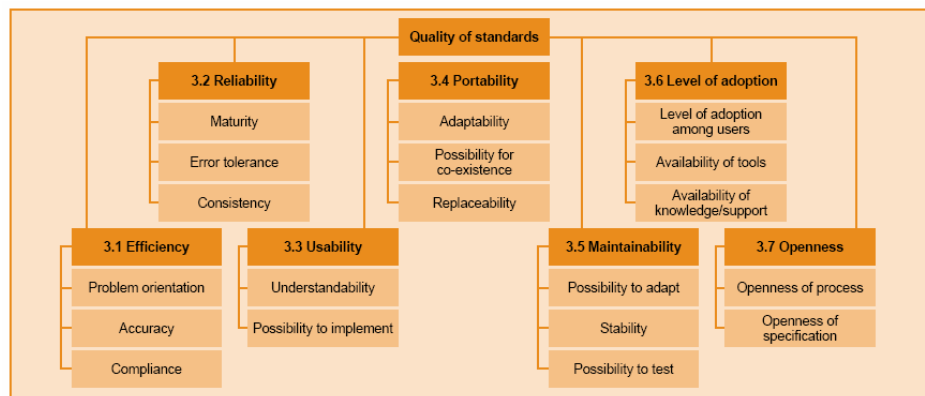


Figure 2-3. Quality model for standards based on ISO 9126, adopted from (Folmer, Putner 2011)

Table 2-4 briefly describes categories depicted in Figure 2-3 above.

Quality Factor	Description
Efficiency	The extent to which the standard offers and implements the functions which are explicitly or implicitly required in the specific situation.
Reliability	The extent to which a standard continues to perform at a specified level under specific conditions, such as incorrect implementations or differences in implementation between parties.
Usability	The extent to which a standard can be understood, learned and used/applied by users in the specific situation.
Portability	The extent to which a standard has the possibility for use in different environments.
Maintainability	The extent to which a standard can be easily modified to a changing situation.

Level of adoption	The extent to which the standard has been accepted by different parties.
Openness	The extent to which the standard meets the criteria for openness in the field of intellectual property and (maintenance and management) processes.

Table 2-4. Quality model for standards adapted from (Folmer, Putner 2011)

2.4 Ontologies

The Semantic Web is an evolving extension of the current web, in which information is given well-defined meaning and is machine process-able (Berners-Lee et al. 2001, Antoniou, Harmelen 2008): It provides a common framework that allows data to be shared and reused across applications and community boundaries (W3C 2011). Semantic web technologies provide more powerful means of defining concepts and their relationships in a domain, which results in more clarity and less ambiguity in the domain model.

Ontologies are an important element of semantic web technologies. The origin of the term ontology is in philosophy, where it is mainly concerned with the study of the nature of existence. More recently however, ontology has become a technical concept in the computer and information science discipline, where it is typically defined as “an explicit and formal specification of a conceptualisation” (Gruber 1995 pp. 1). Conceptualisation is further defined as the intended models within which a set of logical axioms are designed to account for the intended meaning of a vocabulary (Guarino 1998). Ontologies provide a formal description of concepts and their relationships within a domain (W3C 2011), which result in a shared understanding – the ultimate goal of information standards. Ontologies may be considered to be the glue between real world semantics and formal semantics and provide models of the world which reflects reality as perceived by human beings (Fensel 2001).

2.4.1 Ontology Engineering

There exist various ontology development methods in the literature. Uschold and King's method (Uschold, King 1995), Gruninger and Fox methodology (Gruninger, Fox 1995) and METHONDOLOGY (López et al. 1999) are examples of traditional methodologies. More recent ontology engineering methodologies take into account several soft aspects of the ontology development process, such as the aspect of community and evolution. Examples of more recent methodologies are Developing Ontology-grounded Methods and Applications (DOGMA) (Spyns et al. 2008), Ontology-based Knowledge Management method (OTKM) (Sure, Staab 2003), DILIGENT (Tempich et al. 2006), the method for supporting Meaning Evolution Support Systems (MESS) (De Leenheer 2008) and OntoEng (Al-Debei, Fitzgerald 2009). The following provides a brief chronological overview of a number of prominent and relevant ontology engineering methods with the aim of finding a suitable approach to be adopted in this research. Detailed surveys on ontology creation methods can be found in (Jones et al. 1998, Simperl, Tempich 2006, Sánchez, Cavero and Martínez 2007).

Uschold and King (1995) proposed one of the first ontology creation methods, which consists of four stages and three activities as depicted in Figure 2-4.

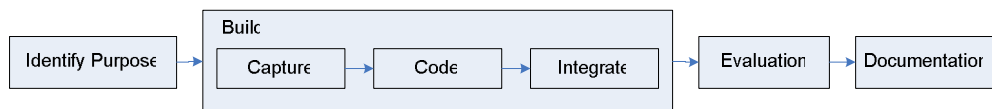


Figure 2-4. Uschold and King's Ontology creation process

This method was later extended by Uschold and Gruninger (1996), who provide three strategies for identifying concepts in the Capture activity of the methodology. These strategies are: bottom-up, top-down and middle-out. The bottom-up approach starts with the definition of the most specific classes based on the instances available in the real world, with subsequent grouping of these

classes into more general concepts. The top-down approach starts with identifying the most general concepts, organising them into a high-level taxonomy and system of axioms, and proceed to more specific concepts and axioms. This approach, if conducted appropriately, is believed to cover the basic structure of the domain under study. Finally, the middle-out approach starts by identifying the basic terms and then specifying and generalising them as required. Uschold and Gruninger (1996) believe the latter results in the most suitable balance in terms of level of details.

The main drawback of the Uschold and King (1995) method is reported to be the lack of recommendations for knowledge formalisation and conceptual modelling (Gomez-Perez et al. 2004, Fernández-López, Gómez-Pérez 2002).

The Gruninger and Fox methodology (Gruninger, Fox 1995) is inspired by the development of knowledge based systems using first order logic. They suggest using competency questions for scoping the domain and also defining the terminology and relationships while creating ontologies. Competency question are of two types in this method: informal and formal. Informal competency questions are defined on the basis of a set of motivating scenarios which should be defined as ontology requirements and are written in natural language. These competency questions play a central role in the requirement specification for an ontology and are considered as a suitable means for ontology evaluation. Formal competency questions are defined in an axiomatic manner based on the notation the ontology is defined in. They could be considered as ontological queries which could be run on top of an ontological model. This method is presented in Figure 2-5.

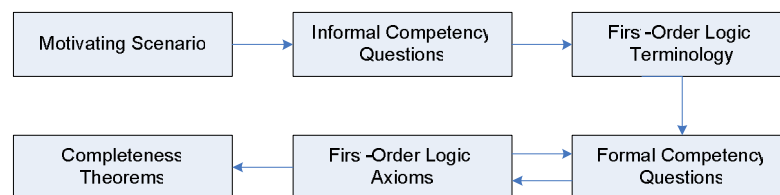


Figure 2-5. Gruninger and Fox's Ontology design and evaluation process (Gruninger, Fox 1995)

METHONTOLOGY (López et al. 1999, Fernandez et al. 1997) framework facilitates the construction of ontologies at the conceptual level and has its roots in software engineering and knowledge engineering methodologies. It consists of: (a) an ontology development process with the identification of the main activities, such as, conceptualisation, configuration, management, evaluation, integration implementation; (b) a life cycle based on evolving prototypes; and (c) a methodology, specifying the steps for performing the activities, the techniques used, the outcomes and their evaluation. METHONTOLOGY development process and lifecycle is presented in Figure 2-6.

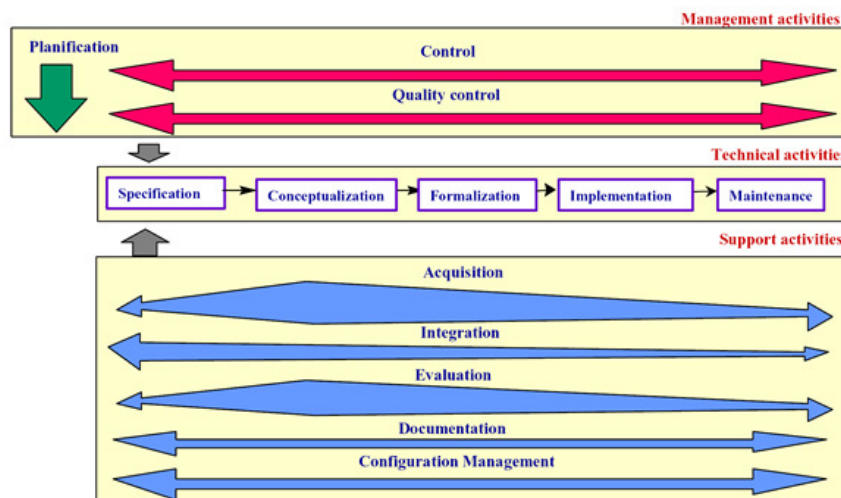


Figure 2-6. METHONTOLOGY development process and lifecycle

Seeing conceptualisation as a core activity of ontology development, METHONTOLOGY, unlike Uschold and King (1995) method, pays a special attention to ontology conceptualisation and therefore provides a set of guidelines for this purpose as demonstrated in Figure 2-7.

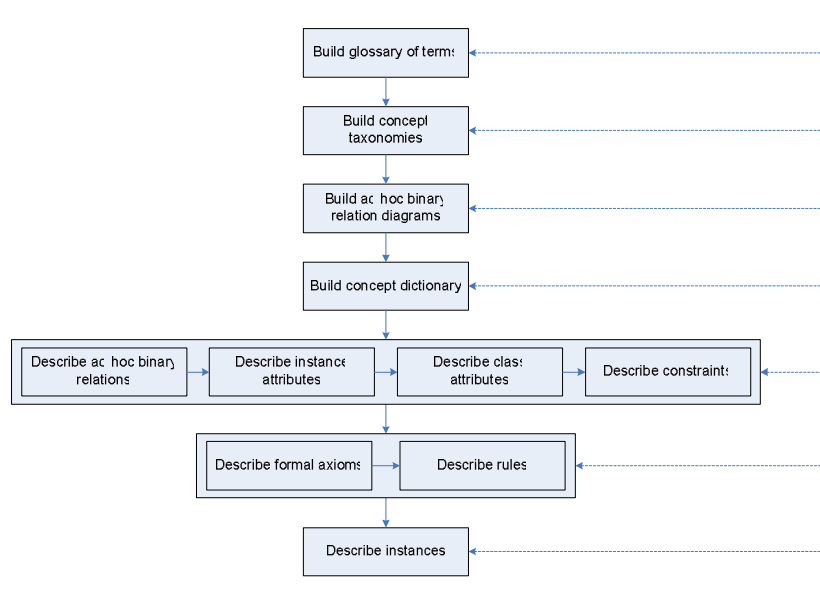


Figure 2-7. METHONTOLOGY Conceptualisation task

On-To-Knowledge Methodology (OTKM) (Sure, Staab 2003, Staab et al. 2001) was developed in the context of a European project with the aim of applying ontologies to electronic information in order to improve the quality of knowledge management in organisations (Gomez-Perez et al. 2004). OTKM consists of five steps, each containing a number of sub-steps as depicted in Figure 2-8. This methodology very loosely takes into account the collaborative aspect of ontology development and the importance of reaching consensus in this process. OTKM supports the underlying concepts of the METHONTOLOGY for knowledge elicitation and representation (Gomez-Perez et al. 2004).

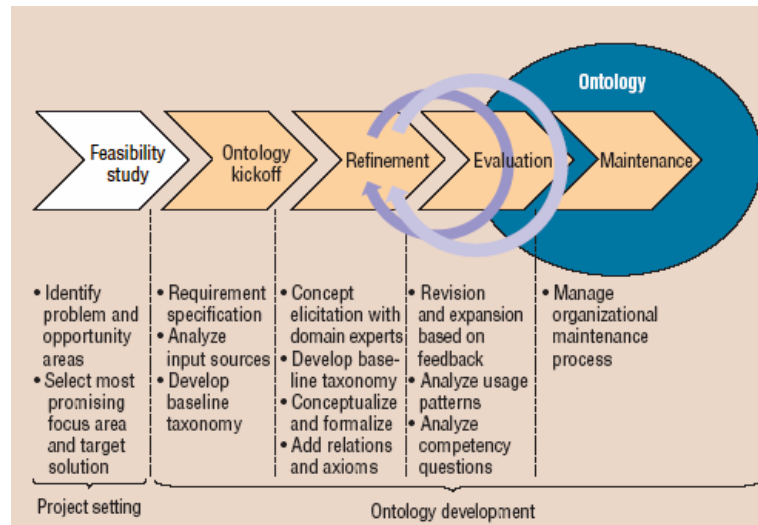


Figure 2-8. OTKM Knowledge metaprocess, adopted from (Staab et al. 2001)

DILIGENT (Distributed, Loosely controlled and Evolving Engineering of ontologies) (Tempich et al. 2006) framework, adopting OTKM, pays more attention to the decentralised development/community based aspects of ontology development, while providing more detailed guidelines for conceptualisation. It is composed of five main activities and their associated actions (sub-activities) as depicted in Figure 2-9.

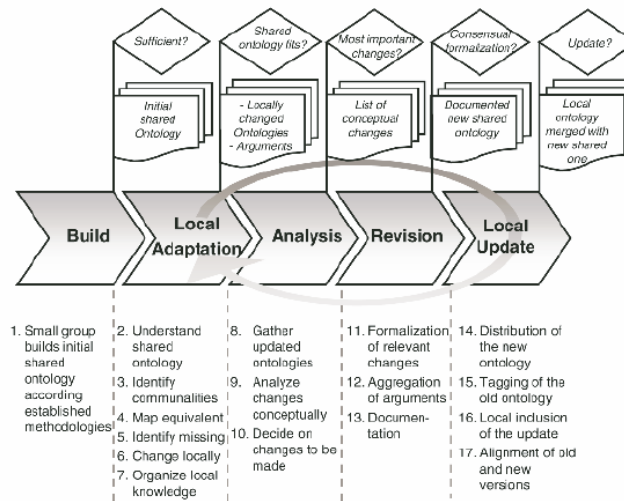


Figure 2-9. DILIGENT methodology process stages (1-5), actions (1-17) and structures, adopted from (Tempich et al. 2006)

DOGMA (Jarrar, Meersman 2002) is a database inspired ontology development methodology, which, similar to DILIGENT, takes into account the community aspect of ontology development and introduces ontology the *double articulation* principle (Jarrar, Meersman 2008). The double articulation decomposes an ontology to two layers: Ontology base and commitment layers, which ensures the ontology to be extended in a gradual means. DOGMA is claimed to contain the best practises of older ontology creation methods, is grounded in natural language facts analysis and is a community-based approach. All these characteristics are significantly important factors for standards development. DOGMA methodology is presented in Figure 2-10.

OntoEng (Ontology Engineering) (Al-Debei, Fitzgerald 2009) is a systematic design method for ontology engineering in information systems, which is inspired by design science research. OntoEng is developed based on the lessons learnt from the existing ontology development methodologies and from the experience of developing the V⁴ Business Model ontology for mobile networks and telecommunication service providers. It consists of five phases comprising twelve design activities as shown in Figure 2-11 (Al-Debei, Fitzgerald 2009).

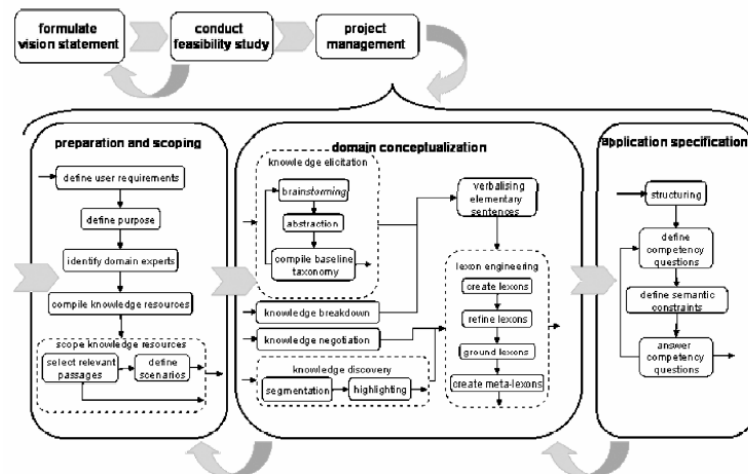


Figure 2-10. DOGMA Methodology

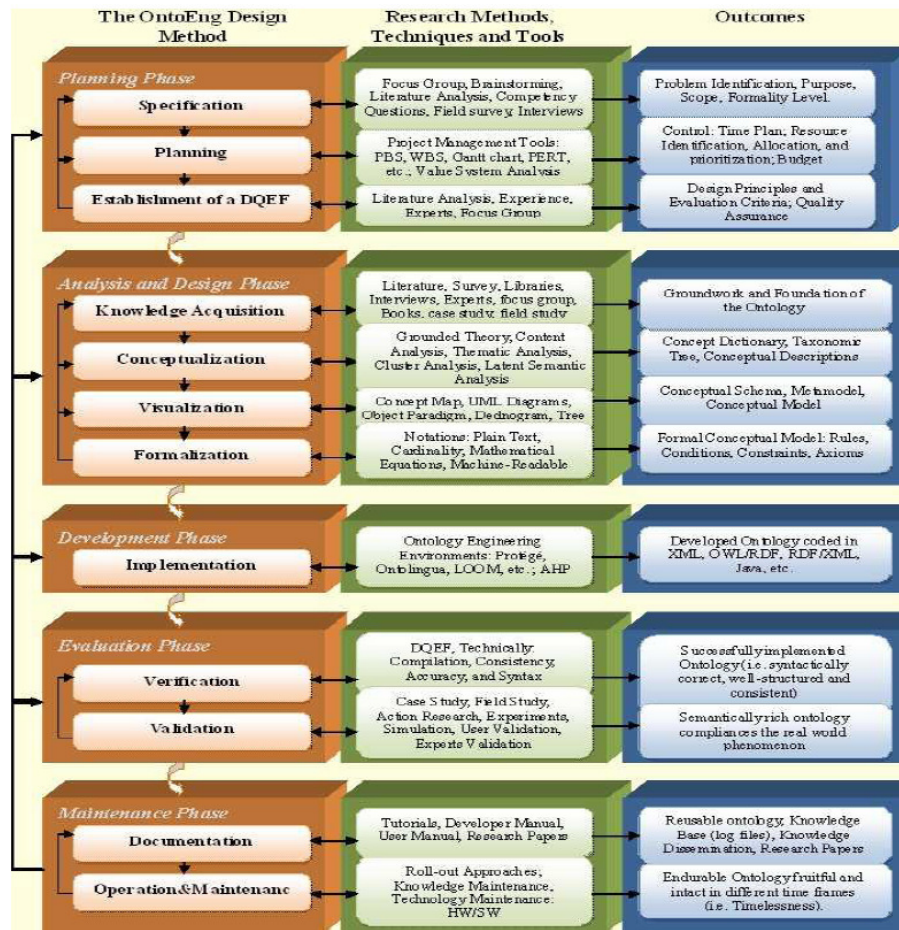


Figure 2-11. OntoEng Methodology

Table 2-5, provides a comparison between the above ontology engineering methodologies. This comparison uses the framework proposed by Gomez-Perez, Fernández-López and Corcho (2004), which itself is based on an earlier work of the same authors (Fernández-López, Gómez-Pérez 2002) and provides various aspects and features for the purpose of comparing ontology engineering methods. The Community aspect is added to this framework as it is an important factor for this research. The various aspects of ontology development could be filled with three types of values: (a) ‘Described’, means the methodology describes briefly how and when each task in the proposed activity should be performed and who should do it and (b) ‘described in details’ is when specific detail are provided for the tasks. (c) ‘Proposed’ means the methodology

identifies the feature but doesn't provide enough guidelines for it to be performed. (d) 'Not proposed' means that it is not mentioned in the guidelines.

Feature		Uschold & King	Grüniger and Fox	METHONTOLOGY	OTKM	DILIGENT	DOGMA	OntoEng	
Ontology Management	Scheduling	NP	NP	P	D	From OTKM	P	P	
	Control	NP	NP	P	D	From OTKM	D	P	
	Quality Assurance	NP	NP	P	D	From OTKM	D	D	
Ontology Development	Pre Development	Environment study	NP	NP	NP	P	From OTKM	P	NP
		Feasibility study	NP	NP	NP	D	From OTKM	P	P
	Development	Specification	P	DD	DD	DD	D	D	D
		Conceptualisation	NP	DD	DD	P	DD	DD	DD, From Uschold & Grunier
		Formalisation	NP	DD	D	D	DD	D	D
		Implementation	P	D	DD	D	D	P	P
	Post Development	Maintenance	NP	NP	P	P	DD	D	P
Use		NP	NP	NP	P	D	P	D	
Ontology Support	Knowledge acquisition	P	P	DD	D	P	DD	DD	
	Evaluation	NP	DD	DD	P	P	P	D	
	Community aspect	NP	NP	NP	P	D	DD	NP	

Table 2-5. A summative comparison on various ontology engineering methods
 P: Proposed, NP: Not Proposed, D: Described, DD: Described in Detail

DOGMA is one of the most complete ontology engineering methods according to Table 2-5. Additionally, DOGMA takes into consideration the community based aspect of development and also pays attention to natural language fact analysis. These led the researcher to choose the DOGMA framework as the most suitable ontology engineering approach to be utilised in this research. More details on DOGMA are provided in Chapter 4.

2.4.2 Quality of Ontologies

Various factors are provided in the literature for the quality of ontologies. Gomez-Perez (2001) identifies three criteria for validating ontologies: *consistency*, *completeness* and *conciseness* as defined in Table 2-6. Based on their work and the work of Gruber (1995), Al-Debei and Fitzgerald (2009) sketch out six criteria for ontology evaluation, which are *clarity*, *coherence*, *conciseness*, *preciseness*, *completeness* and *customisability*.

Consistency	Refers to whether it is possible to obtain contradictory conclusions from valid input definition. A given definition is consistent if and only if the individual definition is consistent and no contradictory knowledge can be inferred from other definition and axioms.
Completeness	<ul style="list-style-type: none"> - All that is supposed to be in the ontology is explicitly defined or can be inferred. - Each definition is complete
Conciseness	<ul style="list-style-type: none"> - Does not have any unnecessary or useless definition - Free of explicit redundancies - No redundancies can be inferred.

Table 2-6. Ontology evaluation factors by (Gomez-Perez 2001)

Wand and Weber (1993) propose that an ontology should be *complete* and *clear* in order to be considered as a faithful representation of its domain. They note that ontological incompleteness, or construct deficit, happens when there exists a real world phenomenon in the domain which does not map to any defined construct in the model (see Figure 2-12).

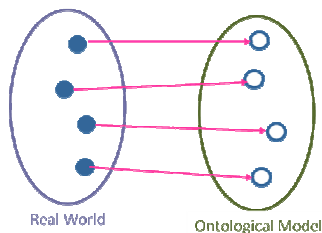


Figure 2-12-a. Ontological completeness

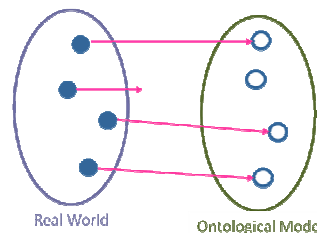


Figure 2-12-b. Ontological incompleteness or construct deficit

Ontological clarity on the other hand is when a modelling grammar is free of (a) *construct overload*, (b) *construct redundancy* and (c) *construct excess* as explained in the following:

- a) *Construct overload* is when there is a construct in the ontological model which can represent more than one real world phenomena as depicted in Figure 2-13.
- b) *Construct redundancy* occurs when for one real world phenomena, there are more than one ontological constructs in the model as shown in Figure 2-14.
- c) *Construct excess* exists when a construct in the model does not map to any real world thing as portrayed in Figure 2-15.

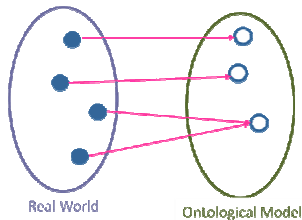


Figure 2-13. Construct overload

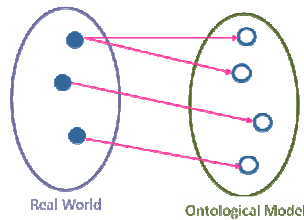


Figure 2-14. Construct redundancy

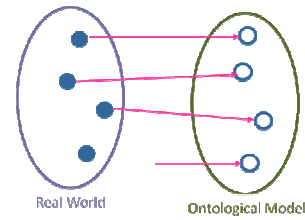


Figure 2-15. Construct excess

These factors are taken into account in Chapter 3 for defining a set of suitable evaluation criteria in this research.

2.5 Ontology for Semantic Interoperability

There is a considerable number of publications in the literature denoting the importance of semantic web technologies in B2B integration and interoperability in the last decade (Cui et al. 2002, Obrst et al. 2002, Obrst et al. 2001, Obrst 2003, Seng, Lin 2007, Vujasinovic et al. 2010, Legner, Wende

2007, Kajan, Stoimenov 2005, Wu et al. 2006, Gong et al. 2006, Höfferer 2007). The following provides a review on the relevant work in this area.

Obrst (Obrst 2003) argues that ontologies offer the richest representation of machine-interpretable semantics, which eventually results in a greater semantic interoperability and integration. He stresses that ontologies support semantic interoperability in various domains, in particular in e-Business and e-Government (Obrst et al. 2002, Obrst 2003). Sánchez, Cavero and Martínez (2007) further recognise semantic integration and domain modelling as two most important applications of ontologies. Likewise, Rebstock, Fengel and Paulheim (2008) identify ontology as an appropriate means for achieving semantic interoperability.

Based on Berners-Lee's Semantic Web vision (Berners-Lee et al. 2001), Singh, Iyer and Salam (2005) provide a vision for semantic e-Business. They introduce semantic e-Business as an approach to managing knowledge for the coordination of e-Business processes through the systematic application of semantic web technologies (Singh et al. 2005). They further argue that semantic e-Business will be enhanced through more rigorous information and knowledge exchange and that ontologies can capture the definitions and interrelationships of concepts in a variety of domains resulting in a shared understanding of the domain.

Daga et al. (2005) argue that ontology offers an appropriate basis for providing a clear description of the objects in a domain, which lead to improved semantics, better interoperability and less complexity. They propose an ontological approach, called Content Sophistication, for extracting and documenting the business knowledge hidden within business data in legacy information systems (Daga et al. 2005). Utilising Daga et al.'s (2005) approach, Aldin, De Cesare and Lycett (2009) provide a framework for the semantic

discovery and reuse of business process patterns. Their framework defines a dual lifecycle model with the aim of (a) deriving business process patterns from legacy content through the use of ontologies and (b) business modelling, reusing the patterns defined in the initial lifecycle.

Emphasising on the importance of ontologies for business process integration, Wu et al. (2006) provide an e-Business process modelling framework that outlines the required building blocks for enabling e-business process automation. Their framework however, does not provide a comprehensive ontology for B2B process interoperability. Seeing the inter-organisational business process collaborations as significant factors for business development, Gong et al. (2006) also recognise semantic web technologies as the most promising direction for integration and collaboration. They provide a semantic agent based approach for achieving inter-organisational process interoperability. Similar to Wu et al. (2006), their ontology does not provide a comprehensive solution for interoperability of business process standards and they themselves consider their work as a foundation for further research in the area.

In addition to the above, there is a multitude of papers discussing ontology based integration of XML data sources. For example Cruz , Xiao and Hsu (2004) (and also Xiao and Cruz in another paper (Xiao, Cruz 2006)) propose an ontology mediated architecture for integrating heterogeneous XML sources. Other examples are (Vujasinovic et al. 2010, Xiao, Cruz 2006, Huma et al. 2005, Cruz, Nicolle 2006, Dong, Linpeng 2008), a number of which are discussed in the next section since they are addressing standards based interoperability .

The above publications show the importance of ontologies and semantic web technologies in inter-organisational interoperability. They further manifest that

ontologies are considered as an appropriate approach for formalising domain knowledge in a clear and unambiguous manner, which eventually facilitates interoperability.

2.6 Ontologies and Standards in the Literature

Using ontologies as a means of formalising standards structure has gained momentum in the last few years. There are a growing number of ontologies developed for various standards and specifications. Examples are oXPDL, an ontology for XPDL (Haller et al. 2008), an ontology for WS-BPEL (Nitzsche et al. 2007), ebXML Registry Profile for OWL (OASIS ebXML Registry Technical Committee 2006), ebXML ebBP (Business Process Specification Schema) (Heravi et al. 2010a) and OntologUBL, which provides an ontology for Universal Business Language(The Ontolog Forum 2003).

García and Gil (2007) provide a solution for automatic transformation of XML Schemas and XML documents to OWL and RDF respectively. However, the examination of the ontologies extracted using this approach (specifically the ebBP ontology), revealed that the approach does not cover both the semantics and the syntax of standards models. For example, none of the data properties in the ontology have domain and range, none of the Object Properties have a domain and most of the Object Properties do not have ranges. The data types that exist in OWL such as *int*, *string* and *IDREF*, are ignored in this ontology, for each data type a class is defined. This is a result of automatic translation, without paying attention to the semantics of the entities.

There are also a few works which focus on utilising ontologies in conjunction with standards. Anicic, Ivezic and Jones (2006) propose a methodology for Semantic Enterprise Application Integration Standards, which utilises semantic web technologies for achieving interoperability between two business document

standards - STAR and AIAG - both of which are based on the OAGIS standard. They have used an automated tool for transforming existing XML schemas and instances into OWL ontologies. They have further created a merged ontology and then translate the instance data from one standard to another using their mapping technique and merged ontology. Their methodology requires developing ontologies for each standard in the first place, which is done using automated tools in this project. No implementation of their automated tool was available at the time of this research to be able to test the richness of the ontological models created using the automated transformation.

Liegl, Huemer and Zapletal (2009) provide a methodology for building a global reference business documents ontology based on the UN/CEFACT (the United Nations Centre for Trade Facilitation and Electronic Business) Core Components Specification (CCTS). Vujasinovic et al. (2010) provide a semantic mediation architecture for standard based B2B interoperability. This work emphasises on the importance of standard development organisations in achieving standard based semantic B2B integration and thus highlights the importance of ontologies in relation with e-Business standards.

Conrad, Newman and Smith (2004) provide a case for the ontological expression of e-Business standards and the way ontologies may improve such standards. They present a set of potential benefits from using ontologies in the process of standards development and in particular for conceptual modelling. They suggest using upper ontologies for standards ontological development, but do not provide a methodological approach - neither for developing ontologies for existing standards nor for using ontologies in the process of standards development and conceptual modelling.

Grenon and De Francisco (2009) claim that ontology-strength industry standards facilitate knowledge representation and sharing and present an

ontologisation of a set of telecommunication and clinical trial standards (Grenon, De Francisco 2009, Grenon et al. 2011). They support the view that producing ontologies for standards has the potential of furthering and enhancing standards' development, dissemination, and operationalisation and postulate that ontologisation of standards should be part of the standards development life cycle. They however, do not provide a methodological approach for such convergence.

OASIS may be considered as the first standard development organisation to address ontology technologies and their synergy with standards. The Semantic Support for Electronic Business Document Interoperability Technical Committee (SET TC) (OASIS SET TC 2009) is considered to be the first ontology related initiative in OASIS. SET TC aims at developing specifications for machine process-able semantic content of the electronic business documents, and in particular UBL (OASIS UBL TC 2008), which itself is based on the UN/CEFACT Core Components Technical Specification (CCTS). Another relevant TC in OASIS, which may be considered as the first official ontology oriented standard Technical Committee, is called OASIS Quantities and Units of Measure Ontology Standard (QUOMOS) Technical Committee (OASIS QUOMOS TC 2010), which aims at developing an ontology to specify the basic concepts and systems of quantities, measurement units and scales, metric prefixes, rules for constructing various derived units, and designations of the most common derived units to be used across multiple industries. Ontolog forum (The Ontolog Community 2010) is another relevant initiative which addressed the importance of ontologies for standard community in their 2009 summit theme: "Toward Ontology-based Standards". In fact OASIS QUOMOS TC was a consequence of discussions in the Ontolog forum.

The above efforts show the significance of ontologies in the standards world and further the importance of both ontology and standards for achieving

semantic interoperability, which imply that it is time for the intersection of these two communities. Nevertheless, almost no effort has yet been expended on utilising ontologies for developing, authoring or improving standards. With the current trend in utilisation of ontologies in conjunction with standards efforts, it is believed that using ontologies in the process of standards development is an important area of research for the future of standards development which deserves to be explored extensively.

2.7 Literature Findings and Research Direction

In this chapter the literature on three facets of ontology-based standards development was studied and analysed, separately and in combination. These facets are: (1) Semantic interoperability, (2) standards and their development process and (3) ontology engineering, which are considered to be the backbone of the design methodology developed in this research.

To utilise ontologies in the standardisation process and standards development, there is a need for re-engineering existing standards' development methodologies. Therefore, the first step in this journey was to study the existing standards development processes used in various standardisation bodies. This study led the researcher to understand not only that standards lack in supporting semantic interoperability, but also their development approaches considerably lack rigour and that most standards are currently developed in a semi ad-hoc manner. To address this, a more rigorous approach for standards development needs to be designed, which could cater for most standards development organisations in various domains.

In addition to the above, the literature review reveals that there has been an increasing interest in developing ontological representations of various standards and that it is commonly believed that ontologies would greatly

facilitate knowledge management and also interoperability. However, no methodological approach exists in the literature addressing either: (a) Developing ontologies for existing standards; or (b) an approach which could be utilised in the process of standards development. Likewise, there aren't enough guidelines in the standards development literature for developing robust conceptual models of standards or even for drafting the standards specification, which is considered to be the most important part of any standardisation activity.

In contrast to the lack of existence of well defined standards development methodologies, the literature reveals that there are a considerable number of very well defined ontology engineering and development methodologies. Therefore, it is considered reasonable to use the best practices of ontology engineering in conjunction with existing, loosely defined, standards development methodologies. In addition to using the best practices of ontology engineering for designing a novel standards development methodology, it is proposed that the standards themselves should also be based on ontologies, aiming at higher level of formality and clarity and also semantic interoperability.

Taking the above into consideration, this research aims at addressing the following gaps, which arise from reviewing the relevant literature:

- (a) Lack of robust guidelines for defining conceptual models of standards;
- (b) Lack of a systematic design methodology for standards development;
- (c) Lack of methodological guidelines for ontologising existing standards;
- (d) Lack of a coherent methodological approach for designing ontology-based standards and utilising ontologies throughout the standards development life cycle.

This research aims at addressing the above gaps by designing a systematic standards development methodology, which is based on ontologies, while taking into account the best practices of ontology engineering and standards development. This would cater for a more robust and cohesive standards based semantic interoperability.

CHAPTER THREE: Research Method

3.1 Introduction

Research methods are necessary for conducting systematic research in any discipline. There exist various research methods in the Information Systems (IS) discipline which differ in aspects such as their research process, techniques and philosophical underpinning. This chapter investigates and presents the research methodology adopted in this thesis for designing an ontology-based information standards development methodology – Design Science Research (DSR). DSR's fundamental underpinning is that improving practice and solving problems is essential, as the utility of the solutions is important. While discussing DSR as an accepted, valid and legitimate IS research method, this chapter demonstrates the justification behind choosing this framework for executing this research.

This chapter is structured as follows. Section 3.2 provides an overview on existing research paradigms in information systems research and an introduction to DSR itself. Section 3.3 investigates the design science research method in more detail and presents its phases and outputs as well as its evaluation methods. Section 3.4 presents the research design of this thesis and its iterations to provide the reader with a holistic picture of the research. Section 3.5 summarises the chapter.

3.2 Research Paradigms in Information Systems

Research in information systems is broadly characterised by two paradigms: natural science (aka. behavioural science) and design science. Natural science mainly attempts to understand reality, explain or predict human or organisational behaviour and produces general theoretical knowledge, whereas design science is concerned with designing novel and innovative artefacts to serve human and organisational purposes (March, Smith 1995, Hevner et al. 2004). These two approaches are not mutually exclusive – the findings of natural science may be used in design science to improve the quality of designed artefacts so that they would serve human purposes the best.

Hevner et al. (2004) propose a conceptual framework for IS research combining behavioural science and design science paradigms as depicted in Figure 3-1: They state that behavioural science research is mainly concerned with the development and justification of theories, aiming at finding ‘truth’, while design science research is materialised through building and evaluation of artefacts aiming at ‘utility’. Hevner et al. (2004) further argue that truth and utility are inseparable, meaning that an artefact’s utility may be due to some undiscovered truth. Likewise a theory may contribute to design of new and useful artefacts.

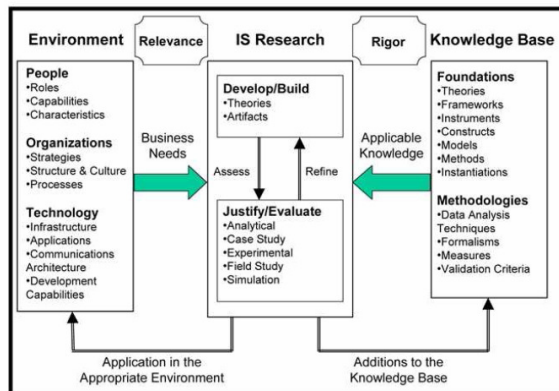


Figure 3-1. Information Systems Research Framework (Source: Hevner et al. 2004)

3.2.1 Design Science Research in IS

Design research has its roots in engineering and the sciences of the artificial (Simon 1996). However, the importance of design is well recognised in the IS literature, and design science has received increasing attention from IS scientists over the last decade (March, Smith 1995, Peffers et al. 2007, Vaishnavi, Kuechler 2007, Hevner et al. 2004, Glass 1999, Markus et al. 2002, Nunamaker et al. 1991). Design science research is a problem solving research paradigm, which is aimed at designing artefacts as a solution to research problems. A Design science artefact is typically of the form of a construct, model, method or an instantiation, as described in table 3-1.

Output	Description
Constructs	The conceptual vocabulary of the domain which provide the language in which problems and solutions are defined and communicated.
Models	A set of propositions or statements expressing relationships between constructs. In other words models use constructs to represent a real world situation, the design problem, and its solution space.
Methods	A set of steps and guidelines on how to perform tasks and solve problems.
Instantiations	The realisation of constructs, models and methods.

Table 3-1. The Outputs of Design Research (Vaishnavi, Kuechler 2007, Hevner et al. 2004)

There is a debate in the literature about ‘theory’ and whether it should be considered as an artefact or an output of design research in IS at all. Few publications in the literature consider ‘theory’ as a design research artefact (Vaishnavi, Kuechler 2007), while some consider it as a design research output, but not an artefact (Hevner et al. 2004, Markus et al. 2002, Nunamaker et al. 1991, Puroo 2002, Walls et al. 1992). The reason for the latter classification is the term artefact itself which is used to describe something that is artificial, or constructed by humans, as opposed to something that occurs naturally (Simon 1996). A contrasting view also exists (March, Smith 1995), which considers

‘theory’ neither an artefact of design science nor an output of that. They believe that ‘theory’ is a result of natural science research and not design research.

Walls et al. (1992) have a compelling view with regards to theories and describe a design theory as a prescriptive theory based on theoretical underpinnings and argue that since design theories are prescriptive, they are different from natural science theories, which are explanatory and predictive. They argue that another difference between natural science and design theories is in the way they treat a goal: Goals in natural science theories are meaningless and social science theories may deal with goals as objects of study, while goal in design science theories is a prerequisite. The objective of design theories are, however, not to achieve the goal, but to “explain why specific goals exist or predict outcomes associated with goals” (Walls et al. 1992 pp. 40). Walls et al. (1992) further discuss that a design theory should address two types: (1) the ultimate *design product* and (2) the *design process* which is used for deriving the design product. They eventually (Walls et al. 1992) introduce *kernel theories*, which are originated from natural and social sciences and govern the design requirements and process.

Supporting the view point of Walls et al. (1992), Venable (2006b) strongly argues that theories should be a primary output of design research and play a central role in the process of design research. He (2006b) proposes a framework for design science research which signifies the importance of theories and theorising in design science research in IS as shown in Figure 3-2.

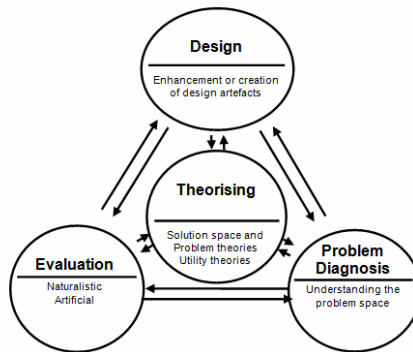


Figure 3-2: An Activity Framework for Design Science Research (Venable 2006b, Venable 2006a)

Based on Walls et al.'s (1992) definition of *kernel theory*, Venable (2006b) introduced a new type of theory called *utility theory*. He argues that utility forms a key focus for the outcomes of design research and that a utility theory makes an assertion that a particular type or class of technology has some level of utility in solving or improving a problematic situation (Venable 2006b). Consequently Venable (2006b) argues that, the meaning of a utility theory should be specified in terms of its impact for solving or improving the problem(s) within the problem space. In other words a utility theory makes an assertion that a particular type of solution (from solution space) has some level of utility for solving or improving the defined/observed problem(s) (from problem space). Figure 3-3 presents the utility theory and its relation to problem and solution space.

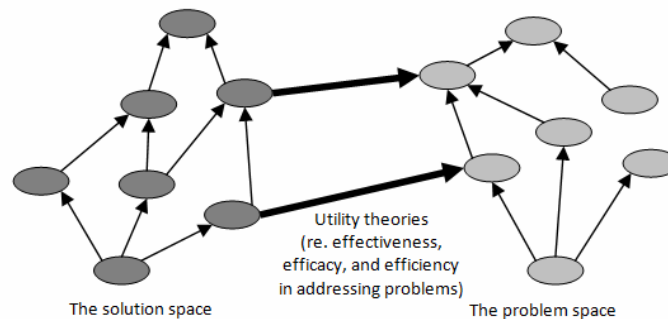


Figure 3-3: Components of utility theories and hypotheses (Venable 2006b)

This research follows the definition of Venable (2006b) and considers ‘theory’ as a requirement in the process of design and as a design research output in parallel with artefact.

Regardless of the type of the artefact, Hevner et al. (2004) point out that in most cases the artefacts constructed in design science research are not mature enough to be used in practice. However, they note that the design artefacts are “innovations, that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation and use of information systems can be effectively and efficiently accomplished” (Hevner et al. 2004 pp. 83).

3.3 Design Science Research Methodology

A number of publications exist in the literature, which provide various methodologies or guidelines for conducting design science research in IS (March, Smith 1995, Peffers et al. 2007, Vaishnavi, Kuechler 2007). Although existence of these publications show that there is not a single well accepted methodology for conducting design science research in IS, there are enough commonalities between these publications to enable researchers to derive a methodology that will suit them for conducting their own design research. Table 3-2 summarises the three most cited DSR methodologies.

(March, Smith 1995)	(Vaishnavi, Kuechler 2007)	DSRM (Peffers et al. 2007)
	Awareness of Problem	Problem identification and motivation
	Suggestion	Define the objectives for a solution
Build	Development	Design and development
		Demonstration
Evaluate	Evaluation	Evaluation
	Conclusion	Communication

Table 3-2. Comparison of activities/phases in various Design Research Methodologies

Studying and analysing various DRS methodologies, this work formulates and adopts a research method, which is seen as the most suitable for the purpose of this research as depicted in Figure 3-6.

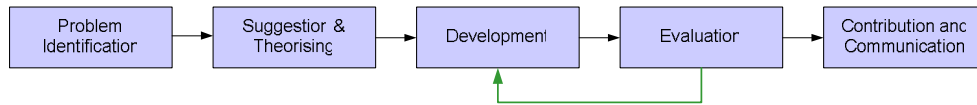


Figure 3-4. Adopted Design Science Methodology

Details of these phases, their reason behind their selection and their intended meaning in this thesis are as follows:

3.3.1 Problem Identification

As discussed earlier, a ‘problem’ is the very first requirement of any design research. In two of the three aforementioned methodologies (Peffer et al. 2007, Vaishnavi, Kuechler 2007), problem identification/awareness is explicitly defined as the first activity in the design science research. Furthermore, March and Smith (1995) have implicitly mentioned ‘problem’ as the prerequisite of the design science research process. Other publications, e.g. (Hevner et al. 2004, Nunamaker et al. 1991, Walls et al. 1992), also agree on the necessity of a problem in the design science research. Therefore defining a relevant problem is considered as an important activity in design science research and is included in a design research methodology used in this thesis.

The objective of design science research is to obtain knowledge in order to solve an important and relevant problem, which should be defined in this phase. It is important to note that problems are perceived and therefore each stakeholder may have a different view on a similar class of problems. Thus a solution to a problem from a view point of the perceiver of the problem would be a desirable one as opposed to the ultimate good or best solution (Walls et al.

1992). The output of this phase is a formal or informal proposal for a new research effort (Vaishnavi, Kuechler 2007).

3.3.2 Suggestion and Theorising

The methodologies of Vishnavi (2007) and Peffers et al. (2007) propose ‘suggestion’ and ‘define the objectives for a solution’ as their second activity respectively. These two activities are considered to be similar activities as the aim of both is to define what is going to be done as the result of the research. Moreover, Walls et al. (1992) state that theorising is a central activity in design research and relate very closely to the problem identification and the solution suggested for that problem. As discussed earlier, this research considers theories, and in particular utility theories, as a requirement for the process of design. Thus, ‘Suggestion and Theorising’ are considered to be an important part of the design science research and therefore included in the design research framework in this research.

In this phase the type of solutions which may suit the defined problem is provided. During this phase researchers draw the objectives of a solution from the problem definition and knowledge of what is possible and feasible (Peffers et al. 2007). The output of this phase is a Tentative/meta design, and its underlying utility theory, as a set of suggestions believed to provide appropriate solutions to the problem defined in the previous phase.

3.3.3 Development

As it is evident from table 3-2, ‘development’/’build’ and ‘evaluation’ are highly accepted activities in the three methodologies. In addition to the DSR methodologies presented in Table 3-2, these two activities are regarded as important activities in design research in some other publications, e.g. (Hevner

et al. 2004, Nunamaker, Chen et al. 1991). Therefore these two activities are considered as integral parts of the design science research and are included in the design research methodology adopted in this thesis.

The artefact, which is the solution to the stated problem and its defined objectives, is built in the 'development' phase. The output of this phase is an artefact or a set of artefacts (Vaishnavi, Kuechler 2007). The techniques for developing the artefact vary depending on the type of the artefact to be constructed. As stated by Vaishnavi (2007), the implementation of the artefact itself does not need to involve novelty beyond the state-of-practice for the given artefact; the novelty is primarily in the design, not the construction of the artefact.

3.3.4 Evaluation

As noted above evaluation is a well accepted activity in DSR and is therefore included in the research method in this thesis. Furthermore 'demonstration' is an activity which only appears in DSRM methodology (Peppers et al. 2007) and aims at demonstrating the use of the designed artefact to solve one or more instances of the defined problem. In this thesis 'demonstration' is considered to be an optional part of the 'evaluation' phase or one way of doing so.

Evaluation is the key phase in demonstrating the utility, relevance and efficacy of design artefacts, as noted by several authors (March, Smith 1995, Hevner et al. 2004, Walls et al. 1992). This phase observes and measures how well the artefact contributes to a solution for the problem (Peppers et al. 2007). The design artefact(s) need to be evaluated according to a set of defined criteria and therefore developing these criteria is a part of the evaluation in design research.

Although evaluation is repeatedly mentioned as a significantly important phase in design research, there is little in the literature that provides clear guidance on

the choice of strategies and methods for evaluation in design research. Extant work is of three kinds in this area as follows:

- *The type of work which provide general frameworks for categorising the nature of evaluation and/or choosing among them.* One such framework is Venable’s (2006a) classification of the design research evaluation approaches into two main categories: (a-1) artificial and (a-2) naturalistic. Artificial evaluation of a design artefact is “evaluating a solution technology in a contrived, non-real way” (Venable 2006a pp. 5). Naturalistic evaluation on the other hand evaluates the performance of a solution in its real environment: real users using real solution to solve a real problem (Pries-Heje et al. 2008). Another dimension for categorising design evaluation is the time at which evaluation takes place: (a-3) *ex-ante* and (a-4) *ex-post*. The *ex-ante* evaluation takes place before the artefact is created while *ex-post* evaluation is after the construction of the artefact (Pries-Heje et al. 2008). Pries-Heje, Baskerville and Venable (2008) provide an evaluation framework for design science which takes into account the above factors as depicted in Figure 3-7.

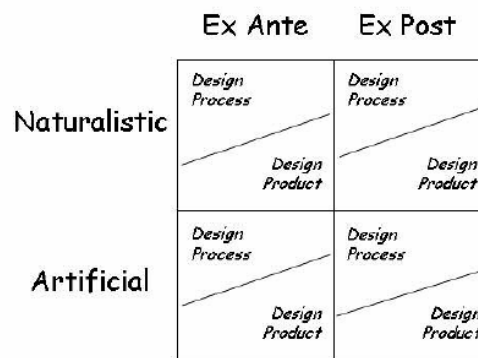


Figure 3-5. Strategic DSR evaluation framework (Pries-Heje et al. 2008)

- *A kind of work which provide a general set of evaluation methods, for example observational, analytical, experimental, testing and descriptive as suggested by Hevner et al. (2004) and depicted in Table 3-3.*

Evaluation Method	Description
1. Observational	Case study: Study artefact in depth in business environment.
	Field study: Monitor use of artefact in multiple projects.
2. Analytical	Static analysis: Examine structure of artefact for static qualities (e.g. complexity).
	Architecture analysis: Study fit of artefact into technical IS architecture.
	Optimisation: Demonstrate inherent optimal properties of artefact or provide optimality bounds on artefact behaviour.
	Dynamic analysis: Study artefact in use for dynamic qualities (e.g. performance).
3. Experimental	Controlled experiment: Study artefact in controlled environment for qualities (e.g. usability).
	Simulation: Execute artefact with artificial data.
4. Testing	Functional testing: Execute artefact interfaces to discover failures and identify defects.
	Structural testing: Perform coverage testing of some metric (e.g. execution paths) in the artefact implementation.
5. Descriptive	Informed argument: Use information from knowledge base to build a convincing argument for the artefact's utility.
	Scenarios: Construct detailed scenarios around the artefact to demonstrate its utility.

Table 3-3. Design science research evaluation methods, adopted from (Hevner et al. 2004)

- *The work which provide a set of evaluation criteria for various types of artefacts, for example March and Smith's criteria (1995) as depicted in Table 3-4.*

For a better understanding of the DRS evaluation in IS, a study on a number of well cited DSR publications was conducted to investigate the ways in which they have approached evaluation, with special attention paid to the evaluation of methods. A summary of this study is provided in Table 3-5.

Metrics \ Artefacts	Artefacts	Constructs	Models	Methods	Instantiations
Completeness		✓	✓	✓	
Simplicity		✓			
Elegance		✓			
Understand-ability		✓			
Ease of use		✓		✓	
Validity		✓			
Fidelity with real world phenomena			✓		
Level of detail			✓		
Robustness			✓		
Consistency			✓	✓	
Operationality				✓	
Efficiency				✓	✓
Generality				✓	
Effectiveness				✓	✓
Quality of result				✓	

Table 3-4. Evaluation criteria according to March and Smith (1995)

	Publication	Evaluation
1	Developing DSRM: Design Science Research Method (Peffer et al. 2007)	Peffer et al. (2007) demonstrate DSRM by applying it to four already published IS research projects. DSRM is evaluated for three loosely defined objectives as follows by providing some narrative justification: (1) Should be consistent with prior DS research theory and practice; (2) should provide a nominal process for conducting DS research in IS; (3) Should provide a mental model for characteristics of research output.
2	The CATCH Data Warehouse for Health Status Assessments (Berndt et al. 2003, Berndt et al. 2001)	They evaluate the research by claiming that the CATCH method had been used and refined in more than 20 US counties for more than 10 years, which shows that evaluation of an IS design can take quite some time.
3	A Software Reuse Measure Developed (Rothenberger, Hershauer 1999)	The research is evaluated using a case study, in which the developed measure was used to assess the reuse rates of five projects at MBA Technologies using structured interviews.
4	SIP-Based Voice- and Video-Over IP Software (Chatterjee et al. 2005, Gemmill et al. 2004)	This project is demonstrated through implementation of the software, which was evaluated via a test process consisting of debugging and then sharing the software with stakeholders using a web portal in the way that the software would automatically report the use time by downloaded parties. Furthermore the

		developed software client is stated to be successfully tested for performance, usability and usefulness. It is also mentioned that the performance of the software was satisfactory, without providing any metrics for the satisfaction and its level. They also claimed that “The Internet2 working group was pleased with the efforts, judging the design process successful”.
5	Developing a Method at Digia to Generate Ideas for New Applications that Customers Value (Peffer, Gengler et al. 2003, Peffer, Tuunanen 2005)	The methodology designed in this research project is demonstrated for its feasibility and efficacy. This is done via structured interviews and qualitative clustering followed by a workshop with six experts. In the evaluation activity it is stated that the developed method met the project’s objectives. It is also stated that the stakeholders were enthusiastic about the results of the workshop and that their feedback and the successful implementation of the method enabled the researchers to present initial “proof-of-concept”-level validation of the new method.
6	The Design and Implementation of Anonymity in GDSS (Gavish, Gerdes Jr. 1998)	The evaluation of this project consists of two activities. First each mechanism is proved to correctly provide the claimed anonymity in Group Decision Support Systems - Formal proof methods are used to validate the effectiveness of the designed mechanisms. It is then followed by a cost benefit analysis.
7	A Workflow Language for Inter-organisational Processes (XRL) (Van der Aalst, Kumar 2003)	This project is evaluated in the following ways: 1- Compared with similar existing languages and systems. 2- The fit of XRL is studied in the field and relevant standards body and it is explained that relevant specifications do not include such a language and why they need to do so. 3- A research prototype has been implemented for the proposed language. 4- It is stated that the fact that the XRL language is developed on Petri-net foundation, makes it possible to claim for it to be verified for correctness and performance.
8	Information Systems Design for Emergent Knowledge Processes (Markus et al. 2002)	In evaluating this project it is stated that no formal evaluation was attempted in the sense of comparison with other artefacts. Hevner et al. (2004) point out that “this is not surprising, nor is it a criticism of this work. There simply are no existing artefacts that address the same problem”. They however, call for a future research, as a separate project, for conducting a comparative

		assessment for effectiveness of other possible approaches in the relevant contexts, while reiterating that this is not a criticism of the work. While the researchers of the project claim that “the evidence suggests that TOP Modeler was successful in supporting organisational design”, they point out that “only the accumulated weight of empirical evidence will establish the validity” of the claims made in the project (Hevner et al. 2004).
--	--	--

Table 3-5. How different design research are evaluated in the literature

Amongst all in Table 3-5, perhaps the most interesting evaluation method is evaluation of Peffers et al.’s (2007) DSRM methodology itself (1), which is demonstrated by applying DSRM to four already published IS research projects as case studies (2-5) as presented in Table 3-5 and evaluated against three objectives of the project. These defined objectives are: (a) should be consistent with prior design science research theory and practice, (b) should provide a nominal process for conducting design science research in IS and (c) should provide a mental model for characteristics of research output. In addition to the above, four other projects (6-8) are included in Table 3-5 which are presented and evaluated by Hevner et al. (2004) to demonstrate the application of their well cited design science research guidelines.

The above evaluation categories provide an overview of the type of evaluation methods and criteria that may be used to evaluate an IS design artefact. None of them however, provide clear guidance on the choice of strategies and methods for design science evaluation. This is evident from the various, and in most cases very subjective, forms of evaluation taken in various design research publications as clear in Table 3-5.

As it is clear from the above analysis, no evaluation method exists which fits all sorts of DSR artefacts and therefore a set of evaluation criteria should be designed for each specific project, based on its requirements. Thus, it is

important to be clear about the type of artefacts in each project and the way they need to be evaluated.

Evaluation in this thesis

The ultimate design artefact in this thesis is the OntoStanD methodology, which is a method in DSR terms. As presented in Table 3-4, the design research literature provides a set of criteria for evaluating methods: completeness, ease of use, consistency, operationality, efficiency, generality, effectiveness and quality of result. However, as it is evident from Table 3-5, evaluating methods in design science have been undertaken in an ad-hoc manner and the above factors are not considered to be easy and practical to measure against. Therefore, in conjunction with the discussed factors for evaluating design methods, a more theoretical position is taken for evaluating methods in this thesis, which considers the method to provide a form of theory - accepting, as such, that it is fallible. This research takes the view that it is impossible to evaluate objectively but, steering clear of a naïve relativism, that some approaches will relate better to the structure and practice of the world – it is the structure and practice of application that is the arbiter.

Consequently, a pragmatic view is taken in this research, based around the notion of practical adequacy of methods (Sayer 1992). This notion proposes that knowledge should be judged for its usefulness rather than being true or false and must generate expectations of the world that are: (a) actually realised; and (b) intersubjectively intelligible and acceptable.

In this thesis, the designed method (OntoStanD) is evaluated for its *utility* and *practical adequacy*, which are measured by: (a) Determining if the method meets its intended purpose which is primarily providing methodological approach for achieving a high quality and machine process-able conceptual

model of a standard; (b) Ability to cover normative statements of a standard in an ontological manner and on top of the defined ontological conceptual model.

The output of the application of OntoStanD is a set of ontological models, which are of type models in DSR terms, and a standards specification, which itself is a model consisting of a set of constructs. Considering the fact that the ultimate goal of OntoStanD is achieving higher quality standards, it is presumed here that if the ontological models of a standard, resulting from the application of OntoStanD are of high quality, it could be inferred that the method used for modelling is a practical and appropriate one. Therefore evaluating the resulting ontological models plays an important role in evaluating the utility and practical adequacy of OntoStanD. To evaluate the ontological models of standards, a set of suitable evaluation factors are required which are discussed in the following.

An important step in OntoStanD is developing a conceptual model for a standard, which needs to be a faithful representation of the standard and its domain. A conceptual model of a domain is considered to be a faithful representation if it is *accurate, complete and consistent* (Shanks et al. 2003). Since this thesis suggests ontologies should be used for defining conceptual models of standards, the quality factors for evaluating ontological models are also required to be taken into account. Thus, in addition to DSR evaluation factors for models, OntoStanD incorporates and builds upon the following research domains: 1) Standards development, 2) conceptual modelling and 3) ontology engineering. Table 3-6 summarises the quality factors found in the literature for the four aforementioned facets of OntoStanD (i.e. design research, standards development, conceptual modelling and ontology engineering, the two latter were discussed in Chapter 2).

Evaluation criterion	Publication	Standard	Conceptual Model	Ontology	Construct	Model
Consistent	(Al-Debei, Fitzgerald 2009, Gomez-Perez 2001, Shanks et al. 2003).	✓	✓	✓		✓
Complete	(March, Smith 1995, Al-Debei, Fitzgerald 2009, Gomez-Perez 2001, Wand, Weber 1993, Shanks et al. 2003).		✓	✓	✓	✓
Concise	(Al-Debei, Fitzgerald 2009, Gomez-Perez 2001)			✓		
Precise	(Al-Debei, Fitzgerald 2009)			✓		
Clear	(Al-Debei, Fitzgerald 2009, Wand, Weber 1993)			✓		
Efficient	(Folmer, Putner 2011)	✓				
Reliable	(Folmer, Putner 2011)	✓				
Usable	(Folmer, Putner 2011)	✓			✓	
Portable	(Folmer, Putner 2011) (Similar to customisable)	✓				
Maintainable	(Folmer, Putner 2011, Al-Debei, Fitzgerald 2009)	✓		✓		
Adoptable	(Folmer, Putner 2011)	✓				
Open	(Folmer, Putner 2011)	✓				
Accurate	(Shanks et al. 2003).	✓	✓			
Simple	(March, Smith 1995)				✓	
Elegant	(March, Smith 1995)				✓	
Understandable	(March, Smith 1995, Folmer, Putner 2011)	✓			✓	
Ease of use	(March, Smith 1995, Folmer, Putner 2011)	✓			✓	
Valid	(March, Smith 1995)				✓	
Fidelity (faithful)	(March, Smith 1995)				✓	
Detailed	(March, Smith 1995)					✓
Robust	(March, Smith 1995) (mature, accurate and reliable)	✓				✓

Table 3-6. Summary of relevant evaluation factors for evaluating ontological conceptual models

Based on the above analysis and believing that a high quality ontology-based standard has to represent that it has a high quality ontological conceptual

model, OntoStanD employs the following criteria for evaluating the ontological conceptual models of standards:

1- Consistency

Consistency refers to the possibility of obtaining contradictory explicit or implicit (by inference) conclusions from valid input definition (Gomez-Perez 2001, Shanks et al. 2003). A standard needs to be consistent and its consistency depends on the consistency of its conceptual model which would be improved if the standards are defined in an ontological manner since any syntactic or semantic inconsistency is to be discovered much more easily. Thus an ontological representation of the conceptual model of a standard has to be checked for its internal consistency. This could be done with any ontology development tool such as Protégé; If no inconsistency is reported by the reasoner, it could be deduced that the ontological model of a standard is consistent.

2- Completeness

Completeness in this thesis follows Wand & Weber (1993) definition for ontological completeness and clarity, which states there should be an explicit or implicit (via inference) one to one mapping between the design constructs and the real world phenomena in the domain. Gomez-Perez (2001) also believe an ontology is incomplete if one or more central phenomena of the domain under study is not explicitly defined in the ontological model or could not be inferred from the ontological model. Proving that there is a one to one relationship between the ontological constructs and real world constructs is not a straightforward task. It is normally more practical to disprove the completeness of an ontological model in an inverse manner by showing that a concept is missing from the model. As long as completeness is not disproved, it could be considered that the ontological model is complete. In addition to human intervened check in the standards TC, competency questions are believed to be

a suitable means for disproving the completeness of a model as explained later in this section.

In addition to the above, it is important that an ontological model represents the rules and axioms which should be enforced to the structure of the conceptual model – the commitment layer. Proving the completeness of the commitment layer of an ontology is also not a straightforward task. The approach taken in this research is that each mandatory normative statement in a specification **MUST** be modelled in the commitment layer. Recommended and permitted normative statements are suggested to be modelled; however, their non-existence would not affect the completeness of the ontological model in *OntoStanD*. Competency questions are considered as a suitable means for examining the above: if an axiom which should have been defined is not defined, there wouldn't be a suitable answer for a specific competency question.

As explained in Chapter 2, Gruninger and Fox (1995) recommend using competency questions for scoping and evaluating the completeness and expressivity of a designed ontology. This work utilises their approach for evaluating the completeness and also usability of an ontological model of a standard. For this, a set of competency questions should be defined by the domain experts, i.e. standards developers/TC members, for each scenario and the ontological model of the standard would then be evaluated against these competency questions. Gruninger and Fox (1995) define two sorts of competency questions: (1) informal competency questions, which are written in natural language, possibly in the earlier stage of ontology development; (2) formal competency questions, which are formalised using a query language on top of the formal, axiomatised ontology. Therefore, the formal competency questions should be formalised in a language compatible with the language the ontology itself is defined in. In this respect, if an ontological model is capable of answering the defined competency questions, it is concluded that the

ontology is complete and usable (Gruninger, Fox 1995) and consequently the method used for their development is practically adequate.

To sum up, a complete ontological model of a standard in the context of this thesis is one which (a) provides a one to one mapping to the real world phenomena, (b) has a complete ontology base with respect to Wand and Weber (1993) and (c) has a complete commitment layer as defined above. All these three, are measured by the means of competency questions, defined by the domain experts, and also manual check in a standards technical committee.

3- Usability

Folmer and Putner (2011) define usability of a standard as the extent to which it could be understood, learned and used/applied by users in the specific situation. This work fully supports their definition. The usability of a standard and its ontological model is not easy to measure and evaluate unless it is used by its own community. However, as explained above, competency questions could be used for evaluating the usability of an ontological model to a good degree. Further evaluation would be possible only after an ontology-based standard is adopted and used in a community.

4- Customisability

Gruber (1995) defines customisability as minimal ontological commitment and extendibility, while Gomez-Perez (2001) only sees on expandability of an ontological model. Ontological models of standards are conceptual foundations which should be designed in a way that leaves room for different users to monotonically instantiate and specialise the ontology so as to fit their particular settings. It also facilitates using the ontological model as a basis for profiling standards. Thus, an ontological model of a standard should be designed in a manner which enables various users expand the existing standard - and its ontology - without altering the original one. It should also allow users to use

only a part of the standards and its shared vocabulary - a certain profile of a standard. Therefore customisability here is seen as both customisability and expandability as Gruber (1995) suggested. It also makes a standard *maintainable*.

The reason for choosing the above four criteria is that the main point here is to measure the quality of the conceptual model of a standard which is in this case modelled using ontologies. Therefore a set of criteria which cover all these aspects are selected. Furthermore, factors such as level of adoption or availability of tools are not of our concern at design time which is the main concern of this work, since they could be measured only a while after a standard is actually ready to use and are not necessarily affected by the quality of the standard itself, but by political factors and market forces.

An alternative for evaluating the ontological models may be considered to be comparison of resulting ontological models of standards by some existing gold standards. This research takes the view that there is no guarantee that there exists a gold standard for any intended domain, which a standard is being developed for. More importantly, gold standard comparison is normally undertaken for comparing an automatically generated ontology with one which is created manually by domain experts. This is not the case in this research since the ontology is created manually by the domain expert. Besides, as indicated by the name, the aim of a standard is to provide a gold standard in its domain, which may be used for the purpose of interoperability, as well as comparison. Thus the ontology developed for a standard, by its stakeholders, may be used for future gold standards comparisons for evaluating the quality of automatically generated ontologies.

Regardless of the evaluation method taken, at the end of the evaluation phase the researchers can decide whether to iterate back to the 'development' phase to

try to improve the effectiveness of the artefact or to continue on to the next phase and leave further improvement to subsequent projects (Peppers et al. 2007). Iterating back, if needed, can provide essential feedback to the design and development phase as to the quality of the design process and the design artefact under development (Hevner et al. 2004). A design artefact is considered to be complete and effective when it satisfies the requirements and constraints of the problem it was meant to solve (Hevner et al. 2004). Typically this phase is due when the results are ‘good enough’ and not necessarily the best. In other words there may be still deviations in the behaviour of the artefact from the revised hypothetical predictions (Vaishnavi, Kuechler 2007).

3.3.5 Contribution and Communication

‘Conclusion’ and ‘communication’ are the last activities of Vishnavi (2007) and Peppers et al.’s (2007) methodologies respectively. However, they do not serve the same purpose. The aim of ‘conclusion’ is to finalise the research and write about the results, while ‘communication’ refers to publishing scholarly papers and demonstrating the importance of the solved problem to the research community. Hevner et al. (2004) also propose the need for communication in their proposed guidelines, which indeed corresponds to the ‘communication’ activity of Peppers et al. (2007) methodology. Although these two activities, ‘conclusion’ and ‘communication’ are not exactly the same, they can complement each other and are better to be included in a design science research methodology as a phase called ‘contribution and communication’.

The ‘contribution and communication’ is therefore the final phase in the design science research methodology as adopted in this thesis. Effective design science research must clearly contribute to the new and interesting solutions to unsolved problems (Hevner et al. 2004). In this phase the problem and its importance, the artefact, its utility and novelty, the rigor of its design and its

effectiveness need to be communicated to the relevant audience (Peppers et al. 2007). Communication may be done by publishing papers in related journals or conferences (Peppers et al. 2007), participation in various related academic or practical projects or standardisation activities.

3.4 The Application of Design Research

This research presents a systematic design method for developing ontology-based standards, called ‘OntoStanD’, which not only is developed based on the design science guidelines, but also incorporates the same design paradigm. Following design research guidelines, this research is conducted in an iterative and incremental manner where each iteration extends, refines and eventually evaluates the artefact(s). Figure 3-8 summarises the research design and iterations of this thesis, which are described in more details in the following.

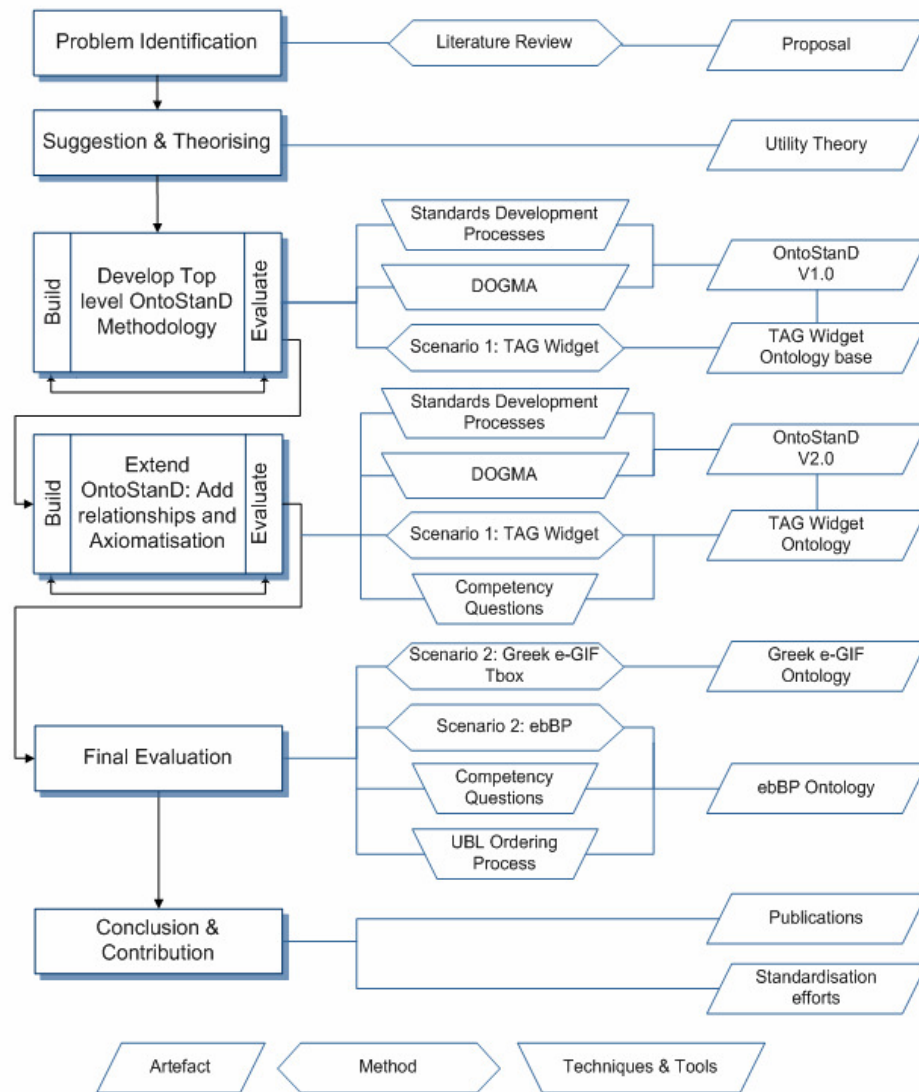


Figure 3-6. Research Design

3.4.1 Problem Identification

This phase starts by understanding the problem in the field to suggest a type of solution that forms a proposal to be developed further in the remaining of the research. The problem identification in this thesis has been achieved via a literature review as presented in Chapter 2 where the problem space and solution space were studied. The problem, as defined in Chapter 1, is that *no*

systematic design methods exist for developing the structure, content and semantics of standards, which hampers their quality, efficacy, comparability and interoperability. OntoStanD approaches the above problem from a semantic perspective and has been developed (iteratively) to provide the benefits noted in Chapter 4 - with particular emphasis on the interlinked points of *developing new standards, formalising concepts in existing standards and the reengineering of existing standards based on ontological analysis.*

3.4.2 Suggestion & Theorising

After clarifying the problem, a utility theory, which aims at suggesting a meta-design for the above problem is defined. This is done by conducting a more detailed study on the problem space and the solution space, synthesising various types of standards and also ontology engineering methods and sketching out the requirements for a robust standards development process. The utility theory in this thesis is that *computational ontology, when applied methodologically, will efficiently provide improvements in the structure, content and semantics of standards specifications.*

3.4.3 Development

Development is when the main artefact of the research is created. In the realm of this thesis a *method* (according to March & Smith (1995) definition) is created. The corresponding artefact in this thesis is the OntoStanD methodology. However, there are a set of other artefacts created in the various iterations of this thesis. Table 3-7 presents the artefacts which are the output of the design research in this thesis and demonstrates how they relate to various phases of DSR.

Design Research Facet	OntoStanD Interpretation	DSR Phases
Utility theory	Ontological approach – view that clear articulation of things, relationships and axioms improves quality.	Problem identification/Suggestion & theorising
Construct	The resulting standard and its modelling grammar	Development
Model	Ontological conceptual models	
Method	OntoStanD methodology	
Instantiation	Application of OntoStanD to various scenarios, e.g. OASIS TAG Widget, Greek e-GIF and ebBP.	Evaluate

Table 3-7. DSR artefacts produced in this thesis

As noted earlier, design research is an iterative process, where each iteration builds on the previous one. As new knowledge is gained during the various iterations in design research, new suggestions may be used for the subsequent iterations. The ‘development’ phase in this research therefore is composed of two iterations as follows, followed by final evaluation phase as explained in Section 3.4.4:

Iteration 1- Develop OntoStanD: Ontology base layer

This iteration proposes a top level architecture for OntoStanD, providing a set of guidelines for developing the conceptual model of a standard in an ontological manner. The outputs of this iteration are (1) the first version of the method and (2) the OASIS TAG ontological model – the base layer.

Evaluation takes place at the end of this iteration by applying OntoStanD to the OASIS TAG Widget specification. The OASIS TAG Widget specification is a scenario which OntoStanD is applied to, for the purpose of demonstrating the consistency, completeness, usability and customisability of the model artefact and consequently demonstrating utility and practical adequacy of OntoStanD at this stage.

Iteration 2 - Extend OntoStanD: Commitment layer

The aim of this iteration is to further improve and extend OntoStanD. In this iteration OntoStanD is extended by providing guidelines for defining and formalising the normative statements of a standard on top of the ontological conceptual model, and optionally defining test assertions, by adding axioms and rules to the existing ontology – the commitment layer. Similar to Iteration 1, this iteration further presents a prototype application of OntoStanD by applying it to the TAG Widget pseudo specification scenario. The output of this iteration is the second version of OntoStanD methodology and a complete ontology for TAG Widget. The resulting artefact in this iteration, the TAG Widget ontological model, is evaluated against the evaluation criteria defined earlier in this chapter.

3.4.4 Evaluation

The aim of this phase is to demonstrate the utility and practical adequacy of OntoStanD by applying it to two more scenarios and evaluating the quality of the resulting models. The scenarios used for this purpose are: (a) The Certification Framework for Public Administration Sites and Portals specification of the Greek e-Government Interoperability Framework (e-GIF) standards and (b) the OASIS ebXML Business Process Specifications (ebBP). The learning from this iteration would be reflected on the OntoStanD as the final artefact of this research and the designed ontological models are evaluated against the model evaluation criteria defined earlier in this chapter.

This research, broadly takes an ex-post evaluation of a design artefact of type *descriptive*, according to Hevner et al. (2004), , utilising both scenarios and informed arguments for evaluating the utility of OntoStanD. The evaluation in this research is also positioned as artificial in the sense that the users are not

realistic (they are the researchers), and the methodology is a prototype scenario – the problem to which OntoStanD is applied is real enough however. The evaluation perspective and approach is summarised in Table 3-8.

Evaluation Perspective	Approach Taken
When evaluation takes place	Ex-post in a context of simulated pilot scenarios. The approach is exploratory with a focus on identifying usability problems rather than validating usability per se.
What is evaluated	The ability of OntoStanD as a design process primarily in terms of its fitness for use (i.e., how well it guides the process of production and specification).
How it is evaluated	Artificially via application to sub-section of standards from the OASIS TAG Widget, Greek e-GIF and OASIS ebBP.

Table 3-8 – Summary of evaluation approaches taken

The evaluation in this thesis is undertaken at two levels: (a) At the end of each iteration, evaluating the resulting artefacts of each iteration, mainly of type ‘model’ based on the defined criteria; (b) A final evaluation of OntoStanD as an artefact of type method.

Models are evaluated with respect to the defined evaluation criteria for ontological conceptual models as follows:

1- Consistency:

Consistency of the models are checked using automated tools. In this thesis, the consistency of models are checked utilising Protégé and the Pellet reasoner.

2- Completeness

Completeness of models are measured at two levels: (a) ontology base and (b) commitment layers. The completeness of the ontology base layer of standards should be examined with respect to the Wand and Weber (1993) guidelines. This is a manual process and should be executed by standards developers until they reach consensus. Simple competency questions can also facilitate this process. Evaluating the commitment layer is however, heavily dependant on the competency questions. Therefore defining a set of suitable competency

questions is an important activity at this stage which should be performed at the end of each iteration of this research. The first step at this stage would be to define informal competency questions and then formalise them using an ontological query language. DL queries are a query language designed for OWL ontologies and embedded in Protégé, which are used in this thesis for the purpose of formalising the informal competency questions.

3- Usability

As mentioned in Section 3.3.4, competency questions are further used for demonstrating the usability of models: if a model is capable of answering the competency questions, consequently it is considered to be a usable model in its domain. However, a full usability evaluation would only be possible only after an ontology-based standard is adopted and used in a community. This level of usability test hasn't been possible to achieve in this research since more time is needed for standards communities to actually adopt and use the method.

4- Customisability

Customisability of the models in this thesis is evaluated with respects to their minimum ontological commitment and expandability, while maintaining their completeness. The double articulation of OntoStanD is considered as the most important factor in providing customisability to the models developed on its basis.

3.1 Summary

This chapter presented the research method adopted in this thesis for designing and evaluating OntoStanD methodology, its artefacts, and the rational behind choosing this method. To choose the most suitable design research method, various existing DSR methods were analysed and compared, and a suitable research method for this thesis was defined

with respect to the requirements of this project. Furthermore an in depth discussion on the design evaluation was provided as evaluation is believed to be an important but neglected aspect in the design science research. As a result of this analysis a suitable evaluation method and a set of criteria were defined for evaluating the design artefacts in this thesis. Afterwards the research design of this thesis was articulated and each iteration of the research was discussed in details.

CHAPTER FOUR: Towards Ontology-based Standards

4.1 Introduction

This chapter discusses ontology-based standards and the reasons why they are considered to be a desirable approach for standards development. It further introduces the OntoStanD methodology, as the main design artefact of this research, which provides a methodological approach for designing and developing ontology-based standards. OntoStanD is applied to a simple pseudo specification of OASIS TAG TC, the Widget specification, as a working example, while key aspects of OntoStanD are being discussed in details. Please see the Widget specification in Listing 4-1.

This chapter is organised as follows. Section 4.2 provides an overview of semantic interoperability and the role of standards in interoperability. Section 4.3 discusses standards development process and how ontologies could be utilised in such a process. Section 4.4 presents characteristics of a knowledge intensive community, seeing a standards development community as such a community. Section 4.5 provides a more detailed overview of the DOGMA methodology. Section 4.6 introduces OntoStanD and provides an overview of its top level architecture, while providing a set of guidelines for designing ontological conceptual models for standards. Section 4.7 evaluates the first iteration of this work and Section 4.8 summarises this chapter.

4.2 Semantic Interoperability

Standards may be argued to be important enablers for achieving interoperability as they aim to provide unambiguous specifications for error-free exchange of documents and information. The quality of standards is an important factor in achieving interoperability. Juran (1988) define quality in general as fitness for use, which is inline with the ISO 9126 definition of software quality as “the totality of characteristics of an entity that bears its ability to satisfy stated and implied needs” (ISO/IEC. 2001, Folmer et al. 2011 pp. 100). Importantly, Folmer et al. (2011) further define the quality of standards as their ability to achieve semantic interoperability effectively and efficiently.

Semantic interoperability is considered to be an essential factor for effective interoperation. Semantics require that the meaning of terms, their relationships and also the restrictions and rules in the standards should be clearly defined in the early stages of standard development and act as a basis for the latter stages. By implication, therefore, it is important to model and represent the concept of a standard in a clear, precise and unambiguous way.

Folmer et al. (2011) have conducted a survey on the current state of the quality of standards, as part of which they question whether the quality of standards can be increased by improving the standards development process. As an outcome, 64% of the standards developers who participated in the survey (from 34 different standards) agreed with the proposition. More importantly 66% of the respondents (standards developers) agreed that improved quality of standards leads to improved interoperability. This implies that the quality of standards and the way they are developed are subject to improvement. The European Commission has also recognised the need for modernising standardisation in the EU and set a policy in 2009 for increasing the quality, coherence and consistency of ICT standards (European Commission 2009).

In addition to the absence of a formal description, Sherif et al. (2005) identify a set of problems which hinder the quality of standards. They include missing information, inconsistency in the same specification, lack of tools/solutions for conformance and interoperability test, absence of formal descriptions and miscommunication. Standards specifications are usually written over a relatively long period of time, which inevitably lead to a certain degree of ambiguity and inconsistency due to the way different people, in most cases from different countries, use and interpret natural language. The author is in agreement with Sherif et al. (2005) that using formal representation languages for standards will substantially help in producing less ambiguous and clearer standards specifications and therefore higher quality standards and consequently better degree of interoperability.

4.3 Standards Development and Ontology

Standards development organisations usually provide a set of guidelines for the process of developing and approving standards (as explained in Chapter 2). These guidelines are more concerned with the administrative aspect of the process and/or provide a template for textual specifications to be developed however. As a consequence, the state-of-the-art in the field lacks practical support for developing the structure, content and semantics of a standard specification. In short, there is no systematic development method currently available: (a) For developing the conceptual model underpinning a standard; and/or (b) to guide a group of stakeholders to develop a standard specification, which a number of parties may need to comply with at a sector, national or international level. This research proposes the use of semantic web technologies, and in particular ontologies, for developing standards which are clear an unambiguous, more test friendly, and of higher quality. Ontologies may

contribute to the *development, extension and improvement* of standards specifications in the following ways:

1. *Formalising concepts within existing standards*, which would result in a more stable definition of semantics in a standard and allow the writing of expressions based on clear, unambiguous terms and categories. Specifications serve as a foundation for developed ontologies and therefore the ontologies can be evaluated against the standards specifications and/or their XML Schemas (The Ontolog Community 2010).

2. *Reengineering of existing standards based on ontological analysis*, identifying their potential problems and semantic ambiguities with the aim of improving their quality (The Ontolog Community 2010).

3. *Facilitation of integration between different standards*, or multiple implementations of the same standard. This is possible when standards are already defined in an ontological manner, preferably using a methodological approach, such as OntoStanD as explained later in the chapter.

4. *Development of standards*, wherein ontologies are used throughout the standard development phases, from start to finish. This approach can be taken where appropriate in developing new standards or new versions of existing standards (Heravi et al. 2010b).

Ontology-based standards development not only brings all the components of a standard specification under one single umbrella but also it formalises the concepts and relationships of the standard, enables logic based reasoning, assists in automatic consistency and conformance checking, facilitates change management and reduces errors in the specifications and their conceptual models, which potentially leads to higher quality standards. Another benefit of having ontology-based standards is the possibility of deriving (semi)automatic textual/HTML specifications from the ontology itself. This would require

annotating the concepts, relationships, restrictions and rules in the standard ontology.

4.4 Community-based Knowledge Management

A standard is usually developed by a group of stakeholders. Thus, a standards development committee is considered as a knowledge intensive community, where information systems are represented by stakeholders. Semantic interoperability between systems is a focal point in such communities, which is accomplished via a coherent community based ontology evolution process performed by its stakeholders (De Leenheer 2008).

De Leenheer (2009) introduces a business semantic management methodology for community based knowledge management and ontology modelling (presented in Figure 4-1), which is aimed at bridging the gap between the social and technical parts of a community. This thesis strongly supports their proposition that there should be an amalgamation of the semantic web and Web 2.0 for developing standards. Web 2.0 is mainly about interaction and collaboration on the web for creating user-generated content in a virtual community and facilitates achieving a collective intelligence in a self-organising manner in a knowledge intensive community such as a standards development committee (O'Reilly 2005).

Standards development has been undertaken in a community-based manner for long. Thus it could be said that the standardisation committees have been materialising the underlying concepts of Web 2.0 for a relatively long time. However, the semantic web has hardly benefited from Web 2.0 phenomena and the combination of the two – the Social Semantic Web (Breslin et al. 2009), with standards development, is considered to be well due now.

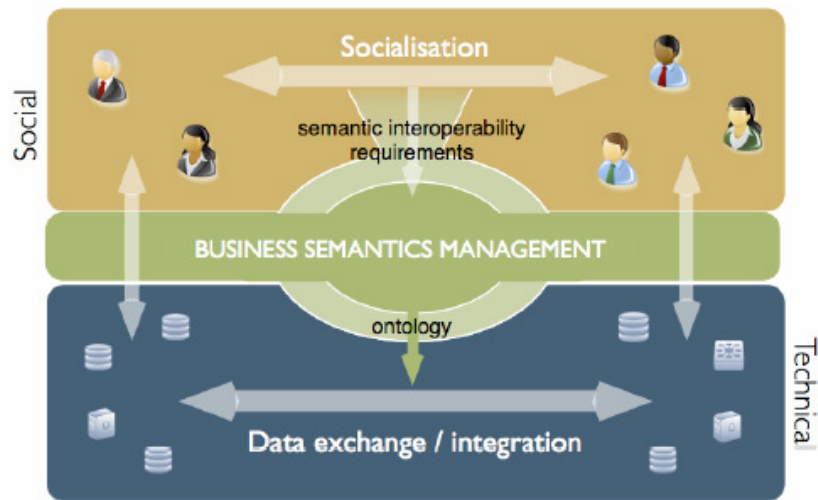


Figure 4-1. Business Semantic Management paradigm, adopted from (De Leenheer 2009)

De Leenheer (2009) identifies a set of principles for business semantics management in knowledge intensive communities as follows:

- *ICT Democracy*: An ontology should be defined by its community, and not by a single developer.
- *Emergence*: Semantic interoperability requirements emerge from community evolution processes.
- *Co-evolution*: Ontology evolution processes are driven by community evolution processes.
- *Perspective Rendering*: Ontology evolution processes must reflect the various stakeholders perspectives.
- *Perspective Unification*: In building the common ontology, relevant parts of the various stakeholder perspectives serve as input for the unified perspective.

- *Validation*: The explicit rendering of stakeholders' perspectives allows us to capture the ontology evolution process completely, and validate the ontology against these perspectives respectively.

It is clear from the above principles that they all hold for standards development and in the case of this thesis, for ontology-based standards development. Thus, the above Business Semantics Management guidelines are taken into consideration for developing OntoStanD. Similar to OntoStanD, De Leenheer (2009) supports using the DOGMA methodology as a suitable framework for community based knowledge management and evolution and adopts it in their own business semantic management methodology. DOGMA is explained in details in the following section.

4.5 DOGMA

DOGMA is a community-based ontology engineering approach, which is grounded in natural language fact analysis. These, in addition to its other capabilities, presented in Table 2-5 in Chapter 2, make DOGMA a suitable method for ontological domain conceptualisation to be utilised in OntoStanD. DOGMA explicitly distinguishes between general purpose semantic constructs rooted in informal natural language descriptions, which are mainly intended for human interpretation, and the formal representation of these constructs for specific computerised applications such as semantic interoperability (Jarrar, Meersman 2008, Spyns et al. 2002). In other words the concepts and their relationships are separated from their semantic constraints in DOGMA. This separation is called the double articulation principle (Spyns et al. 2002).

The DOGMA double articulation decomposes an ontology into an ontology base layer and a commitment layer (Spyns et al. 2002). An ontology base is a set of binary fact types, which are called *lexon* and represent the basic domain

vocabulary and relationships between terms. The commitment layer defines a set of rules (axioms) on top of the ontology base, constraining an ontology base in a specific context. Therefore an ontology base may have different ontological commitment through various commitment layers defined. A schematic representation of the DOGMA double articulation is presented in Figure 4-2 using a simple example of a ‘book’, adopted from (Spyns 2005). Note that the commitment layer of the ontology in Figure 4-2 is presented using ORM (Object Role Modeling) constraints, which largely correspond to OWL restrictions.

ORM is a method for designing and presenting conceptual models, which is also used for ontological modelling, particularly in DOGMA. ORM uses natural language and intuitive diagrams which make it easily understandable. Being used in DOGMA, based on natural language and easily understandable, makes ORM a suitable choice for graphical representation of conceptual models in this thesis. However, it is important to note that ORM only provides a graphical representation of the conceptual models, which could be replaced by any other graphical notation if preferred. The details of ORM are out of the scope of this thesis and therefore not discussed any further here. For more information on ORM please refer to (Halpin 2001). The point here is to show that the concepts and their relationships are defined in the ontology base layer and their restrictions in the commitment layer.

Figure 4-2’s ontology base layer indicates that a ‘book’ has title, price, author, publisher and ISBN. It however, doesn’t include any restrictions on the relationships between the ‘book’ and its related entities. These restrictions are defined in the commitment layer of Figure 4-2, which reads a ‘book’ has at most one ‘title’ and that a ‘title’ must belong to one ‘book’. Furthermore, the relationship between ‘book’ and ‘price’ is defined as an m:n relationship. The

application layer is the realisation of the concepts and relationships defined in the other two layers.

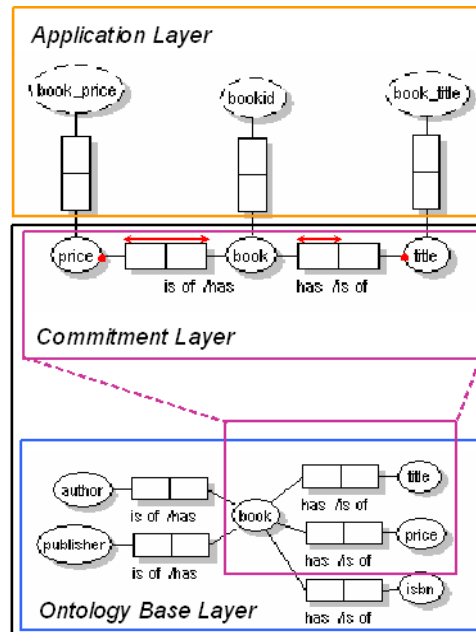


Figure 4-2. The DOGMA double articulation, adopted from (Spyns 2005)

4.6 OntoStand

As noted earlier, this research proposes that ontology can help standards developers and stakeholders for improving the conceptual models (Shanks et al. 2003) and providing a robust and shared understanding of a domain. With this background, a methodological approach for developing ontology-based standards and their conceptual models is provided in this chapter, which is called OntoStand (Ontology-based Standards Development).

OntoStand is a systematic design methodology that utilises semantic web technologies to assist standards development process and/or refining standards specifications, seeing the following commonalities between standards and ontologies:

- *Community aspect*; a standard is created by a group of stakeholders while an ontology is created by a community of domain experts.
- *Interoperability aspect*; a standard records a statement of sharable items for conformant parties, while an ontology contains formal agreements made by domain experts within a community of interests.

OntoStanD may be used for developing new standards specifications from scratch or new versions of existing standards, re-engineering existing standards and also formalising concept in existing standards as explained earlier. OntoStanD seeks to provide suitable solutions towards addressing the problems and gaps identified in Chapter 2 as follows:

- Providing robust guidelines for defining clear and unambiguous conceptual models of standards, with formal basis;
- Providing a systematic design methodology for standards development;
- Providing a set of methodological guidelines for ontologising existing standards;
- Provides a coherent methodological approach for designing ontology-based standards and utilising ontologies throughout the standards development life cycle.

The potential value of OntoStanD is in providing a comprehensive, clear and unambiguous method for developing robust IS standards, which are more test friendly and of higher quality. OntoStanD also potentially facilitates standards conformance testing and change management, impacts interoperability and assists in improved communication between standards development team. Last, OntoStanD provides an approach that is repeatable, teachable and potentially general enough for creating any kinds of information standard.

In addition to corresponding to design research phases, the phases and activities of OntoStanD take into account the typical standardisation process as presented in Chapter 2, which was the result of extensive study and analysis on various standard creation processes being used in different standardisation bodies. It also utilises DOGMA (Spyns, Tang et al. 2008) mainly for the phases concerning capturing domain semantics, i.e. *Domain Conceptualisation*.

The first version of OntoStanD, which was designed in the Iteration 1 of this research, is presented in Figure 4-3. Each round-cornered rectangle is either a *phase* or an *activity*. A phase is a collection of activities, which can be a composition of several other activities. The dashed rectangles represent optional activities and an arrow-tipped bar indicates an execution path between two phases or between two activities. Note that each phase has an output, which is used as an input for its follow-up phases.

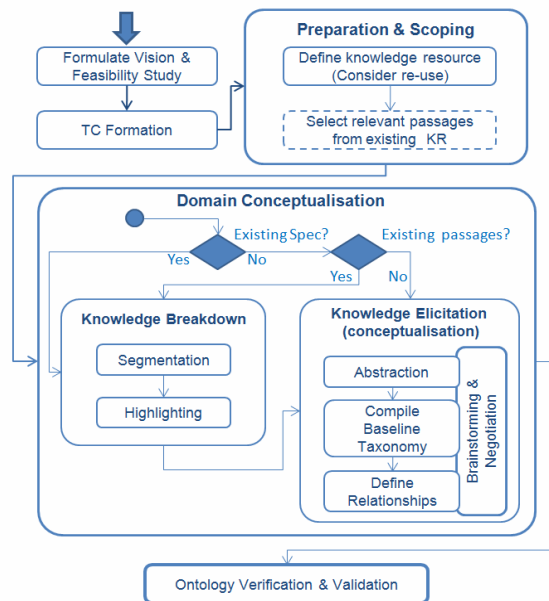


Figure 4-3. OntoStanD V1.0

The details of the phases and activities of OntoStanD are provided in the following.

4.6.1 Formulate Vision and Feasibility Study

A vision, or proposal, is a compelling view of a standard that is going to be built. Based on the analysis in Chapter 2, typically a small group of interested parties develop a proposal for a new standardisation effort to be submitted to a standardisation body. The proposal may be created by any community of parties that are interested in creating a future standard and is usually submitted to a standardisation body for a feasibility study. The standardisation body takes the decision on whether it is feasible to create the proposed standard or not. Once a positive decision is taken, and if the standardisation follows an open process (e.g. an open standard is being developed), the proposal will normally be available on-line for comments after the initial acceptance of the standardisation body. The process of submitting a proposal to a standardisation body and its acceptance may vary in different standardisation bodies.

4.6.2 Technical Committee Formation

Technical committees of the standardisation bodies are usually composed of a number of interested parties, communicating via email, routine (usually weekly or bi-weekly) teleconferences and sporadic, optional face to face (F2F) meetings. When a standard proposal is accepted and finalised, a call for participation would be posted on-line for the interested parties to join the Technical Committee. The TC members may be individuals or representatives of public or private organisations. At this stage the TC has to be formed based on the drafted requirement in the previous phase. Depending on the standardisation body, a TC may be composed of various roles. For example an OASIS TC is usually composed of one or two TC chairs, zero or more secretary(ies), one or more OASIS staff contact(s), any number of voting members, any number of members and any number of observers.

OntoStanD's minimum role requirement, following DOGMA's guidelines, is to have one knowledge engineer, who may be one of the TC members, one key domain expert, who may be the TC chair and several domain experts, who are the TC members. A domain expert, who is also a stakeholder in the TC, has good insights about the domain which reflects the interests from his/her organisation. They form the main part of a TC and are responsible for providing domain specific knowledge. The key domain expert holds an overview of the knowledge in the domain and knows the overlapping interests of these organisations. She/he is responsible for scoping the problem, defining knowledge resources and helping the knowledge engineer by accelerating the negotiation processes. The knowledge engineer is responsible for running OntoStanD, in collaboration with other TC members, and needs to know how to model the information, which is either from the sessions of text processing (in the phases of domain conceptualisation, axiomatisation and specification reification) or from the brainstorming sessions (in the activity of knowledge elicitation), into desired formats (e.g., ontological models). Note that the knowledge engineer doesn't need to be a domain expert or a key domain expert. In this phase, the TC is established and the members are appointed to play one of the above three roles.

4.6.3 Preparation & Scoping

The goal of this phase is to scale down the problem, if needed, in order to facilitate reaching the final goal. This phase is composed of two activities: define knowledge resources and select relevant passages, which are depicted in Figure 4-4 and explained in the following two subsections.

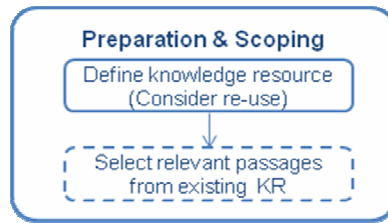


Figure 4-4. OntoStanD Preposition & Scoping Phase

Define knowledge resources

The knowledge resources need to be defined according to the purpose and requirements defined in the *Formulate Vision and Feasibility Study* phase. The key domain expert and the knowledge engineer are responsible for collecting and selecting the knowledge resources, which will be used in subsequent phases. In particular, structured domain resources, such as domain dictionaries, database schemas and taxonomies, provide material for eliciting concepts in the phases of *Domain Conceptualisation*. Logical statements (structured or unstructured) provide material for defining rules and axioms as explained in Chapter 5. Other types of resources, such as existing standard specifications, textual descriptions, meeting minutes, mailing lists, stories, images and videos may be used to assist designing a standard's specification itself.

Select relevant passages from existing knowledge repositories

After the knowledge resources are defined and collected, the knowledge engineer, key domain expert and/or a group of domain experts (TC members) need to categorise them in the following mutually inclusive categories:

- *Core texts and explanatory texts/resources*; Core texts are the documents which will be used to create concepts and axioms in the ontology. Examples of such resources are existing specifications, papers or technical reports. The explanatory resources, which can be texts, videos and audios, are considered as the supplementary materials to support defining the ontology. Examples of explanatory resources are

recordings of teleconference and/or face to face (F2F) meetings and meeting minutes.

- *Application specific resources*; Resources that are relevant to the application of a standard should be carefully selected. For example if a Business Process standard is being formulated, white papers or best practices in business process management could be considered as relevant application specific resources. Information on how the business processes are packaged and transmitted between trading partners may, however, be considered as irrelevant.
- *Structured and unstructured resources*; Structured resources have clear syntax and are well organised. For example, an XML file is a structured resource, while a white paper or a teleconference minute is an unstructured resource.

The output of this step is a set of relevant passages. Note that the main focus of OntoStanD is to deal with textual passages. Nevertheless, it does not mean that the passages that are in other formats should be thrown away. They are possibly not as straightforward to be used directly, yet important to support the textual passages.

4.6.4 Domain Conceptualisation

The first, and arguably most important, stage of developing a standard is to agree on the scope and the terms used in the domain and the relationship between them - the conceptual model of a standard. Domain conceptualisation, or conceptual modelling, is the process of structuring and formally describing some aspects of the real world, the domain knowledge, into a conceptual model which demonstrates the problem and its solution with the aim of understanding and communication (Fernandez et al. 1997, Mylopoulos 1992). Conceptual

modelling of a standard forms the kernel of a standard and therefore is a significant phase in the standards development lifecycle. Attaining high quality conceptual models for standards from the early stage of their development is a critical goal and thus it should be conducted in a rigorous way. Errors made in the early stage of standards development could be costly to fix later or even worse, have an adverse effect on the interoperability which is the ultimate goal of standards.

Gruber (1995) defines an ontology as a specification of a conceptualisation, which then lead others to consider theories of ontology an important basis for improving the quality of conceptual models (Guarino 1998, Weber 2003). Ontological theories are aimed at providing faithful representations of a part of the real world, which as noted in Chapter 3, are accurate, complete and consistent (Shanks et al. 2003). Using ontologies allows for inclusion of the semantics of the domain as part of its conceptual model. Ontological models are further intended to be read and processed by machines. Therefore, conceptual models represented in an ontological manner can benefit from automated processing, reasoning and verification, which is not the case if presented in a textual or graphical manner (Bera, Wand 2010).

As noted in Chapter 2, there is no clear guideline in standardisation bodies regarding the *Domain Conceptualisation* of standards. Accordingly OntoStanD proposes the use of ontology as a basis for standards conceptual modelling and the means of providing a clear and repeatable guideline to be followed for this purpose, forming a solid foundation for its following phases and activities. This phase results in a stable ontological conceptual model which has consensus among the people involved in the development of the standards.

The *Domain Conceptualisation* phase of OntoStanD covers the definition of the ontology base layer of a standard and is composed of two main activities as

depicted in Figure 4-5. It starts by considering if there are any existing specifications or other relevant material to be re-used. If this is the case the *Knowledge Breakdown* activity should be followed. Otherwise the user can proceed to the *Knowledge Elicitation* phase.

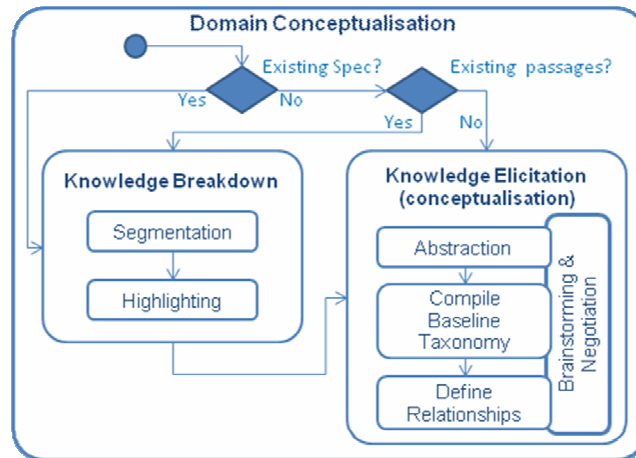


Figure 4-5. OntoStanD Domain Conceptualisation phase

Knowledge Breakdown

In this activity, the domain of interest is decomposed into a number of smaller topics, which are more easily manageable. This activity is performed when there are existing specifications, earlier versions, or a set of passages selected in the *Preparation and Scoping* phase. Otherwise this activity is omitted and the *Knowledge Elicitation* activity is performed straightaway.

Spyns et al. (2008) suggest using Narratological Schemas for knowledge breakdown in DOGMA. Narratological Schema (NS) has its root in *stories* (or *storytelling*), which are considered as an appropriate method for scoping knowledge. NS allows tracing/logging changes, supports the communications between technical and non-technical domain experts and articulates tacit knowledge. Using NS, the knowledge engineer focuses easily on smaller but well scoped relevant passages. An NS contains the following items:

- Settings: show the background knowledge of the elements in the NS.
- Characters: defines the actors in the NS.
- Episodes: a scenario normally contains a set of episodes in a chronological sequence. If there is no chronological sequence that can be applicable, then they are considered as simply collected statements.

An empty NS form is presented in Table 4-1.

Narratological Schema (NS) Form					
Title	<title of this narratological schema>				
Author	<the author name of this NS and optionally his/her contact information>	ID	<ID of this document >	Date	<creation date of this NS>
Theme	<theme of this NS>				
Purpose	<short description of the purpose>				
Scope	<the reference to the ontology scoping form (the resource which the NS is being built from)>				
Settings					
S1	<setting of the NS>				
S2	<setting of the NS>				
...	...				
Characters					
ID	Role	Description			
C1	<character of the NS>	<description of the role>			
C2	<character of the NS>	<description of the role>			
...			
Episodes					
E1	<episode or scenario of the NS>				
E2	<episode or scenario of the NS>				
...	...				

Table 4-1. An empty Narratological Schema (NS) form, adopted from (Spyns et al. 2008)

As mentioned in the introduction, OntoStand is applied to a small pseudo standard from OASIS, called TAG Widget as presented in Listing 4-1. The reason for choosing the OASIS TAG Widget Specification extract is that while it is a small and handy specification, it covers various aspects of a standard’s specification, and therefore is considered as a suitable example for demonstrating the application of OntoStand.

Section 100: A widget MUST be of rectangular shape.

Section 101: A widget of medium size MUST use exactly one AA battery and have a red button on top.

Section 102: It is RECOMMENDED for a widget to be waterproof. If it is not waterproof then it MUST have a warning label stating that it is not waterproof.

Section 103: A widget MAY have a metallic casing. If it does have a metallic casing it MUST have a waterproof coating.

Section 104: Localizations of Widget Size

For implementations of widgets for use in the European Union a widget that weighs between 100g and 300g and is from 5 to 15 centimetres long in its longer dimension, is a medium-size widget. However, in USA the widget is medium-sized if it weighs between 4oz and 12oz and is from 2 inches to 6 inches long.

Listing 4-1. The OASIS TAG Widget specification extract (OASIS TAG TC 2011)

Table 4-2 presents an example of a NS for the TAG Widget specification extract. In this specific case however, the specification itself is highly structured and therefore the NS doesn't provide a high degree of knowledge segmentation. In many cases however, this is not the scenario and the details of a NS should be extracted from text passages and graphics. Thus, using NSs are considered optional for knowledge breakdown in OntoStanD. Depending on the requirement of any TC and the degree which the existing knowledge resources are structured, any other mechanism may be selected by TC members to be used for knowledge segmentation.

Segmentation and Highlighting

When the NSs are defined, they should be gradually decomposed into structured conceptual blocks. The goal of segmentation (Zhao, Meersman 2005) is to discover atomic textual segments, which are stored as a scenario, an episode or a paragraph in the NSs. The segmentation technique is used to trim a long sentence into several Subject-Verb-Object style, atomic sentences.

The goal of highlighting on the other hand is to discover important concepts, relations and possible axioms (Zhao 2004). Spyns (2008) suggests that three types of phrases are mostly to be highlighted: noun phrases, verb phrases and prepositional phrases. OntoStanD expands their suggestion, proposing that restriction keywords, such as 'MUST', 'SHOULD' and 'MAY', should also be highlighted since they are fundamental in the standards specifications and significant for creating axioms in the ontology. The highlighting is demonstrated in colours in Table 4-2. Yellow highlights (lighter grey in black & white) are the terms and green highlights (darker grey in black & white) are the restriction keywords.

Title	OASIS TAG Widget Spec extract				
Author	Bahareh R. Heravi	ID	TAG.Section.5	Date	20/10/2010
Theme	An example Widget specification for OASIS TAG TC Guidelines				
Purpose	To provide an example of a standard's specification for defining Test Assertions				
Scope	Section 5 of the OASIS TAG Guidelines.				
Settings					
ID	Description				
S1	A description of Widget – not provided in the spec extract.				
Characters					
ID	Role	Description			
C1	Widget				
Episodes					
E1	A widget MUST be of rectangular shape.				
E2	A widget of medium size MUST use exactly one AA battery and have a red button on top.				
E3	It is RECOMMENDED for a widget to be waterproof. If it is not waterproof then it MUST have a warning label stating that it is not waterproof.				
E4	A widget MAY have a metallic casing. If it does have a metallic casing it MUST have a waterproof coating.				
E5	For implementations of widgets for use in the European Union a widget that weighs between 100g and 300g and is from 5 to 15 centimetres long in its longer dimension, is a medium-size widget. However, in USA the widget is medium-sized if it weighs between 4oz and 12oz and is from 2 inches to 6 inches long.				

Table 4-2. An example of Narratological Schema for the OASIS TAG Widget specification

Automatic or semi-automatic techniques may be used in this phase in order to reduce the burden of the domain experts. The particular process is also called “ontology learning” (Buitelaar et al. 2004). Classical supportive techniques are in the fields of Natural Language Processing (NLP), Artificial Intelligence (AI) and machine learning (Maedche, Staab 2000, Cimiano 2006, Wong 2009). Semi-automatic specification generation and its techniques are however out of the scope of this thesis.

Knowledge Elicitation

This phase is concerned with the conceptual modelling of a standard, which leads to the development of the baseline taxonomy of the terms used in a standard. This phase is composed of three sequential activities and the brainstorming parallel with the other three as follows:

Abstraction

The activity of *Abstraction* is to create a set of binary fact types, which can be formalised as quintuples called *lexon* in DOGMA (Spyns et al. 2002). A *lexon* is defined as $(\gamma, t_1, r_1, r_2, t_2)$, where t_1 and t_2 ($t_1, t_2 \in T$) are two terms that represent two concepts; γ is a context identifier, which points to the resources where t_1 and t_2 are originally defined and r_1 and r_2 are the relationships between t_1 and t_2 . An example of a *lexon* is $(\gamma_1, Student, studies, isStudiedBy, Book)$, which contains a fact that, within a context that is defined by γ_1 , a student studies a book, and, a book is studied by a student. A *lexon* in OntoStanD is allowed to be a quadruple or a triple as γ is optional and r_2 is preferred but not necessary.

Note that in the activity of *Compile Baseline Taxonomy*, the pair (r_1, r_2) is specified as $(is, supertypeOf)$. In the activity of define relationships, the pair (r_1, r_2) will be specified with domain relationships. In this activity, the set T is

defined and refined. Table 4-3 presents the lexon table for the TAG widget specification extract.

t1	r1	r2	t2
Widget	hasBattery	isBatteryOf	Battery
Widget	hasButton	isButtonOf	Button
Widget	hasCasing	isCaseOf	Case
Widget	hasShape	isShapeOf	Shape
Widget	hasLable	isLableOf	Lable
Widget	hasWeight	isWeightOf	Weight (int)
Widget	hasLength	isLengthOf	Length (int)
Widget	hasWaterproofCoating		Waterproof (Boolean)
Length	is-a	superTypeOf	Dimension
AA_battery	is-a	superTypeOf	Battery
Rectangular	is-a	superTypeOf	Shape
Circular	is-a	superTypeOf	Shape
Button	is-a	superTypeOf	Red_Button
Red_Button	is-a	superTypeOf	Button
Large	is-a	superTypeOf	Size
Medium	is-a	superTypeOf	Size
Small	is-a	superTypeOf	Size
Warning Label	is-a	superTypeOf	Label
...

Table 4-3: TAG Widget Lexon Table

Compile Baseline Taxonomy

Based on the output from the activity of abstraction, a taxonomy of terms in the domain is compiled in this activity. A baseline taxonomy contains only subtype relations represented by $(\gamma, t_1, is-a, superTypeOf, t_2)$ in the lexon table. Brainstorming and negotiation also support this activity. Figure 4-6 presents the baseline taxonomy of the TAG Widget specification extract, which is derived from the lexons defined earlier in Table 4-3 in conjunction with brainstorming between the researcher and the domain experts. All the relationships appearing in Figure 4-6 are subtype ('is-a' in the lexon table) relationships. Note that the types highlighted in grey ellipses are the ones that appear in the lexon table, while the others are added to the model as a result of brainstorming.

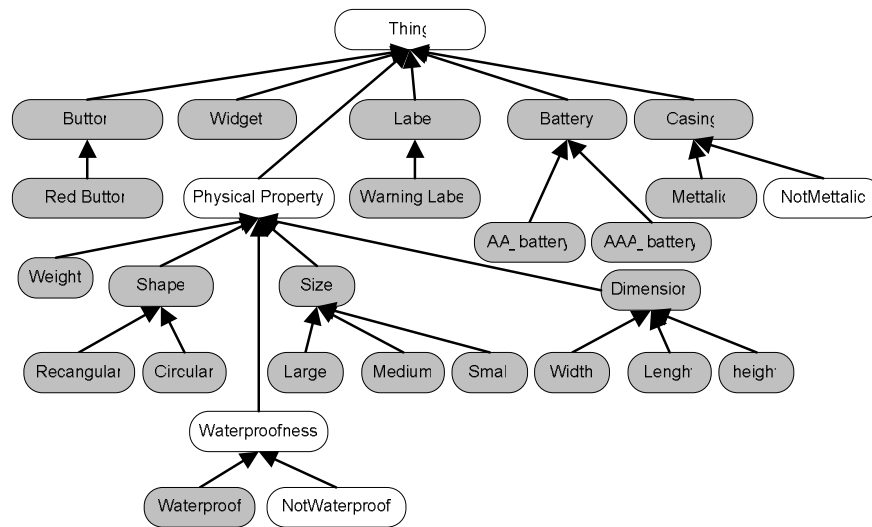


Figure 4-6. TAG Widget Baseline Taxonomy

Define Relationships

The subtype relationships were defined during the *Compiling Baseline Taxonomy* activity. At this stage, the remaining relationships between the terms are added to the ontological model of a standard. They include:

- Mereology/Aggregation relations (whole-part relation): 'part-of'/'has'.
- Domain relationships: already defined as roles (r_1, r_2) in the lexon base.
- Annotation relationships, if any.

These relationships are already captured in the lexon table as presented in Table 4-3 and should be refined and defined in an ontological manner. Figure 4-7 depicts relationships of the 'Widget' class with three other classes. Note that no restrictions on these relationships are defined yet.

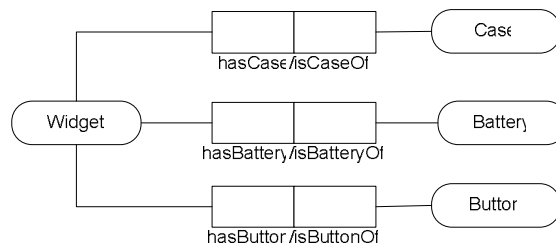


Figure 4-7. Three of TAG Widget relationships

These relationships should be further defined in an ontology language such as OWL. Figure 4-8 depicts a snapshot of the defined classes and relationships in Protégé.

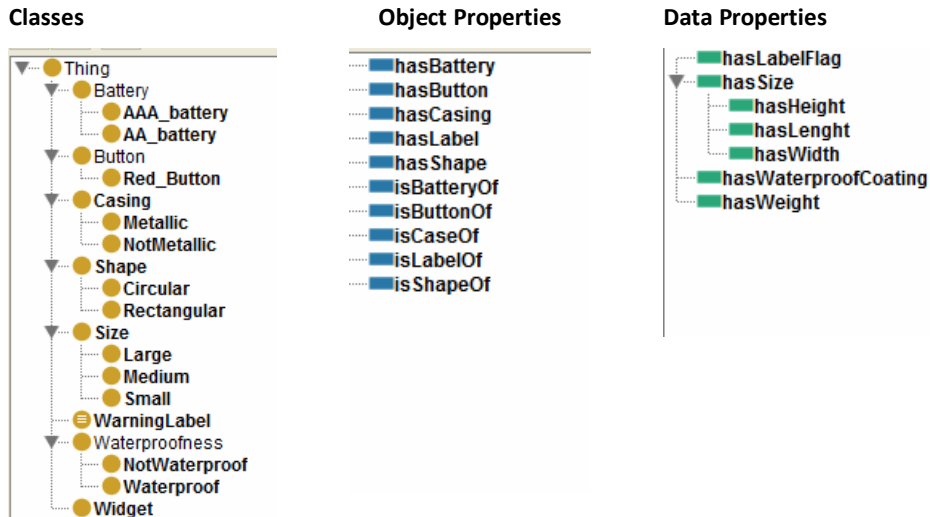


Figure 4-8. OASIS TAG Widget specification extract classes and relationships defined in Protégé

At the end of this activity the ontology base layer is completed and is formalised using an ontology modelling language such as OWL.

Brainstorming & Negotiation

As noted earlier, standard development is a collaborative process and therefore brainstorming and negotiation is an integral part of the process, which usually takes place during the teleconferences and face to face (F2F) meetings. Brainstorming (Osborn 1993, Faure 2004) is a classical method that supports a decision group to find solutions for a problem. Having a formal, clear and unambiguous base for any domain makes the process of brainstorming more meaningful and productive as each participant has a shared understanding of the domain.

When conflicts arise during the brainstorming sessions in the teleconferences or F2F meetings, the process of *negotiation* (Churchman 1995) takes place. A brainstorming and negotiation cycle of capturing concepts for building an ontology is suggested in De Moor et al. (2006). If there is a conflict, e.g., stakeholder A considers 'Medium' as a subtype of 'Size' but stakeholder B considers it as a subtype of 'Shape', the meaning negotiation process is required. A democratic vote or hierarchical decision is considered as suitable options to be taken in case of any conflicts. Democratic voting, suggests that the solution with majority votes is selected, while during the hierarchical decision, the key domain expert has to look at the conflicting points and select the solution which he thinks is more appropriate. Democratic voting is a well known and well practiced process in standards groups and is normally preferred over hierarchical decision making.

4.6.5 Ontology Validation and Verification

After development of the ontological model of a standard, it needs to be evaluated in order to ensure that the model developed by different domain experts is consistent. According to Gomez-Perez (2001) ontology evaluation is composed of two activities of ontology verification and validation. Ontology verification is mainly concerned with the syntactic correctness of the model and cleanness of ontology. Verification of an ontology can be done by utilising ontology development tools which provide verification facilities, such as Protégé.

Validation, on the other hand, refers to the process of ensuring that an ontology corresponds to the part of the real world that it is supposed to represent. Various automatic techniques could be used to validate an ontology, however, OntoStanD doesn't recommend a specific technique, but suggests that in addition to any technique used, the ontology to be validated with complete

human intervention, e.g. TC members and the knowledge engineer in an iterative manner using a set of criteria until consensus is reached. The identified factors for evaluating ontology-based conceptual models of standards in Chapter 3, are recommended to be used at this stage for evaluating the quality of the ontology.

Table 4-4 provides a summary of the key outputs of each phase of OntoStanD (V1.0).

OntoStanD Phase	Output
Formulate Vision & Feasibility Study	Proposal
TC Formation	The TC
Preparation & Scoping	A set of relevant passages and resources
Domain Conceptualisation: Knowledge Breakdown	NSs, Segmented and highlighted passages
Domain Conceptualisation: Knowledge Elicitation	Ontology base layer
Ontology Verification and Validation	Evaluation/Consensus report

Table 4-4. Summary of key output of each phase of OntoStanD (v1.0)

4.7 Evaluation of Iteration 1

In this section the artefacts of Iteration 1 of this research are evaluated. This iteration mainly provided a methodological approach for developing ontological conceptual models for a standard. The method is further applied to the OASIS TAG Widget Specification extract. This iteration has resulted in the artefacts depicted in Table 4-5, which need to be evaluated accordingly as explained in Chapter 3.

Design Research Facet	OntoStanD Interpretation
Kernal/utility theory	Ontological approach – view that clear articulation of things, relationships and axioms improves quality.
Model	OASIS TAG Widget Ontology base
Method	OntoStanD V 1.0

Table 4-5. Artefact to be evaluated at the end of Iteration 1

Consistency of the Model

The consistency of the resulting ontological model (the ontology base layer) in this stage was checked manually and automatically using the Protégé reasoner. Both forms of check showed the model to be consistent.

Completeness of the Model

The TAG Widget ontological conceptual model (ontology base layer) is considered to be complete with respect to Wand and Weber (1993) criteria - the one to one mapping between the real world and designed ontological phenomenon. The completeness may be examined by querying the ontology for any concept in the specification, which is considered as simple competency questions test.

As an example, imagine Section 100 of the TAG Widget specification (A widget **MUST** be of rectangular shape). The developed ontological model so far defines that a Widget has a shape, but doesn't limit it to only rectangular. In other words, no commitment is defined for this relationship. In this case if a Widget is defined to have a Circular shape, which is a wrong expression with respect to the Widget specification, no inconsistency would be picked up by the reasoner. This could be tested by running a very simple formal competency question, asking if there are any widgets with their shape as circular as follows:

Widget and hasShape some (Circular)

The above query is expressed in DL query, which is explained in detail in Chapter 5 . If the result set for the above query is not null, meaning that there are one or more Widget(s) with a shape other than Rectangular, then it means the model is not correct with respect to the standards' normative rules. However, this behaviour is expected to happen in an ontological model without a commitment layer (which defines the normative rules and statements), as the

ontological base layer is indeed supposed to have minimum commitment. Thus, the fact that there is an instance for a Widget of shape other than Rectangular in the ontology base of the Widget specification doesn't make the ontological model wrong, but incomplete with respect to its normative statements. All these implications show that since no commitment layer is defined for the ontological model, the TAG Widget ontological model is not considered as complete in terms of its commitment layer and needs further work. This is undertaken in the next iteration.

Usability of the Model

As discussed in Chapter 3, usability of a model is thoroughly examined only when it is used in its community. However, competency questions may be used for a limited usability test of a model. Therefore no decision could be taken in this regard at this stage.

Customisability of the Model

The developed ontological model is highly customisable at this stage since no commitment layer is defined for it and therefore the model has the minimum ontological commitment. Defining more rules and axioms would indeed hinder the customisability of a model but add to its completeness. However, the fact that there is a separation between ontology-base and commitment layer of an ontology in OntoStanD (the double articulation), significantly increases the customisability of any model as one ontology base layer can have various commitment layers in various situation and contexts. It also makes it more practical and manageable for future maintenance and also for standards profiling.

Practical adequacy of the Method

The intended purpose for designing ontology-based standards was defined as: (a) a more accurate conceptual model of a standard, (b) ability to cover normative statements of a standard in an ontological manner and on top of the defined ontological conceptual model.

The first part of the defined purpose is materialised and in Iteration 1 (OntoStanD V1.0): It is utilised and also intersubjectively intelligible and acceptable. However, the second part is not as OntoStanD V1.0 does not provide any guidelines for defining the normative statements of a standard, which are indeed a fundamental part of standards. It could be concluded that OntoStanD V1.0 designed in this iteration, partially meets the meta-requirements, but not fully and therefore is not fully fit for its intended use and therefore not as practically adequate. The model so far could be used for standards interoperability and ontology mapping and matching. However, for the specification to be reflected in the ontological model, its normative statements should be models in an ontological manner. In other words another iteration is required for providing a set of guidelines for defining the commitment layer on the ontological model of standards. This is done in the next chapter.

4.8 Summary

This chapter started by discussing the benefits of ontology-based standards and their importance for semantic interoperability. It further discussed the importance of community based knowledge management for standards development and the role of ontologies in achieving successful collaborative standards development. DOGMA ontology engineering methodology was reviewed in more detail and selected to be used and extended by OntoStanD.

OntoStanD was subsequently presented as a methodological approach for designing and developing ontology-based standards. OntoStanD was further applied to a simple pseudo specification of OASIS TAG TC, the Widget specification, as a working example and also for the purpose of evaluation. Lastly the resulting artefacts of this iteration were evaluated and justified with respect to the evaluation factors defined in Chapter 3. The OntoStanD V1.0 presented in this chapter provided guidelines for developing conceptual models of standards, which in other words forms the ontology base layer for standards.

CHAPTER FIVE: OntoStanD

5.1 Introduction

The Iteration 1 of the research resulted in the first version of OntoStanD, which provided a methodological approach for designing ontological conceptual models of standards - the ontology base layer. The methodology in Iteration 1 however, did not provide any guidelines for modelling normative statements of a standard, which are the core of each standard - the commitment layer. This chapter extends OntoStanD by providing guidelines for the above, while applying them to the OASIS TAG Widget specification and its ontology base layer created in Iteration 1.

This chapter is organised as follows. Section 5.2 presents an extended version of OntoStanD and also its application on the OASIS TAG Widget specification extract. It also discusses the Open World Assumption and the way it affects the ontology-based standards development and conformance checking. Section 5.3 evaluates the resulting artefact of Iteration 2 and finally Section 5.4 summarises this chapter.

5.2 Extended OntoStanD

OntoStanD V1.0 assists in achieving a robust foundation for capturing the vocabulary in a domain and the relationships between them, which could be unambiguously shared and understood. This, potentially, benefits the standards community by eliminating the misunderstandings and inconsistencies in terms

of the concepts and their relationships. However, important parts of each standard, the normative statement, are still not bound to the above ontological model of the standards in *OntoStanD V1.0*. Doing so would be of great advantage for the standards community as it can improve the consistency and correctness of a standard, their conformance checking and result in improved interoperability as a consequence.

As briefly noted in Chapter 4, the ontology double articulation principles divide an ontology to two layers: ontology base and the commitment layers. The ontology base layer is mainly concerned with the domain axiomatisation, and the commitment layer is about application/contextual axiomatisations (Jarrar, Meersman 2008). This means that multiple commitment layers can (re)use the same ontology base layer in various contexts and applications. The double articulation principle increases the usability and re-usability of an ontological standard model as it allows different parties to use a standard's ontology base with various ontological commitments which they may require. It also allows easier interoperability between the standards implementations which are built on the same ontological base even if they use different ontological commitments.

This iteration provides guidelines for adding the commitment layer to the ontology base layer. As the main aim of the commitment layer is defining a set of axioms and rules for constraining the usability of lexons (Jarrar, Meersman 2008), it is utilised in *OntoStanD* for defining the normative statements of standards, which indeed are a set of rules and restrictions on the base layer of any standard. Various commitments layers may be defined during the process of axiomatisation for a standard's ontology base to be used in various contexts, or to model various conformance profiles of a standard. This assists in covering all the normative aspects of a specification, which may later be used for the purpose of implementation and conformance checking.

Iteration 2 of this research extends OntoStanD V1.0 by adding a set of methodological guidelines for covering the normative statements in the commitment layer of a standard. This provides a great step forward in standards development as no approach exists at the moment for formalising the normative statements of a standards and more importantly binding them to the standard's conceptual models. This chapter also briefly discusses specification reification and the approval and maintenance phases of standards development. The extended and complete overview of OntoStanD is presented in Figure 5-1 and explained in the subsequent subsections.

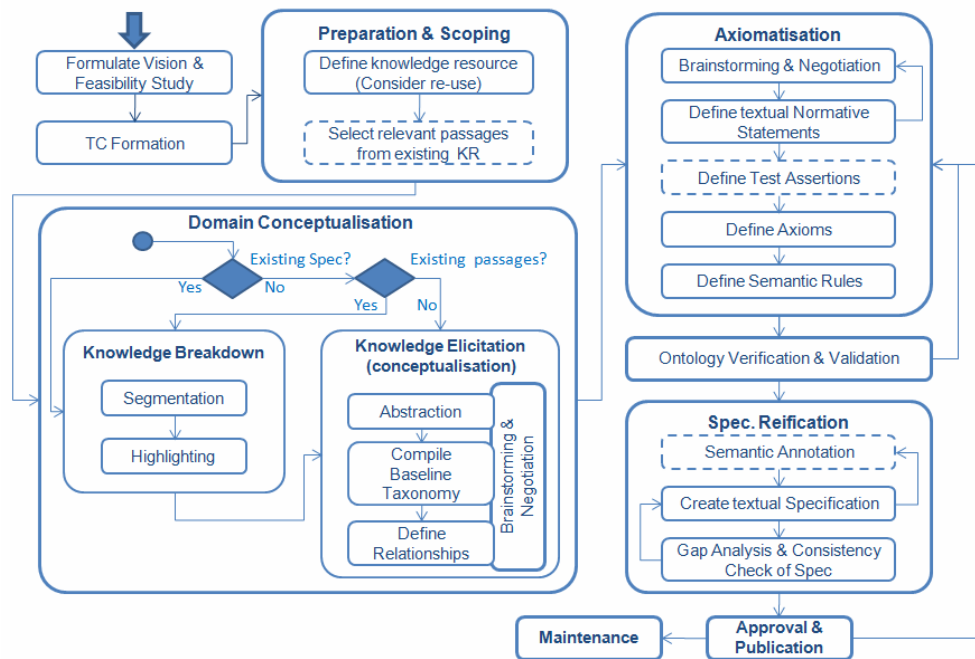


Figure 5-1. An overview of the OntoStanD methodology

5.2.1 Axiomatisation

In the *Axiomatisation* phase the normative statements of a standard are defined and modelled in a semantically enriched manner. This is an important step in gluing the normative statement of a standard to its conceptual model and is when the ontology of a standard is enhanced by adding axioms and rules to the

ontology base. This phase forms the commitment layer of a standard's ontology, where the axioms and rules – the normative statements of a standard – are added to the ontology and results in a complete ontological model of a standard. The first activity in this phase is brainstorming and negotiation as explained in Chapter 4. At this stage, brainstorming and negotiation are mainly undertaken for defining the normative statements of a standard or refining the existing ones, if a new version of a standard is being developed.

Define Textual Normative Statements

Normative statements define the prescriptive requirements on a normative source (OASIS TAG TC 2011), which are distinguished by the use of the reserved keywords such as **MUST**, **SHOULD** and **MAY** as explained in detail in Chapter 2. In this activity the normative statements of a specification should be defined on a textual basis. Brainstorming and negotiation are integral part of this activity and these two activities repeat until the final agreement on the normative statements between the members of the TC is reached. These normative statements will later be formalised in a semantically enriched manner. If the ontology is being developed for an existing standard or if an earlier version of a standard exists, there may be existing normative statements at this stage. In this case, this activity may not be performed.

Define Test Assertions

A test assertion is defined as “a testable or measurable expression for evaluation the adherence of an implementation (or part of it) to a normative statement in a specification” (OASIS TAG TC 2011 pp. 9). Test assertions are a means of formulating the normative statements, with the aim of facilitating test case development and automatic conformance testing. Figure 5-2 shows how test assertions relate to different parts of a specification and conformance testing.

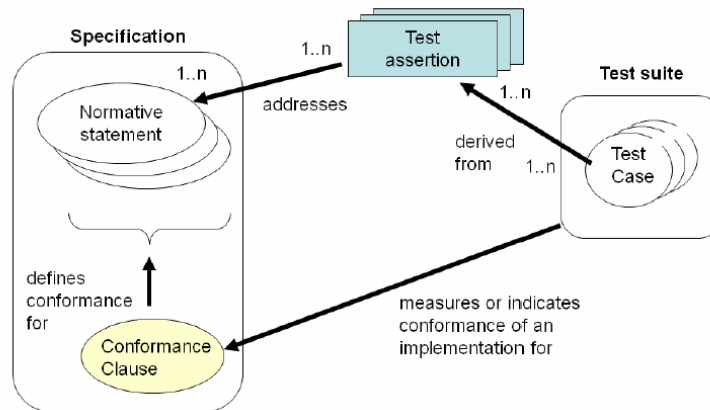


Figure 5-2. How test assertions relate to specification and testing, adopted from (OASIS TAG TC 2011)

Test assertions aim at providing a better understanding of what is expected from an implementation to conform to a standard. They also express a more precise knowledge of testing conditions, which could be used as a blueprint for developing test suite (OASIS TAG TC 2011).

Following test assertion guidelines, *OntoStanD* suggests defining test assertions for the standards specifications, which may be used as a guideline for defining the axioms and rules of the commitment layer. This activity however, is an optional step in *OntoStanD*, since the axioms could be defined directly from the normative statements. Test assertions are the furthest standardisation communities have gone to formalising normative statements and are not in any way bound to the conceptual models of standards, even if they (conceptual models) exist in one shape or form. Having test assertions however, would facilitate defining axioms and rules in the subsequent activities, as they give a more structured format to normative statements.

Table 5-1 below provides test assertions for the OASIS TAG Widget specification as presented in Chapter 4.

Test Assertions
<p>TA id: widget-TA100-1 Normative Source: specification requirement 100 Target: widget Predicate: [the widget] is of rectangular shape Prescription Level: mandatory</p>
<p>TA id: widget-TA101-1a Normative Source: specification requirement 101, part 1 Target: medium-size widget Predicate: [the widget] uses exactly one AA battery. Prescription Level: mandatory</p>
<p>TA id: widget-TA101-1b Normative Source: specification requirement 101, part 2 Target: medium-size widget Predicate: [the widget] has a red button on top. Prescription Level: mandatory</p>
<p>TA id: widget-TA102-1 Normative Source: specification statement 102, part 1 Target: widget Predicate: [the widget] is waterproof. Prescription Level: preferred</p>
<p>TA id: widget-TA102-2 Normative Source: specification statement 102, part 2 Target: widget Prerequisite: (widget-TA102-1 = false) Predicate: [the widget] has a label warning that it is not waterproof. Prescription Level: mandatory</p>
<p>TA id: widget-TA103-1 Normative Source: specification statement 103, part 1 Target: widget Predicate: [the widget] has a metallic casing. Prescription Level: permitted</p>
<p>TA id: widget-TA103-2 Normative Source: specification statement 103, part 2 Target: widget Prerequisite: widget-TA103-1 Predicate: [the widget] has a waterproof coating over its metallic casing. Prescription Level: mandatory</p>
<p>TA id: widget-TA104-1 Normative Source: specification requirement 104 Target: widget Predicate: [the widget] weighs between WEIGHT-A and WEIGHT-B. Prescription Level: mandatory Tag: normative_property = medium-sized</p>
<p>TA id: widget-TA104-2 Normative Source: specification requirement 104 Target: widget Predicate: [the widget] is from LENGTH-A to LENGTH-B long in its longer dimension. Prescription Level: mandatory Tag: normative_property = medium-sized</p>

Table 5-1. TAG Widget test assertions as defined in (OASIS TAG TC 2011)

Define Axioms

Normative statements form the core of a specification. Therefore, it is essential for each statement to be clear, concise, and unambiguous (OASIS 2007a). The better the normative statements are formalised, the easier and more accurate the conformance test and also implementation would be. *OntoStanD* suggests modelling normative statements in an ontological manner, so that they would benefit from greater stability and also better connection to the conceptual model of standards. This results in a more stable standard as a whole and more accurate conformance checking.

Normative statements and their test assertions can be considered as logical statements with a prescription level. In ontological terms normative statements can be considered as restrictions and rules, imposed on defined terms and relationships. *OntoStanD* utilises a combination of logical axioms and semantic rules for modelling normative statements of a standard specification. Both semantic rules and axioms contain rules. *OntoStanD* recommends using axioms as a default option for defining normative statements to keep the ontology as simple as possible and use rules for more complex normative statements as explained later in the chapter.

Not all the normative statements in a specification are of the same behavioural type. After studying various standards specifications - i.e. ebBP (OASIS 2006), TAG, ebXML Core Components (UN/CEFACT 2003), ebXML Reg/Rep (OASIS 2005), Greek e-GIF (Greek e-GIF 2010), WS-BPEL (OASIS 2007b) , the following three categories are defined for normative statements:

- **Static:** These statements represent well-formedness rules and relationships between entities. An example would be ‘a Widget **MUST** be of rectangular shape’.

- **Dynamic:** These normative statements are mainly concerned with state management and transitions. An example from the ebBP specification (OASIS 2006) is ‘A Receipt Acknowledgement (if required) MUST always occur before an Acceptance Acknowledgement’.
- **Run-time:** These should be handled by the implementation machine, e.g. BSI (Business Service Interface) in case of ebXML. An example is ‘If a Business Transaction fails then it is null and void and each partner MUST terminate and release any shared statement established by the transaction’. This is run-time because an implementation of the BSI should respond to this.

Interestingly the OASIS Technical Advisory Board (TAB) has also come up with a set of categories for conformance targets of a specification, which more or less correspond to the above categorisation, as follows (OASIS TAB 2011):

- **Data artefacts:** These are specified as formats and syntaxes. Examples of these are text documents, templates, messages, textual representations of process definitions, interfaces definitions, configuration definitions. This corresponds to *OntoStanD*’s Static normative statements.
- **State machines:** These are specified as logical workflows of tasks or events, or transition rules for states. Implementations of these are processes and procedures, exchange protocols between parties, transactions. This corresponds to *OntoStanD*’s Dynamic normative statements as defined earlier.
- **Processors:** These are specified as behaviours and interfaces for engines that process data artefacts and/or execute state machines. Implementations of these are workflow engines, document processors, message handlers. This also corresponds to *OntoStanD*’s Run-time normative statements.

OntoStanD recommends that the static and dynamic normative statements must be covered in the ontological model of a standard. The run-time normative statements however, do not necessarily need to be modelled in the ontological model as they should to be taken into consideration by the software developers only for implementation purposes. Nevertheless the specification should clearly distinguish between these types so that the developers would know which ones are explicitly targeting the implementation of a standard. Note that all the normative statements in the TAG Widget specification extract are considered to be static.

On Axioms Representation

According to Jarrar (2005), axiomatisation of any domain and application model cannot be a list of terms and roles and their implicit meanings but need to be represented by means of a formal language. Two primary choices of formal languages may be considered for formal representation of axioms: (1) languages with focus on expressiveness and reasoning power, e.g. description logic based languages such as OWL, (2) languages with focus on knowledge representation and retrieval, e.g. database driven languages such as ORM-ML, which is based on ORM. The choice of language for axiomatisation depends on the application scenario and perspectives for developing the ontology. For example description logic based languages, such as OWL, are more suitable for reasoning application scenarios, where expressiveness and reasoning power matters. For database and XML based application scenarios knowledge representation languages, e.g. ORM, seem more suitable as they are comprehensive in their treatments of data sets integrity (Jarrar 2005).

DOGMA, same as OntoStanD is a language neutral methodology. It however, proposes and utilises ORM and ORM-ML for ontology development. Based on

the Object Role Modelling methodology (Halpin 2001), Spyns et al. (2008) propose to use the following semantic constraints for defining axioms:

- Uniqueness constraint; e.g. if a uniqueness constraint is applied on the lexon (γ , *Widget*, *hasShape*, *isShapeOf*, *Shape*), it means that each *Widget* has AT MOST ONE *Shape*.
- Mandatory constraint; e.g. if a mandatory constraint is applied on the above lexon, it means that each *Widget* has AT LEAST ONE *Shape*, or each *Widget* MUST have a *Shape*. If we merge the two above constraints, then we will get a fact that each *Widget* has EXACTLY ONE *Shape*.
- Other constraints, which can be used to model axioms and semantic rules, are subset, cardinality, value range, set member, external mandatory, external uniqueness, inclusive-or and exclusive-or. Logical operators, such as XOR connector, negation, necessity (its model is equivalent to a mandatory constraint), possibility, conjunction (its model is equivalent to an inclusive-or constraint) and disjunction (its model is equivalent to an exclusive-or constraint). Refer to (Tang, Meersman 2009) for the details.

OntoStanD leaves it to the standards developers to choose the formalisation language based on their requirements and preferences. However, due to the importance of expressiveness and reasoning power in OntoStanD, popularity of OWL and the fact that it is an expressive ontology definition language with strong reasoning capabilities and also it being a W3C standard, there is a great tendency for using OWL as an ontology definition language in the ontology community. OWL is considered a de-facto standard for ontology development and thus special attention is paid to this language in this research. Accordingly in the remainder of this thesis OWL is used as the default ontology

representation language and all the examples and guidelines provided in this research are based on OWL.

The above ORM constraints can be presented by OWL constraints, or any other language. If OWL is chosen, the following semantic constraints for defining axioms could be used to model the above (Motik, Patel-Schneider, P. F., Parsia, B. 2009):

- Existential constraint; which describes classes of individuals that participate in at least one relationship along a specific property to individuals of a specific class (Horridge 2011). For example if an Existential constraint is applied on the lexon (γ , *Widget*, *hasLabel*, *isLabelOf*, *Label*), it means that each *Widget* has at least one *Label* or each *Widget* MUST have a *Label*. The Open World Assumption of OWL, however, affects such an expression to be used as an integrity constraint as explained later in this section.
- Universal constraint; which describes classes of individuals that for a given property only have relationships along this property to individuals that are members of a specific class (Horridge 2011). For example, if a Universal constraint is applied on the lexon (γ , *Widget*, *hasCase*, *isCaseOf*, *Metallic*), it means for all *Widgets*, if they have a *Case*, it (the case) should be *Metallic*.
- Cardinality constraints, which describe the class of individuals that have at least, at most or exactly a specified number of relationships with other individuals or data type values (Horridge 2011).
- Other constraints, which can be used to model axioms, such subclass, disjoint, equivalent class and logical constraints such as Intersection, Union and Complement.

Open World vs. Closed World Assumption

In addition to using OWL as an expressive ontology definition language, in many cases, e.g. in the case of ontology-based standards, it is desirable to use OWL for data validation and constraint checking. However, specific aspects of OWL's semantics make it difficult, if not impossible, to use it for data validation and constraint checking. These aspects are OWL's Open World Assumption (OWA) and the lack of Unique Name Assumption (UNA) (Sirin et al. 2008).

Under the OWA a statement cannot be inferred to be false on the basis of a failure to prove it (Sirin et al. 2008). In other words under the OWA, everything is possible in an empty ontology. For example, if it is defined in an ontology that each *University* MUST have at least one *Student*, but no *Student* is assigned to a specific *University* in the instance level, the OWL reasoner infers that there is a *Student* for that *University* but it is not known. Under the Closed World Assumption (CWA) however, this statement would be considered as false. Under the OWA it is assumed that the information in the knowledge base is incomplete by default, which is useful for reuse, as less commitment makes it easier to use an ontology in another context. In this situation, unless there exists a statement explicitly indicating that an instance of a *University* does not have any students, the reasoner would interpret as 'it has but is not known'.

The lack of the UNA causes an OWL reasoner to treat two resources with different identifiers as the same in specific circumstances. For example if it is defined in an ontology that each *Student* MUST be enrolled in only one *University* and at the instance level *Bahareh Heravi* is assigned to both *Brunel University* and *National Technical University of Athens*, a reasoner would infer that *Brunel University* and *National*

Technical University of Athens are referring to the same thing: two different names for the same *University*.

Both the OWA and the CWA are suitable for certain circumstances, and although the OWA of OWL makes it more difficult to understand and use in certain applications, the expressive power of OWL makes it an appealing choice for ontology definition and also to use it as an expressive schema language. To be able to use OWL as a powerful knowledge representation language as well as a constraint language for data validation, it is necessary to combine open world reasoning and closed world constraint checking (Sirin et al. 2008). In other words it is desirable to adopt the OWA when it is considered that our knowledge of the domain is incomplete and adopt the CWA otherwise, in particular when data validation and constraint checking are required.

In the standards world it is necessary to be able to close some parts of the world since one of the main objectives of ontologies developed for standards is consistency checking, which indeed needs to be done under the CWA. Sirin et al. (Tao et al. 2010) propose Integrity Constraint Checking for OWL in order to overcome the above problem and therefore to be able to use OWL for constraint checks in addition to its strong modelling and reasoning capabilities. While demonstrating various methods for Integrity Constraints (ICs), Sirin, Smith and Wallace (2008) extend OWL to accommodate integrity constraints and further developed an Integrity Constraints Validator (ICV) by extending the Pellet reasoner. This new functionality is expected to be included in the upcoming release of Pellet 3. Pellet ICV provides an alternative semantics for OWL axioms so that they are interpreted under CWA, if required, and a weak form of UNA: if two individuals are not inferred to be the same, then they will be assumed to be distinct.

Sirin et al. (2008) have adopted a rule base approach, through integration with logic programming which provide Negation as Failure (NaF) under the CWA. In a nutshell their approach is to translate OWL ICs to queries (SPARQL in this case) and whenever the answer to the query is not empty, it is concluded that the IC is violated. For example consider the following:

```
(University, hasStudent, isStudentOf, Student)
Individual: Brunel   Types: owl:University
('Brunel', hasStudent, isStudentOf, 'Bahareh Heravi')
```

In the above example *Bahareh Heravi* is not explicitly defined to be a *Student*. However, the reasoner would infer this fact if *Student* is defined as a range for the *hasStudent* class. Even if it is not defined so, no inconsistency would be detected. With the IC semantics, the above range axiom can be treated as a check rather than an inference rule. In that case, the result would be a violation of rules because *Bahareh Heravi* is not defined as a *Student*. For this to be possible the new upcoming Pellet translates the above statements to SPARQL queries and if the answer to query is not empty, it means there is an inconsistency. The following would be the query which would run in the background in the case of our example:

```
ASK WHERE {
    ?x :hasStudent ?y .
    NOT EXISTS { ?y rdf:type :Student . }
}
```

The answer to this query is *Bahareh Heravi* and therefore an inconsistency is generated as *Bahareh Heravi* is not defined as a *Student*. While *OntoStanD* strongly supports the work of Sirin et al. (2008) and seeks to utilise and integrate it when the stable version of Pellet 3 is released, this research has used a combination of rules and manual negation for the purpose of closing the world when needed. This is explained in the next section.

The TAG Widget Axiomatisation

Returning to the TAG Widget example, the following tables (Table 5-2 to Table 5-6) demonstrate how TAG Widget normative statements are modelled in an ontological manner. All the axioms are modelled using OWL syntax.

Normative Statement	Section 100 A widget MUST be of rectangular shape.	
Test Assertion	TA id	widget-TA100-1
	Normative Source	specification requirement 100
	Target	widget
	Predicate	[the widget] is of rectangular
	Prescription Level	mandatory
Axiom	Widget hasShape some Shape	
	Widget hasShape only Rectangular	

Table 5-2. OASIS TAG Widget TA for Section 100

The above table (Table 5-2) denotes that ‘a widget must firstly have a shape, and the shape must be rectangular. The first axiom of Table 5-2 implies that if a *Widget* isn’t defined to have a *Shape* the reasoner is expected to note an inconsistency. The second axiom further indicates that if a *Widget* is defined to have a *Shape*, it has to be *Rectangular*. However, because of the OWA, if no shape is defined for a *Widget*, a normal OWL reasoner wouldn’t pick up an inconsistency. An OWL reasoner however, would pick up an inconsistency if a *Widget* is defined to have a *Circular* shape. Note that *Circular* and *Rectangular* are defined as disjoint classes. The remaining axioms are presented in the next section along with their defined semantic rules.

Define Semantic Rules

In *OntoStanD*, semantic rules may be used to formalise the more complex normative statements. *OntoStanD* recommends using semantic rules when it is necessary to model normative statements which include conditional statements. It is also used here for overcoming the issues with the Open World Assumption where it causes problems for conformance checking. This is used as an

alternative to the ICVs explained above. In the following tables (5-3 to 5-6) the ontological representation of the remaining normative statements of the TAG Widget specification extract are presented and discussed.

Normative Statement	Section 101 A widget of medium size MUST use exactly one AA battery and have a red button on top.	
Test Assertion	TA id	widget-TA101-1a
	Normative Source	specification requirement 101, part 1
	Target	medium-size widget
	Predicate	[the widget] uses exactly one AA battery.
	Prescription Level	Mandatory
Axiom	Medium hasBattery exactly 1 AA_Battery	
SWRL Rule	Widget(?w), hasBatteryFlag(?w, false) -> hasBatteryFlag(?w, true)	
Test Assertion	TA id	widget-TA101-1b
	Normative Source	specification requirement 101, part 2
	Target	medium-size widget
	Predicate	[the widget] has a red button on top.
	Prescription Level	mandatory
Axiom	Medium hasButton exactly 1 Red_Button	
SWRL Rules	Widget(?w), hasButtonFlag (?w, false) -> hasButtonFlag(?w, true) AA_battery(?b), Red_Button(?rb) , Widget(?w), hasBattery(?w, ?b), hasButton(?w, ?rb) -> Medium(?w)	

Table 5-3. OASIS TAG Widget TA for Section 101

As per above normative statement (100), if a Medium Widget is defined with a battery other than AA_Battery or a button other than Red_Button (the author has ignored 'on top' here), an inconsistency would be picked up by an OWL reasoner. No inconsistency would, however, be detected under the OWA if no battery or button is defined for a widget. If more than one battery is assigned to a medium sized widget, and the two individual batteries are defined as different individuals, an inconsistency would again be picked up. If the two

batteries are not defined as different individuals, the reasoner would infer that *AA_Battery1* and *AA_Battery2* are the same battery. The same applies for *Red_Button*.

If an inconsistency error is considered to be necessary for when no battery or button are assigned to a widget, more manual work is needed. For this purpose, a True/False flag should be defined for simulating the NaF in OWL. To do so, a flag property called *hasButtonFlag* should be defined for a button to be assigned to a widget, and another one for a battery called *hasBatteryFlag*. These two flag properties are defined as functional OWL data properties, should manually be set to false as a default option and must be manually set to True when a battery or a button is assigned to a widget. The default option cannot be assigned by rules since there is no order for running rules and even if the value is set to 'True' by one rule it would again be set to 'False' by the other rule which defeats the purpose. The flag property has to be defined as a functional property so that the reasoner wouldn't allow a *Widget* to have two flag properties: one True and one False. Then another rule is added as *Widget(?w) , hasBatteryFlag(?w, false) -> hasBatteryFlag(?w, true)*. This rule forces the reasoner to throw an inconsistency if the *hasBatteryFlag*'s value is False. Since it is set to False by default and force to True, an inconsistency would be picked up. When set to True no inconsistency would be detected by the reasoner.

There is a considerable amount of manual work involved for the above to be done. While proposing the above guideline as a means for closing the world, *OntoStanD* suggests sticking to the wrong assignments (e.g. two batteries instead of one) rather than checking the existence of relationships between individuals of classes.

Normative Statement	Section 102 It is RECOMMENDED for a widget to be waterproof. If it is not waterproof then it MUST have a warning label stating that it is not waterproof.	
Test Assertion	TA id	widget-TA102-1
	Normative Source	specification statement 102, part 1
	Target	widget
	Predicate	[the widget] is waterproof.
	Prescription Level	preferred
Axiom	Widget hasWaterproofCoating only boolean	
SWRL Rules	Widget(?w), hasWaterproofCoating(?w, true) -> Waterproof(?w)	
	Widget(?w), hasWaterproofCoating(?w, false) -> NotWaterproof(?w)	
Note: The above rules are here as a mean to close the part of the world and to distinguish between waterproof and not waterproof widget. This is, however, not necessary to do so.		
Test Assertion	TA id	widget-TA102-2
	Normative Source	specification statement 102, part 2
	Target	widget
	Prerequisite	(widget-TA102-1 = false)
	Predicate	[the widget] has a label warning that it is not waterproof.
	Prescription Level	Mandatory
Axioms	Widget hasLabel some WarningLabel Widget hasLabelFlag only boolean	
SWRL Rule	NotWaterproof(?w) -> hasLabelFlag(?w, true)	

Table 5-4. OASIS TAG Widget TA for Section 102

In this normative statement (102) the situation is similar to the previous one (101). However, the specification itself is enforcing a negation in combination with a conditional statement. Therefore, the above pattern needs to be followed for modelling the negation as failure (NaF) in the condition, hence to be able to move on to the consequent part of a conditional statement. To do this, firstly we need to be able to distinguish between a waterproof and a not waterproof widget. To distinguish between waterproof and not waterproof widgets, two classes are defined as subclasses of coating as *Waterproof* and *NotWaterproof*. A rule determines that a widget is waterproof if it has *WaterproofCoating* as true and not waterproof if it doesn't (*hasWaterproofCoating=false*). It is not

necessary to define waterproof and not waterproof classes as the same thing could be done by just using the *hasWaterproofCoating* property. However, in this case the author believed that it would be easier to distinguish them as waterproofness is used in more than one instance in the specification.

An important difference between this normative statement (102-1) and the previous one (101) is in their prescription levels: ‘preferred’ and ‘mandatory’ respectively. Due to waterproofness being recommended, no inconsistency is required to be picked up if a Widget is not Waterproof. The preferred statement is modelled using the existential constraint: *Widget hasLabel some WarningLabel*. This implies that a Widget should have a Warninglabel, but no inconsistency is necessary to be picked up if it is not waterproof (as it is a ‘preferred’ and not ‘mandatory’ statement).

However, we need to make sure that the model covers the mandatory warning label as noted in the normative statement. For this purpose, a new functional data property is defined as *hasLabelFlag*, which should be set to False as a default. A rule is defined as *NotWaterproof(?w) -> hasLabelFlag(?w, true)* to denote that a widget must have a warning label, if it is not waterproof. The flag property should be changed to True when a label is assigned to a widget. With the above rule and the default value an inconsistency would be detected if a widget is not waterproof but doesn’t have a warning label or if the warning label is assigned to the widget but the flag property is (by mistake) not set to True.

Normative Statement	Section 103 A widget MAY have a metallic casing. If it does have a metallic casing it MUST have a waterproof coating.	
Test Assertion	TA id	widget-TA103-1
	Normative Source	specification statement 103, part 1
	Target	Widget
	Predicate	[the widget] has a metallic casing.

	Prescription Level	Permitted
Axiom	Widget hasCasing only Casing	
Test Assertion	TA id	widget-TA103-2
	Normative Source	specification statement 103, part 2
	Target	Widget
	Prerequisite	widget-TA103-1
	Predicate	[the widget] has a waterproof coating over its metallic casing.
	Prescription Level	Mandatory
Axiom	Widget hasCasing only Casing	
SWRL Rule	Widget(?w), Metallic(?c) , Widget(?w) , hasCasing(?w, ?c) , hasWaterproofCoating(?w, false) -> hasWaterproofCoating(?w, true)	

Table 5-5. OASIS TAG Widget TA for Section 103

In this example in a situation where a specific widget has a metallic casing but not a waterproof coating (*hasWaterproofCoating=false*), an inconsistency should be picked up.. This is modelled using the *hasWaterproofCoatingFlag* defined above (which should be defined to False by default) and defining a rule denoting that if there is a metallic casing for a widget, then it has to have waterproof coating (or it is inconsistent): *Metallic(?c), Widget(?w), hasCasing(?w, ?c), hasWaterproofCoating(?w, false) -> hasWaterproofCoating(?w, true)*. However, since the first part of the statement is a ‘Permitted’ statement, no axiom is defined for restricting the *hasMetallicCasing* property.

Normative Statement	Section 104: Localizations of Widget Size For implementations of widgets for use in the European Union a widget that weighs between 100g and 300g and is from 5 to 15 centimeters long in its longer dimension, is a medium-size widget. However, in USA the widget is medium-sized if it weighs between 4oz and 12oz and is from 2 inches to 6 inches long.	
Test Assertion	TA id	widget-TA104-1
	Normative Source	specification requirement 104
	Target	widget
	Predicate	[the widget] weighs between WEIGHT-A and WEIGHT-B.
	Prescription Level	mandatory

	Tag	normative_property = medium-sized
Axioms	Widget hasLength only integer Widget hasWeight only integer	
SWRL Rule	Together with widget-TA104-2	
	TA id	widget-TA104-2
	Normative Source	specification requirement 104
	Target	widget
	Predicate	[the widget] is from LENGTH-A to LENGTH-B long in its longer
	Prescription Level	mandatory
	Tag	normative_property = medium-sized
SWRL Rule For widget-TA104-1 and widget-TA104-2	Widget(?w), hasLength(?w, ?l), hasWeight(?w, ?weight), lessThan(?l, "15"^^integer), lessThan(?weight, "300"^^integer), greaterThan(?l, "5"^^integer), greaterThan(?weight, "100"^^integer) -> Medium(?w)	
Note: the geographical location is ignored here.		

Table 5-6. OASIS TAG Widget TA for Section 104

As a general guideline *OntoStanD* suggests that all the ‘mandatory’ constraints should be treated under the CWA. ‘Optional’ and ‘recommended’ constraints may also be interpreted under the OWA depending on the exact meaning of the normative statement. If an ontology definition tools used which only operate under the CWA, the above efforts for closing the world is not necessary. Using a CWA reasoner would eliminate reasoning power, but may still be used for data validation and consistency check. The OWA is a subject which is still under investigation and it also needs more work in relation to standards work.

5.2.2 Specification Reification

The purpose of this phase is to develop the textual specifications of a standard, which may be done manually or (semi) automatically. If done (semi) automatically, semantic annotation is required. This phase is not the main focus of *OntoStanD*, however, it is discussed in more detail in the following subsections.

Create Textual Specifications

In this activity the textual specification of a standard is developed. The specification should include all the normative statements and supporting informative material. The textual specifications of a standard can either be created manually, as it is being done currently, or (semi) automatically taking advantage of the developed ontological model as explained in the next section.

Semantic Annotation

(Semi) automatic specification creation is an optional activity in *OntoStanD*, which may or may not be required in the standards development process. However, it has the potential to facilitate specification development as well as change management. Furthermore, if the specification is created in this manner, it would be tightly bound to the conceptual model of a standard and its normative statements. In any case, it is strongly recommended that the specification of a standard should be attached to its ontological model, thus it should be semantically annotated. There are plenty of semantic annotation tools and techniques such as MnM (Vargas-Vera et al. 2001), KIM (Popov et al. 2004) and GoNTogle (Giannopoulos et al. 2010, Bikakis et al. 2010). *OntoStanD*, however, does not provide a guideline for semantic annotation at this stage; neither does it recommend a specific technique or tool.

There is also another possibility to annotate the ontology itself and generate (semi)automatic specifications from the annotated ontology. In both cases, annotated text or annotated ontology, the ontology is bound to the specification, which is an important point for the ontology-based standards.

Gap Analysis & Consistency Check of the Specification

If the semantic annotation and (semi) automatic specification creation is performed in the previous activities, at this stage the generated specification

should be checked, altered and completed manually. In some cases manual specification development may be more straightforward for standards developers as they are already used to it. In these cases, it would be a good practice to generate a (semi) automatic specification from the annotated ontology and compare it to the one which is created manually. The gaps would show whether the ontology is incomplete or perhaps the specification is inconsistent. As mentioned above, *OntoStanD* strongly recommends that the specification of a standard should be semantically annotated regardless of its manual or (semi) automatic generation.

5.2.3 Approval & Publication

As noted in Chapter 2, a formal approval process is normally conducted when the specification draft is ready. This process is usually composed of one or more rounds of technical committee reviews, followed by normally one public review round, both of which may vary depending on the type of the standard. There may be a number of iterations back to the *Axiomatisation* phase until consensus is reached on both the ontology and the specifications. At the end of this step the standard specification may be submitted to be considered for achieving ‘standard’ status. A specification is officially published as a standard if accepted as a ‘standard’; otherwise it might be published as a technical specification.

5.2.4 Maintenance

This phase is concerned with the validity and deprecation of standards and their ontological models, also revising, amendments, modifications, reaffirmation or withdrawal. How a standard is maintained may vary in each standardisation body depending on the type and scope of the standard. However, maintaining

standards which are based on ontologies is an easier and less fallible process. The reason is that all the components of a standard, from the constructs in the conceptual model to normative statements and even textual normative statements in the specification are bound to each other and changing one would (semi) automatically affect the others.

5.3 Evaluation

The first version of *OntoStanD* and the ontological conceptual model of the TAG Widget specification were evaluated in Chapter 4. The TAG Widget ontology base was considered to be consistent, clear and customisable in the previous iteration, but it was not complete. Furthermore *OntoStanD* as a method was considered as partially adequate and fit for its intended use. In this section, the model artefact of Iteration 2 of this research is further evaluated for its consistency, completeness, usability and customisability. The *OntoStanD* methodology is further evaluated for its practical adequacy. For the purpose of evaluation the OASIS TAG Widget ontology is populated by a set of artificial instances as no real data was found for this standard. The TAG Widget ontology and its instances could be found in Appendix III.

5.3.1 Consistency of the Model

Adding the axioms and rules to an ontological model can affect its consistency. Similar to Iteration 1, here the consistency of the resulting ontological model (both ontology base and commitment layers) is checked in Protégé, using the Pellet reasoner, and the model is considered to be consistent.

5.3.2 Completeness of the Model

Ontology base layer of the TAG Widget specification was considered to be complete in Iteration 1. However, since no commitment layer existed for the TAG Widget specification in the previous iteration, the ontological model was considered as incomplete in terms of its commitment layer. In this iteration however, this layer is defined and therefore the ontology is considered to be more complete. The degree of completeness of the ontological model is further tested by the means of competency question in the following section.

5.3.3 Usability of the Model

As discussed in Chapter competency questions are considered as a suitable and accepted means of evaluating usability and completeness of an ontology. In this section, therefore a set of informal simple competency questions are defined and answered by means of instantiating and querying (formal competency questions) the ontology. If the ontological model of the TAG Widget standard is capable of answering the competency questions, it is concluded that the ontology is complete, useful and adequate (Gruninger, Fox 1995). In the case of the TAG Widget specification, the ontology is populated with some artificial data, as no real data was found for this standard. As the TAG Widget is a very small and simple specification, its competency questions are a set of relatively simple questions. However, this usually is not the case and competency questions should normally be defined in a stratified manner, where higher level questions require answer to the lower level questions (Gruninger, Fox 1995). The following informal competency questions are defined for the OASIS TAG Widget standard and answered by means of formal competency questions, defined in OWL DL query language and run in Protégé using Pellet reasoner (Figures 5-3 to 5-7):

Competency question 1. Which widgets have metallic casing?

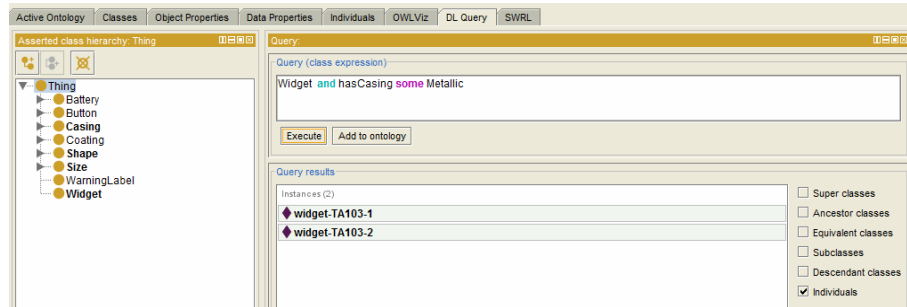


Figure 5-3. Formal consistency question (DL Query) for competency question 1

Competency question 2. Which widgets are waterproof?

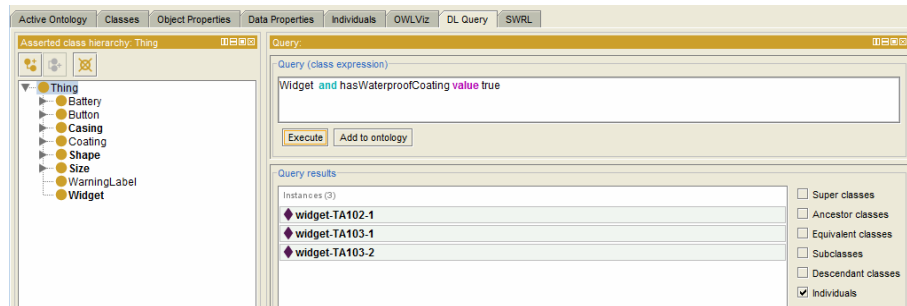


Figure 5-4. Formal consistency question (DL Query) for competency question 2

Since we defined a class *Waterproof*, with a rule denoting that if a widget has a waterproof coating then it is *Waterproof*, the above query could be as simple as the following:

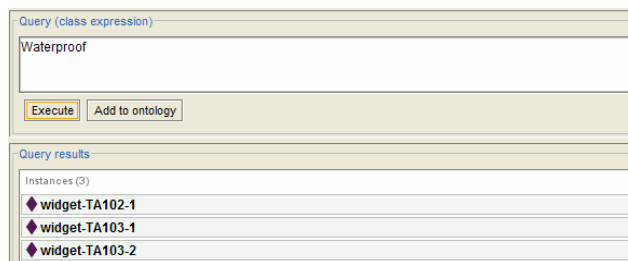


Figure 5-5. An alternative for the formal DL query for competency question 2

Competency question 3. Which widgets are metallic and are not waterproof?

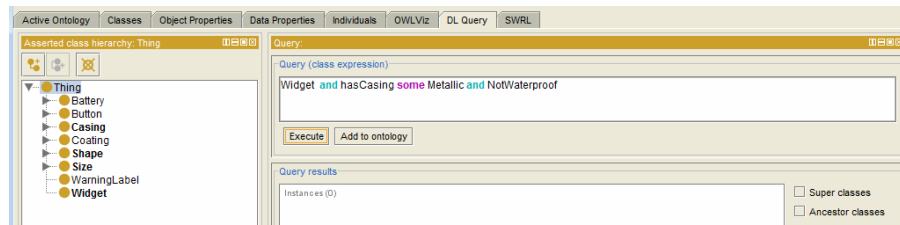


Figure 5-6. Formal consistency question (DL Query) for competency question 3

This set is and should be empty as normative statement 103 denotes if a widget has a metallic casing then it **MUST** have a waterproof coating. This means that no metallic but non waterproof widget should exist.

Competency question 4. Which widgets are considered to be medium size according to the European Union standards?

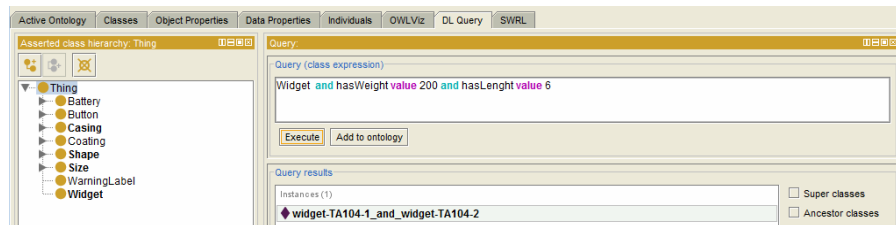


Figure 5-7. Formal consistency question (DL Query) for competency question 4

The above competency questions show that the developed TAG Widget ontology is behaving as expected and answering the defined competency questions in an appropriate manner and as expected. Therefore, we conclude that this ontological model of the TAG Widget standard is complete and usable.

5.3.4 Customisability of the Model

The ontological model in Iteration 1 was considered to be highly customisable. Adding commitment layer and more axioms and rules brings more commitment for an ontology and hence reduces its customisability. However, the fact that there is a clear separation between ontology-base and commitment layer in *OntoStanD*, the resulting models would still have a good degree of

customisability as an ontology base could be used as a base for various commitment layers in different contexts and for various conformance profiles.

5.3.5 Practical adequacy of the Method

As noted in Chapter 4, the intended purpose for designing ontology-based standards is: (a) a more accurate conceptual model of a standard, which was materialised in Iteration 1; (b) Ability to cover normative statements of a standard in an ontological manner and on top of the defined ontological conceptual model. The latter is also materialised in the final version of *OntoStanD* presented in Iteration 2. It shows that *OntoStanD* meets its meta-requirements and is fit for its intended use and therefore is considered to be practically adequate as an ontological design method for designing ontology-based standards. This however, is studied in more detail in the next chapter by applying *OntoStanD* on two other existing standards specification.

5.4 Summary

This chapter extended and complemented the *OntoStanD* methodology V1.0 by providing a set of methodological guidelines for defining the normative statements of standards as the commitment layer on top of the ontology base layer defined in the Iteration 1 of this research as presented in Chapter 4. This chapter further discussed the implication of the Open World Assumption, the effects it could have on ontological standards development and conformance checking and provided a set of guidelines for overcoming the associated problems when required. The final *OntoStanD* was applied on the TAG Widget pseudo specification, which resulted in a complete ontological model of the OASIS TAG Widget specification. The ontological model was later evaluated

for its consistency, completeness, usability and expandability and the *OntoStanD* itself was evaluated for its utility and practical adequacy.

CHAPTER SIX: Application of OntoStand

6.1 Introduction

This chapter forms the evaluation phase of this research. OntoStand is evaluated for its practical adequacy by its application on a subsection of two more existing standards: (a) The Greek e-Government Interoperability Standards, as a non-technical standard and (b) the ebXML Business Process Specification Schema as a technical, XML-based standard. Note that since both ebBP and Greek e-GIF are existing standards, the main focus of this chapter is on the *Domain Conceptualisation* and *Axiomatisation* phases of OntoStand.

The other phases are also discussed briefly as the activities which have been undertaken by standards bodies while developing these standards. The reasons for choosing these two cases were (a) direct access to the standards' specifications and developers and that (b) they are two different types of standards. The Greek e-GIF is aimed at providing interoperability between government websites but is checked by humans (at the moment). EbBP is however, a more technical, XML based business process standard, which is processed by machine. These two, in combination with the OASIS TAG Widget standard, which is a complete non-technical standard show that

OntoStanD could be used for various types of standards and in different contexts.

This chapter is organised as follows: Section 6.2 presents and discusses the application of OntoStanD on the selected part of Greek e-GIF standard. Likewise Section 6.3 examines the application of OntoStanD on the ebBP standard and provides a more extended evaluation on the resulting model. Section 6.4 discusses the practical adequacy of OntoStanD, as the main contribution of this thesis, and Section 6.5 summarises the chapter.

6.2 Application of the methodology: Greek e-GIF Scenario

In this section OntoStanD is applied to a subsection of the Greek e-Government Interoperability Framework (Please see Appendix II). The Greek e-GIF standards are written in Greek. However, only one part of it, which is used in this thesis, was translated to English for the purpose of this research. The Greek e-GIF provides standards and rules for the development and deployment of web based office systems for the Greek public administration at the local level. The aim of the Greek e-GIF standards is accelerating the development of electronic collaborations among public agencies for the delivery and secures one-stop e-Government services to businesses, citizens and other public organisations (Charalabidis et al. 2008).

The e-GIF Management Authority is the standardisation body responsible for the development and maintenance of the e-GIF. A formal standardisation process has been defined, through which new versions of the framework are drafted and published. The e-GIF displays certain characteristics associated with open standards, as drafts are publicly available and the development process involves public participation, by allowing any interested parties to

submit their proposals and provide feedback. However, development does not occur on the basis of an open decision-making procedure, as the e-GIF Management Authority is solely responsible for the approval process.

As far as the e-GIF normative structure is concerned, the e-GIF has adopted three classification levels for its rules and guidelines: Obligatory, Recommended and Under Consideration. Requirement levels are indicated by normative statements, in alignment with the Internet Engineering Task Force (IETF) Request For Comments (RFC) 2119 (IETF 1997). The Greek e-GIF is composed of six core components as follows (Greek Information Society 2009) (this reference is in Greek, English translation of the relevant parts may be found in Appendix II):

- 1) The Certification Framework for Public Administration Sites and Portals
- 2) The Interoperability and Electronic Services Provision Framework
- 3) The Digital Authentication Framework
- 4) The Documentation Model for Public Administration Processes and Data
- 5) The Interoperability Registry
- 6) The XML schema repository

In this section *The Certification Framework for Public Administration Sites and Portals* component is chosen as an evaluation scenario. For more information about the Greek e-GIF standard see (Greek e-GIF 2010).

6.2.1 Formulate Vision & Feasibility Study in Greek e-GIF

In the case of Greece, the lack of interoperability standards had resulted in small scale e-Government projects where the information and services provided were not easily accessible by, or compatible to, information systems,

technologies or business processes in other public sector organisations, or even other offices within the same organisation (Charalabidis et al. 2008). In order to address this issue, Information Society S.A., the government agency responsible for public ICT projects management under the supervision of the Greek Ministry of Interior, performed a study on the strategies deployed in other European countries in terms of interoperability and developed the proposal for the establishment of a national interoperability framework. This proposal detailed the rationale for a new standard and its expected content and application.

6.2.2 Technical Committee Formation in Greek e-GIF

In the Greek e-GIF scenario, the e-GIF Management Authority was established by Information Society S.A. through an open bidding process in March 2006. It originally consisted of members of the consortium Planet S.A., National Technical University of Athens (NTUA) and Athens Technology Centre S.A. The coordination of the project is handled by the Coordination Committee which consists of all voting members, namely the e-GIF Project Manager, a representative of Information Society S.A. and the members of the Supervision and Procurement Committee (a supervising group appointed by the contractee, as per government procurement regulations). The development group consists of the e-GIF Project Manager (the key domain expert), the Expert Group (domain experts) and four Working Groups focusing on specific aspects of the Greek e-GIF.

6.2.3 Preparation & Scoping in Greek e-GIF

As noted above, the Greek e-GIF standards are composed of 6 core components. For the purpose of this thesis one of the components namely *The*

Certification Framework for Public Administration Sites and Portals is chosen. This component provides standards and guidelines to be followed by public agencies at the central or local levels, when designing, developing or deploying e-Government portals and services. The guidelines included in this component are further categorised into eight categories: *General Principles*, *Website Management*, *Electronic Services and Interoperability*, *Assessment of a Website Content*, *Security & Privacy*, *Legal Issues*, *Website Dissemination* and *Website Structure and Organisation*. The latest of the guidelines, the *Guideline for Website Structure and Organisation*, is used as a use case in the context of the OntoStanD methodology in this section. This guideline provides standards for the structure, categorisation, and presentation of the content and services in the public administration web sites.

Define Knowledge Resources

Since there is already a complete standard defined for the Greek e-GIF, the knowledge resources are mainly based on the existing specifications of the Greek e-GIF standards. In this thesis the *Guidelines on the Structure and Organisation* of the *Certification Framework for Public Administration Sites and Portals* standard provides the main knowledge resource.

Select Relevant Passages From Existing Knowledge Repositories

Due to time and space limits, only one section of the *Guidelines on the Structure and Organisation* - Section 2: *Web Page Content* - is chosen to be used as relevant passages. This guideline is a text based semi-structured resource.

6.2.4 Domain Conceptualisation for Greek e-GIF

Since relevant knowledge resources are defined in the previous phase, the Knowledge Breakdown activity should be executed as follows.

Knowledge Breakdown

As explained in Chapter 4, Narratological Schemas (NS) are recommended to be used in at this stage with the aim of scoping the knowledge in the domain. Table 6-1 shows an example of an NS, which contains a number of basic settings and a set of episodes for KY.32 and KII.19 rules of the Greek e-GIF standards.

Title	Public Administration Sites and Portals in Greece (Structure and Content)				
Author	Bahareh Heravi and Yan Tang	ID	GreekEGov.Section.3.3.3 - Guidelines on the Structure and Content	Date	2010-12-03
Theme	Greek eGov Interoperability Framework/Certification Framework for Public Administration Sites and Portals/Website Structure and Organisation/Structure and Content				
Purpose	To describe how an e-government website in Greece can be organised				
Scope	Section 2. web page's content in doc. "Case Study on Selected Guidelines of the Certification Framework for Public Administration Sites and Portals (Guidelines on the Structure and Content of a Web Site) (section 3.3.3 of the Certification Framework)"				
Settings					
ID	Description				
S1	Website management is a set of organisational procedures for development and maintenance of a public site, as well as the proposed organizational schema of the website, roles and responsibilities for managing its content and overall functionality.				
S2	The Greek Government Category List contains the basic categories and subcategories for classifying content and services.				
Characters					
ID	Role	Description			
C1	Government/public servant	They are the actual public administrative service provider (through a public administration portal)			
C2	Citizen, Enterprise, Institution/Organization	The end users of the public administration portal			
C3	Website manager, web designer, web administrator	They are the ones who are responsible for designing and implementing the			

		public administration portal
Episodes		
E1 (KY.32)	Each public web site MUST include:	
	<ul style="list-style-type: none"> the public institution's logo and official name the public institution's services flow chart a short presentation of the public institution information on the institution's administration the legislation with which the institution's operation and competences must comply information on the services offered by the public institution and the way they are provided news and announcements 	
E2 (ΚΠ.19)	It is SUGGESTED that the web sites of Public Administration institutions include:	
	<ul style="list-style-type: none"> the terms and conditions of their use the most frequently asked questions that are submitted with regard to the services provided by the institution and the corresponding answers (FAQs) links redirecting the user to public web sites of relevant content information with regard to the number of times the web site is viewed as well as with regard to the use of the electronic services it provides 	

Table 6-1. an example of a Narratological Schema

Segmentation and Highlighting

The selected Greek e-GIF standard has a relatively formal structure and therefore the selected relevant passages are already in the form of segmented text and we move on to the Highlighting activity straightaway. Table 6-2 shows an example of highlighting of episodes E1 (KY.32) which are already defined in the previous activity, with the aim of discovering important concepts in the segmented passage.

E1 (KY.32)	Each public web site MUST include:
	<ul style="list-style-type: none"> the public institution's logo and official name the public institution's services flow chart a short presentation of the public institution information on the institution's administration the legislation with which the institution's operation and competences must comply information on the services offered by the public institution and the way they are provided news and announcements

Table 6-2 highlighted phrases for the segmentation that is the episode E1 (KY.32) in Table 6-1

Knowledge Elicitation of Greek e-GIF

Brainstorming & Negotiation

In the Greek e-GIF scenario and in the context of this thesis, two knowledge engineer, one key domain expert and two domain experts were collaborating in brainstorming sessions in both *Knowledge Elicitation* activity and *Axiomatisation* phase. The brainstorming activity of both *Knowledge Elicitation* activity and *Axiomatisation* phase started with focusing on the selected relevant passages as discussed in Chapter 4 and Chapter 5, followed by group discussion on each rule/normative statement.

Abstraction

In this activity a set of lexons are created. Following the same example shown in Table 6-1, the lexons shown in Table 6-3 are abstracted.

\mathcal{C}_1	\mathcal{R}_1	\mathcal{R}_2	\mathcal{C}_2
Public website	includesLogo	isIncludedBy	Logo
Public website	includesName	isIncludedBy	Name
Public website	includesServiceFC	isIncludedBy	Service flow chart
Public website	includesPresentation	isIncludedBy	Presentation
Public website	includesInformation	isIncludedBy	Information
Public website	includesLegislation	isIncludedBy	Legislation
Public website	includesNews	isIncludedBy	News
Public website	includesAnnouncement	isIncludedBy	Announcement
Name	isOf	hasName	Public institution
Service flow chart	isOf	hasServiceFL	Public institution
Presentation	isOf	hasPresentation	Public institution
Operation	isOf	hasOperation	Public institution
Competence	isOf	hasCompetence	Public institution
Administration	isOf	hasAdministration	Public institution
Information	isAbout	isRepresentedAs	Administration
Operation	compliesWith	isCompliedBy	Legislation
Competence	compliesWith	isCompliedBy	Legislation
Information	isAbout	isRepresentedAs	Service
Public institution	offers	isOfferedBy	Service
Information	isAbout	isRepresentedAs	Way
Public institution	provides	isProvidedBy	Way
Name	is-a	supertypeOf	Information
News	is-a	supertypeOf	Information
Presentation	is-a	supertypeOf	Information

Announcement	is-a	supertypeOf	Information
Logo	is-a	supertypeOf	Information
Service flow chart	is-a	supertypeOf	Information

Table 6-3. a lexon table that contains the lexons abstracted from Table 6-1 - E1(KY. 32)

Compile Baseline Taxonomy

Figure 6-1 shows how the terms in a standard are organised in the baseline taxonomy of the Greek e-GIF. This is derived based on the lexons defined earlier in conjunction with brainstorming. The types highlighted in grey ellipses are the ones that appear in Table 6-3 and the rest come from the remaining parts of the specification. All the relationships in Figure 6-1 are 'is-a' relationships.

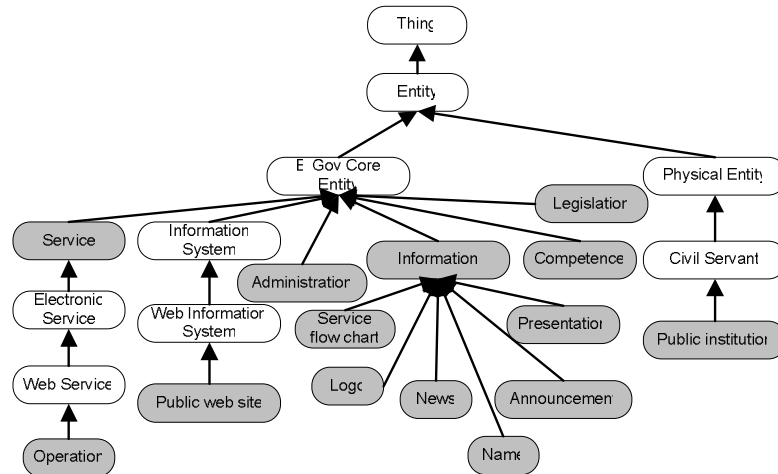


Figure 6-1. an example of type hierarchy for the terms in the Greek e-GIF standard

Define Relationships

In addition to 'is-a' relationships as defined in the taxonomy, Figure 6-2 depicts the mereological (part/whole) relations for the concepts in Table 6-3. Domain relationships, defined as roles (r₁,r₂) in Table 6-3 can also be modeled in the same way.

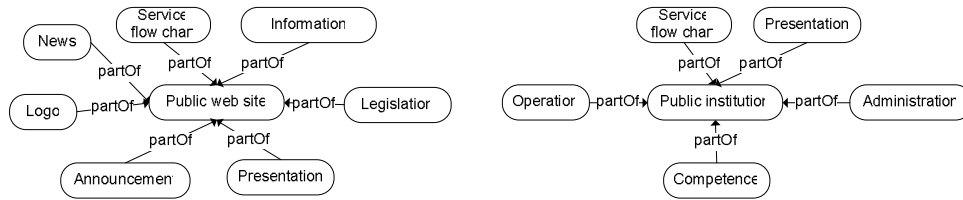


Figure 6-2 model mereological relations for the concepts in Table 6-3

An example of a domain relationship would be a lexon such as (*Public institution, Offers, Is offered by, Service*). Figure 6-3 depicts a snapshot of the above classes and relationships for the selected part of the Greek e-GIF standards in Protégé. The ontology for the selected party of the Greek e-GIF standards can be found in Appendix IV.

Classes	Object Properties
<ul style="list-style-type: none"> Thing <ul style="list-style-type: none"> Entity <ul style="list-style-type: none"> E-GovCoreEntity <ul style="list-style-type: none"> PhysicalEntity <ul style="list-style-type: none"> CivilServant <ul style="list-style-type: none"> PublicInstitution <ul style="list-style-type: none"> Information <ul style="list-style-type: none"> Announcement Logo Name News Presentation ServiceFlowChart Competence Service <ul style="list-style-type: none"> ElectronicService <ul style="list-style-type: none"> WebService <ul style="list-style-type: none"> Operation Legislation InformationSystem <ul style="list-style-type: none"> WebInformationSystem <ul style="list-style-type: none"> PublicWebsite 	<ul style="list-style-type: none"> includesName isProvidedBy provides isOfferedBy offers isCompliedBy compliesWith isOf isAbout isRepresentedAs hasAdministration hasCompetence hasOperation hasPresentation hasServiceFL hasName includesInformation includesAnnouncement includeBy includesNews includesLegislation includesPresentation includesServiceFC includesLogo

Figure 6-3. An ontology for the selected part of the Greek e-GIF standards specification defined in Protégé

6.2.5 Axiomatisation of Greek e-GIF

Define textual Normative Statements

In the Greek e-GIF scenario the normative statements are already defined in the specification and therefore we proceed to the next activity. An example of a normative statement from Table 6-2 is the following:

Each public web site **MUST** include the public institution’s logo.

Define Test Assertions

Test assertions are most useful when a test suit is to be developed. However, the Greek e-GIF standards are to be tested manually and with human intervention and therefore no automatic test suit would be used in the case of this project at the moment, thus a test assertion may not be as useful in this case. Nevertheless a test assertion for the above normative statement is as follows:

Normative Source: E1 (KY.32)/1 - Each public web site MUST include the public institution's logo.

Target = Public Website

Predicate = includes Logo.

Prescription Level: mandatory

Define Axioms

There is already a lexon defined for the concepts participating in the normative statement above as (γ , *PublicWebsite*, *includesLogo*, *isIncludedBy*, *Logo*). This normative statement is a mandatory statement, MUST, and therefore a mandatory constraint should be used to model this normative statement as presented in Figure 6-4.

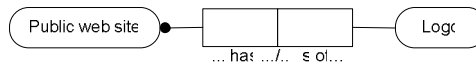


Figure 6-4. A mandatory constraint modelled in ORM

In OWL terms the above axiom would be represented using the two following axioms:

`PublicWebsite includesLogo some Logo`

`PublicWebsite includesLogo only Logo`

The above axioms denote that a public web site should include at least one logo assigned to it. Due to the open world reasoning however if no logo is assigned to a web site no inconsistency would be detected. If something other than logo is assigned to a public web site via the *includesLogo* an inconsistency would,

however, be picked up. Closing the world in this case is explained in the following section.

Define Semantic Rules

Following the rules from Chapter 5, to overcome the OWA a *includesLogoFlag* data property should be defined and manually set to false. Respectively the *PublicWebsite(?p)*, *includesLogoFlag(?p, false) -> includesLogoFlag(?p, true)* rule is added to the ontology to trigger an inconsistency if the *includesLogoFlag* is not set to true. It is important to note that even in closing the world in this way, the developers/users are still free to act under the OWA if they wish. To do so they would just avoid defining the flag property as false. In this case an OWL reasoner would automatically work under the OWA.

6.2.6 Ontology Validation and Verification of e-GIF

The ontology derived in this scenario was developed in OWL and verified using Protégé 4.0.1 and Pellet reasoner. It is further validated by the domain experts (in the National Technical University of Athens) until consensus is reached.

6.2.7 Specification Reification of e-GIF

At this stage of the project the specification is developed in a classic manner. Therefore, ontology annotation activity was not performed, but might be considered for the next versions of the same standard.

6.2.8 Approval and Maintenance of e-GIF

The Greek e-GIF gradually reached sufficient maturity to gain official recognition as a formal standard in November 2008, with the release of the 3rd version of the Framework. Compliance has become mandatory by law for all

public IT systems and services since 1/1/2009¹ and further maintenance of the specifications, as well as compliance monitoring, has become the responsibility of the Informatics Development Agency (Greek Informatics Development Agency 2010), which falls under the jurisdiction of the Greek Ministry of Interior.

6.2.9 Evaluation of the Greek e-GIF Ontological Model

As mentioned in Chapter 3, the designed ontological model of a standard should be evaluated for its consistency, clarity, completeness, customisability and usability, which are discussed in the following:

Consistency of the model

The consistency of the Greek e-GIF ontological model is checked using Protégé 4.0.1, which is also used for developing purpose and the Pellet reasoner. The Greek e-GIF ontology is tested at a Tbox (empty ontology) level since the ontology targets a manual process and no digital data currently exists which could let us instantiate the ontology.

Completeness of the model

The ontology for the Certification Framework for Public Administration Sites and Portals of the Greek e-GIF ontology is considered to be free of any construct deficits, meaning that there is a modelling construct for any concepts defined in the specification. Since the ontology is evaluated at a TBox level competency questions cannot be used for further completeness evaluation.

¹ Law No. 3731/2008.

Usability of the model

As mentioned above no instance data was available during the term of this project and therefore the ontology was evaluated at TBox level. A coherent TBox model (as evaluated above) gives a rather good indication of the practical adequacy of the method. However, it does not allow for a more coherent examination using the competency questions as no data is available to be queried. With this in mind and for a more coherent evaluation of OntoStanD, it is considered necessary to apply OntoStanD to another scenario in ABox level. This is done in the Section 6.3.

Customisability of the model

This ontology is built on top of an existing ontology which addresses the top level standards for the Greek e-GIF. Following OntoStanD guidelines this ontology is designed with the minimum ontological commitment required, which makes it easier to be expanded and customised. It is important to note that the OWA make an ontology more expandable since less inconsistencies are picked up during reasoning under the OWA. It is important to keep this in mind when closing the world for any specific axiom in the design time. This, however, does not hold if a closed world ontology design tool and reasoner is used.

6.3 Application of OntoStanD: the case of ebBP

EbXML Business Process Specification Schema (ebBP) is an XML based B2B process standard which may be used to define the public aspects of B2B processes (OASIS 2006). EbBP is one of the five components of ebXML (Electronic Business using eXtensible Markup Language) framework (Van der Eijk et al. 2001). It is a modular suite of XML based specifications, sponsored by OASIS and UN/CEFACT (United Nations Centre for Trade

Facilitation and Electronic Business), aiming at providing an open, XML based infrastructure that enables the global use of electronic business information in an interoperable, secure, and consistent manner. The other four components of the ebXML framework are as follows:

- ebXML Core Components, which provide basic and reusable building blocks for Business Documents.
- ebXML Registry/Repository (ebReg/Rep), which are as follows: EbXML Repository manages and maintains the shared information as objects in a repository. EbXML registry is an interface for accessing and discovering shared business semantics.
- Collaboration Protocol Profiles and Agreements (CPP/A), which are as follows: The CPP describes the specific capabilities that a trading partner supports. A CPA is a document that represents the intersection of two CPP's and is mutually agreed upon by both trading partners.
- ebXML Messaging Service (ebMS), which is designed for the secure, reliable exchange of e-Business information.

The ebXML framework is designed in a way that specifications of each component can be used independently, composed as desired, or integrated with other evolving technologies (OASIS 2006). In the process of developing an ontological model for the ebBP specification, a complete ontology base layer is defined for the ebBP specification, which takes into consideration the specification and the XML schema. Due to time and space limit, this thesis presents the commitment layer developed for a selected part of the ebBP specification (Section 3.4.2).

6.3.1 Formulate Vision & Feasibility Study in ebBP

The ebXML Business Process TC started its work by publishing its Call for Participation in September 2003. Prior to that, a group of interested parties had identified a need for such a standard and worked together to prepare the call for participation. The call for participation could be found at <http://lists.oasis-open.org/archives/ebxml-bp/200309/msg00000.html>, where the ebBP vision is clearly stated.

6.3.2 Technical Committee Formation in ebBP

Following the publication of the Call for Participation, an initial meeting was scheduled for the group so that interested parties would discuss their interests and the core TC was formed on the day of the teleconference.

6.3.3 Preparation & Scoping in ebBP

As noted above, the ebXML specifications are composed of five core components, each of which discussed in their own specialised TC. Therefore, the scope of the ebBP TC was on the Business Process aspect of the ebXML.

Define knowledge resources

In the scope of this thesis, since there is already a complete standard defined, the chosen knowledge resources are: a) The specifications, b) the XML Schema and c) a set of available ebBP instances. In addition to the specification and the XML Schema, and based on them, a UML diagram was drawn, which gives an overview of the top level view of the elements and their basic relationships in ebBP. This diagram is later used in the Domain Conceptualisation phase and brainstorming sessions.

Select Relevant Passages from Existing Knowledge Repositories

This activity if done before having the specification would be to choose relevant passages to develop the standard. In this chapter however, since the specification already exists and due to the space limit, one section of the specification is chosen to be formalised using OntoStanD methodology. The section to be explored in this section is Section 3.4.2 of the ebBP specifications, entitled “Business Transactions”. The ebBP specification is a text based document along with an XML schema. Therefore both structured and unstructured knowledge resources are available in this case and should be considered in the following phases.

6.3.4 Domain Conceptualisation of ebBP

The *Knowledge Breakdown* activity should be conducted based on the existing specification as follows.

Knowledge Breakdown

To process the textual specification of ebBP and on the basis of DOGMA guidelines, Narratological Schema (NS) are used at this stage to scope down the knowledge. Table 6-4 shows an NS from the Section 3.4.2 of the ebBP specification, which contains basic settings and a set of episodes for the selected part of the ebBP specification.

Title	ebXML Business Process Specification's Business Transactions				
Author	Bahareh R. Heravi	ID	ebBP.Section.3.4.2	Date	16/03/2011
Theme	ebXML Business Process Specification Schema Technical Specification v2.0.4 /Language Overview/Key Concepts of This Technical Specifications/Business Transactions				
Purpose	Articulating the Business Transactions between business partners.				
Scope	Section 3.4.2 of the ebXML Business Process Specification Schema Technical Specification v2.0.4, last updated on 21 December 2006.				
Settings					
ID	Description				

S1	A Business Transaction represents an atomic unit of work that may be associated with a trading arrangement between two business partners.	
...	...	
Characters		
ID	Role	Description
C1	Business Partner	A Business Transaction is conducted between two parties playing opposite abstract roles in that transaction.
C2	Requesting Role	The Requesting role is a placeholder for performing the Requesting Business Activity.
C3	Responding Role	The Responding role is a placeholder for performing the Responding Business Activity.
Episodes		
E1	<p>A Business Transaction MUST succeed or fail from both a technical and business protocol perspective.</p> <ul style="list-style-type: none"> • If it succeeds from both perspectives it MAY be designated as having shared intent between the two business partners, or otherwise govern their collaborative activity. • If it fails then it is null and void, and each partner MUST terminate and release any shared statement established by the transaction. • If it fails from protocol perspective, each party MUST synchronise their state to the state prior to the start of the transaction. 	
E2	A Business Failure is any Failure that is identified by an application or service during the processing of the Business Document(s) and based on information not available in or part of the ebBP instance.	
E3	<p>The Business Transaction is defined as an abstract super class. It is associated with the six concrete Business Transaction patterns defined in the UMM:</p> <ul style="list-style-type: none"> • Commercial Transaction • Information Distribution • Notification: Note, the Notification of Failure business transaction is based on the Notification pattern. • Query Response • Request Confirm • Request Response 	

Table 6-4. An example of a Narratological Schema

Segmentation and Highlighting

The ebBP specification has a relatively formal structure and therefore the selected relevant passages are already in the form of segmented text allowing a move to the *Highlighting* activity straightaway. Table 6-5 shows an example of highlighting of episodes E1 which is already defined in the previous activity, with the aim of discovering important concepts in the segmented passage.

E1	<p>A Business Transaction MUST succeed or fail from both a technical and business protocol perspective.</p> <ul style="list-style-type: none"> • If it succeeds from both perspectives it MAY be designated as having shared intent between the two business partners, or otherwise govern their collaborative activity. • If it fails then it is null and void, and each partner MUST terminate and release any shared statement established by the transaction. • If it fails from protocol perspective, each party MUST synchronise their state to the state prior to the start of the transaction.
----	---

Table 6-5. Highlighted phrases for the segmentation that is the episode E1 in Table 4

Knowledge Elicitation of ebBP

Abstraction

As mentioned earlier in addition to the textual specification, ebBP has an XML schema (a structured resource), which should be considered in the abstraction process. The *Abstraction* activity of ebBP specification therefore starts by processing the structured resources and then proceeds to the highlighted segments of the textual passages. It is important at this stage to ensure the lexon table defined in this activity covers all the concepts in the existing XML schema, if believed to be a correct model of the domain by the standards developer.

The general rule taken in processing the XML schema is to define a class for each element and each complex type in the XML schema. However, to make the ontology more meaningful, this rule is not followed for each and every construct. The ebBP schema is specified using both XML elements and complex types. The latter are hidden in an ebBP XML instance and have little or no semantic value and therefore are ignored for the ontological modelling. This will keep the ontology simple and easier to understand, while covering the semantics.

Each element in an XML schema is composed of zero or more elements and zero or more attributes. For defining the roles in the lexon table, the elements

and attributes of each entity are translated to roles/relationships. Following the OWL naming convention the name of a role in the lexons in general starts by 'has' or 'is' followed the name of the role. For example a BusinessTransaction entity in the ebBP XML schema has one or more *RequestingRole* and one or more *RespondingRole*. These two are modelled as *(BusinessTransaction, hasRequestingRole, isRequestingRoleOf, RequestingRole)* and *(BusinessTransaction, hasRespondingRole, isRespondingRoleOf, RespondingRole)* in the lexon table. OntoStand however, doesn't recommend any specific naming conventions and leaves it to the standards developers themselves.

When, and if, modelling in OWL, XML elements are defined using OWL Object properties and XML attributes are defined using OWL Data Properties. OWL supports most XML types and therefore the range of the Data Properties are generally set based on the type of XML attributes. However, similar to class definitions, some exceptions are considered in defining the Data Properties; There are attributes in the XML schema whose type is *IDREF*. Following the general rule, they should be translated to Data Properties with range *IDREF*. *IDREF* is used in XML to refer to an *ID* type defined for another element.

In the ontology design, however, the *IDREF* attributes are not necessary since we can simply define the range of an Object Property to be another class, rather than a reference to another element. For example in the ebBP XML schema, a `DocumentEnvelope` refers to a `BusinessDocument` with `BusinessDocumentRef` attribute which is of type *IDREF*. This should basically match the `nameID` of a `BusinessDocument`, which is of type *ID*. In the ontology however, the `hasBusinessDocument` property of a `DocumentEnvelope` is not defined as a Data Property of type *IDREF*, but as an Object Property with the range `BusinessDocument`. This makes reasoning over the ontology much more precise and makes more sense as the two classes have a proper relationship in the

ontology rather than being related based on string matching. In addition the 'r_{ref}' part of the property is ignored since it is referring to another class and therefore is not necessary. This also simplifies the ontology.

Following the same example shown and highlighted in Table 6-5, the lexons shown in Table 6-6 are abstracted. In the case of ebBP, the lexons are mainly derived from the specifications and later completed with respect to its XML schema. It is important to ensure that the structured and unstructured resources are covered fully and no concept is left out.

t ₁	r ₁	r ₂	t ₂
Business Transaction	hasRespondingRole	isRespondingRoleOf	RespondingRole
Business Transaction	hasRequestingRole	isRequestingRoleOf	RequestingRole
Business Transaction	hasState	Is include by	BusinessSuccess
Business Transaction	hasState	isStateOf	ProtocolSuccess
Business Transaction	hasState	isStateOf	BusinessFailure
Business Transaction	hasState	isStateOf	ProtocolFailure
RespondingRole	is-a	supertypeOf	Role
RespondingRole	is-a	supertypeOf	Role
Commercial Transaction	is-a	supertypeOf	Business Transaction
Information Distribution	is-a	supertypeOf	Business Transaction
Notification	is-a	supertypeOf	Business Transaction
Query Response	is-a	supertypeOf	Business Transaction
Request Confirm	is-a	supertypeOf	Business Transaction
Request Response	is-a	supertypeOf	Business Transaction
Success	is-a	supertypeOf	State
Failure	is-a	supertypeOf	State
BusinessSuccess	is-a	supertypeOf	Success
ProtocolSuccess	is-a	supertypeOf	Success
BusinessFailure	is-a	supertypeOf	Failure
ProtocolFailure	is-a	supertypeOf	Failure

Table 6-6. A lexon table that contains the lexons abstracted from Table 6-4

Compile Baseline Taxonomy

Figure 6-5 depicts a part of the baseline taxonomy of the ebBP specifications, which is derived based on the lexons defined in Table 6-6. The types highlighted in grey ellipses are the ones which are extracted from lexons in

Table 6-6 and the others are the ones which are related to the Business Transaction and are not mentioned in the selected subsection, but are included in other parts of the specification, XML schema, or are defined to categorise some related concept during the brainstorming sessions. All the relationships at this stage are ‘is-a’ relationships.

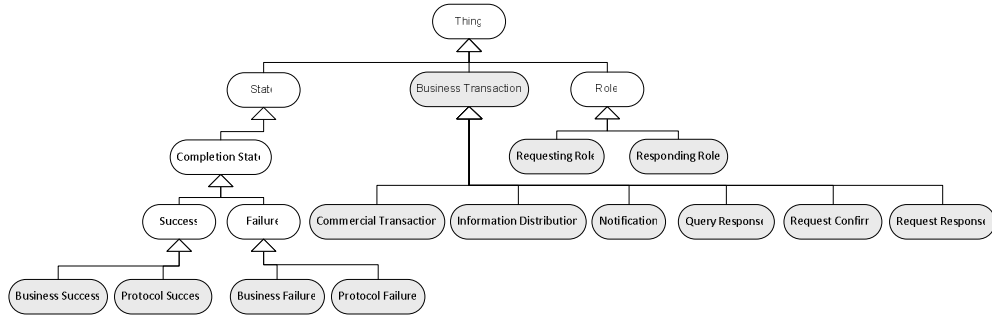


Figure 6-5. An example of type hierarchy for selected part of ebBP specification

Define Relationships

At this stage other relationships, which are defined as r_1 and r_2 in Table 6-6, and are not of type ‘is-a’ relationships, should be added to the above taxonomy. An example of such relationship, extracted from Table 6, is as follows:

(BusinessTransaction, hasRespondingRole, isRespondingRoleOf, RespondingRole)

Figure 6-6 depicts a part of ebBP ontology in three different layouts: Class Definition, Object Properties and Data Properties. The complete ebBP ontology could be found in Appendix V.

Brainstorming & Negotiation

Brainstorming is an integral part of standards development in OASIS, which normally takes place during the teleconferences or face to face meetings. The TC’s mailing lists are also a common place for brainstorming, which were used in a few occasions for brainstorming with the standards developers during the term of this research.

Class Definitions	Object Properties	Data Properties
<ul style="list-style-type: none"> Thing Attachment AttributeSubstitution BinaryCollaboration BusinessTransactionPattern CommercialTransaction BusinessAction <ul style="list-style-type: none"> RequestingBusinessActivity RespondingBusinessActivity BusinessActivityType <ul style="list-style-type: none"> BusinessTransactionActivity CollaborationActivity BusinessCollaboration BusinessDocument BusinessPartnerRole BusinessStateLinkType <ul style="list-style-type: none"> FromLink ToLink BusinessTransaction CollaborationGroup ComplexBusinessTransactionActivity ConditionExpression Decision DocumentEnvelope Documentation ExternalRoles Failure Fork Join MessageMap MultiPartyCollaboration OperationMapping OptName Package Performs ProcessSpecification Role <ul style="list-style-type: none"> RequestingRole RespondingRole Signal <ul style="list-style-type: none"> SignalEnvelopeType SignalEnvelopeType <ul style="list-style-type: none"> AcceptanceAcknowledgement AcceptanceAcknowledgementException ReceiptAcknowledgement ReceiptAcknowledgementException SignalMap Specification Start StatusVisibility Success TimeToPerform Transition 	<ul style="list-style-type: none"> BeginsWhen EndsWhen RefersToBusinessTransaction hasAttachment hasAttributeSubstitution hasBinaryCollaboration hasBusinessAction <ul style="list-style-type: none"> hasRequestingBusinessActivity hasRespondingBusinessActivity hasBusinessActivityType hasBusinessCollaboration hasBusinessDocument hasBusinessPartnerRole hasBusinessTransaction hasBusinessTransactionActivity hasCollaborationActivity hasComplexBusinessTransactionActivity hasConditionExpression hasCurrentRole hasDecision hasDocumentEnvelope hasDocumentation hasExternalRoles hasFailure hasFromBusinessState hasMultiPartyCollaboration hasOperationMapping hasPackage hasPerforms hasPerformsRole hasPostCondition hasPreCondition hasRequestingRole hasRespondingRole hasRole hasSignal <ul style="list-style-type: none"> hasSignalEnvelopeType hasSpecification hasStart hasSuccess hasToBusinessState hasVariable isBusinessActionOf <ul style="list-style-type: none"> isRequestingBusinessActivityOf isRespondingBusinessActivityOf isBusinessDocumentOf isBusinessTransactionActivityOf isBusinessTransactionOf isDocumentEnvelopeOf isPerformsOf isRealisedByBusinessTransactionActivity isRoleOf isSignalEnvelopeTypeOf <ul style="list-style-type: none"> isAcceptanceAcknowledgementExceptionOf 	<ul style="list-style-type: none"> hasAttributeName hasBinaryCollaborationTypePattern hasBusinessCollaborationPattern hasBusinessTransactionHeadPattern hasBusinessTransactionBaseTypePattern hasConditionGuard hasCurrentRoleRef hasDefaultValue hasDocumentationLang hasDuration hasExpression hasExpressionLanguage hasForkType hasInstanceVersion hasInterfaceName hasLegalIntent hasLocation hasMimeType hasName hasNameID hasOperationName hasOperationStep hasPerformsRoleRef hasRoleRef hasSpecificationVersion hasTargetNamespace hasTimeToPerform hasType hasURI hasUuid hasValue haspattern isAuthenticated isAuthorizationRequired isConcurrent isConfidential isGuaranteedDeliveryRequired isInnerCollaboration isIntelligibleCheckRequired isNonRepudiationReceiptRequired isNonRepudiationRequired isPositiveResponse isTamperDetectable waitForAll

Figure 6-6. Part of ebBP ontology in three different layouts, Class Definitions, Object Properties and Data Properties.

6.3.5 Axiomatisation of ebBP

Define textual Normative Statements

In the case of ebBP, the normative statements are already defined in the specification and therefore we proceed to the next activity. The author however, believes that the normative statements in the ebBP specification are defined poorly and are not well connected to their conceptual model. The conceptual model itself is also rather poorly described in the specification and is believed to be more relied on the XML schema. This is more evident when trying to define test assertions and formalise the normative statements as they are done with quite some difficulties and confusion. Furthermore, most of the normative statements in the ebBP schema would be categorised as ‘run-time’, which cannot be implemented in a formal manner without significant amount of

programming involved. The above may be some of the reasons why the ebBP standard was never widely used in the community.

Define Test Assertions

The following is a test assertion defined for one of the normative statements in episode E1:

```
Normative Source: E1 - A Business Transaction MUST succeed or
fail from both a technical and business protocol perspective
Target = Business Transaction
Prerequisite = (Business Transaction is executed)
Predicate = Succeed OR Fail
Prescription Level: mandatory
```

Define Axioms

There is already a lexon defined for the concepts taking part in the above normative statement and its test assertion as (*Business Transaction, hasState, isStateOf, Success*). This normative statement is a mandatory statement and an Existential constraint is suitable to formalise this normative statement as follows:

```
BusinessTransaction ((hasState some Failure) or (hasState some
Success)) and (hasState some CompletionState)
```

The above normative statement is formalised in OWL and implies that a Business Transaction has to have either at least one ‘Failure’ or at least one ‘Success’ and that it has to have a ‘CompletionState’, which itself is a union of classes ‘Success’ and ‘Failure’. In addition, it is necessary to define that ‘Failure’ and ‘Success’ are disjoint. The above normative statement is a static mandatory one and therefore should be treated under the CWA. This could be done using a combination of a flag property and rules as discussed in Chapter 5, or by using the upcoming Pellet ICV reasoner (if OWL used). In any case, the above normative statement causes an inconsistency in the ontology as depicted

in Figure 6-7. In the specification it is mentioned that Success and Failure states belong to the Business Collaboration and not the Business Transaction. While the normative statement indicates that a Business Transaction **MUST** have one of Success or Failure states. These are believed to be contradictory in the specification, which is picked up by the reasoner when an axiom is defined for the relevant normative statement.

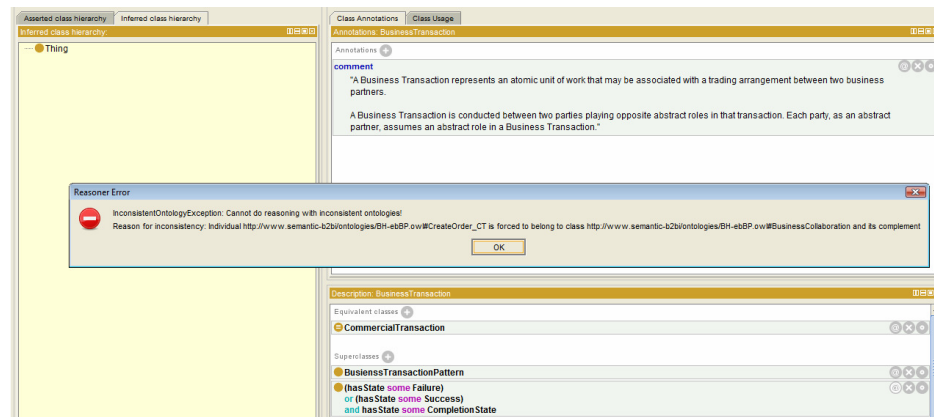


Figure 6-7. An inconsistency in the standards specification picked up by reasoner

Define Semantic Rules

As mentioned in Chapter 5, OntoStanD suggests minimum use of semantic web rules where possible. In this example, rules may be used in combination with the flag property to close the world. However, since the above normative statement is found to cause inconsistency in the ontology, there is no need to define rules to close the world. The inconsistency should be reported to the relevant TC.

6.3.6 Ontology Validation and Verification of ebBP

The above ontological model was developed in OWL and verified using Protégé 4.0.2 and is considered to be consistent after removing the violating normative statement. It is also under further validation by the domain experts in the ebCore TC.

6.3.7 Specification Reification of ebBP

At this stage of the project the specification is developed in a traditional manner. Therefore, ontology annotation activity was not performed. It, however, but might be considered for the next versions of the same standard.

6.3.8 Approval and Publication of ebBP

Approval of the ebBP specifications followed the OASIS approval process and ebBP is accepted as an OASIS standard in December 2006. EbBP is now focused on a submission to ISO for the ISO-15000. Furthermore, the ebBP ontology is currently submitted to be reviewed in the OASIS ebCore TC for future development and integration purposes and/or publication as a deliverable of the TC.

6.3.9 Maintenance of ebBP

EbBP specifications are now maintained under ebCore TC, which is responsible for maintaining all ebXML standards at the moment. The ontological model of the ebBP developed in this iteration is submitted to the ebCore TC and may be considered to be utilised for developing the next versions of the standard.

6.3.10 Evaluation of the ebBP ontological Model

The ebBP ontology is defined using OWL DL ontology language and covers both syntax included in ebBP XML schema and the semantics of the ebBP specification. Protégé 4.0.2 is used for developing the ontology, queries are written using Protégé DL query and Pellet is used as a reasoning engine. It is important to note that the ebBP ontology is fundamentally different from automatic transformation of an XML schema into OWL. The automatic transformation cannot cover the semantics embedded in both the schema and

the textual specifications and leaves some elements without any semantic value. Furthermore, it cannot cover the relationships between classes. The following is the evaluation of the ebBP ontological model with respect to the evaluation criterion defined in Chapter 3.

Consistency of the Model

The consistency of the model is checked by the help of existing tools. In the case of ebBP ontology, any inconsistency would be picked up by the reasoner. The ontology has been extensively tested both as TBox (empty ontology) and ABox (ontology with instances) and it is free of any inconsistencies after removing the normative statement which caused inconsistency.

Completeness of the Model

The ebBP ontology base is considered to be complete with respect to Wand and Weber guidelines meaning that there is a one to one relationship between the ontological constructs and the constructs defined either in the ebBP specification or in its XML schema. Competency questions were further utilised to examine the completeness in terms of the normative statements of the ebBP standard, the commitment layer, as explained in more details in the following section.

Usability of the Model

The usability of the ebBP ontological model is examined by running a set of rather simple competency questions over the ontology. The ontology is instantiated with some real data and questioned for the defined competency questions. This also reflects on the other evaluation criteria as above – in particular on the completeness of the ontology. These competency questions are designed in collaboration with the developers of the ebBP standard, who are the experts in the domain. This demonstration shows that the ontological model is

usable and complete. A subset of these competency questions are provided in this section, which are answered based on an example process introduced in the Motivating Example Section. Further evaluation remains up to the ebBP TC and practitioners.

In ebBP, a Business Process is realised by one or more Business Collaborations. Business Collaborations are composed of Business Transactions, which are expressed as exchange of Business Documents. A Business Transaction in ebBP consists of a Requesting Business Activity, a Responding Business Activity, and one or two document flows between partners. A Business Transaction may also involve the exchange of one or more Business Signals that govern the use and meaning of acknowledgements (OASIS 2006). Figure 6-8 depicts the semantics of ebBP Business Transactions.

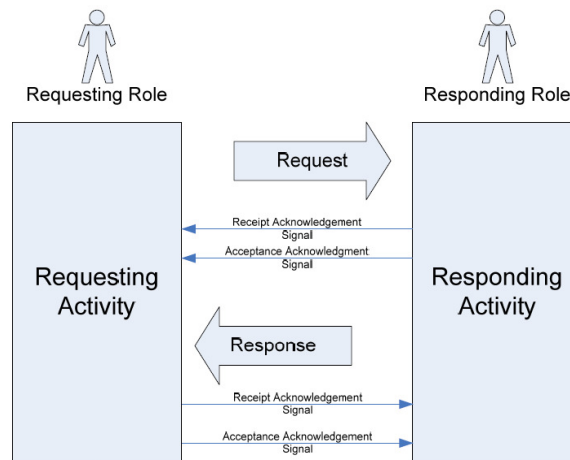


Figure 6-8. Schematic of core Business Transaction semantics, adopted from (OASIS 2006)

In B2B interactions it is usually very important to know:

- Which Business Documents are used in a particular Process?
- Which Business Documents are used in a particular Package?
- Which Collaborations in a particular Process use a Business Document with a specific target namespace?

With regard to Figure 6-8 the following questions are important to answer:

- Which signals do the transactions in a particular Business Collaboration use?
- In which transactions in a particular process does a particular party take a requesting role?

In order to answer these questions for the ebBP ontology an example B2B process is introduced in the following sections and these questions are answered in the context of this process.

Motivating example

Figure 6-9 illustrates a ‘Simple Ordering Process’, which is defined in ebBP v2.0.4 and is based on UBL (Universal Business Language) (OASIS UBL TC 2008). This process is publicly available on the OASIS UBL web site at http://docs.oasis-open.org/ubl/cs-UBL-1.0-SBS-1.0/universal-business-process-1.0-ebBP/ebxmlbp-2.0_ubl-1-order-with-simple-response-1.xml. UBL is a library of standard electronic XML business documents such as purchase orders and invoices which is developed by OASIS. This example is used throughout the rest of this chapter to present the usability and completeness of the ebBP ontology and relevant instances and also to evaluate the ontology with regard to the competency questions.

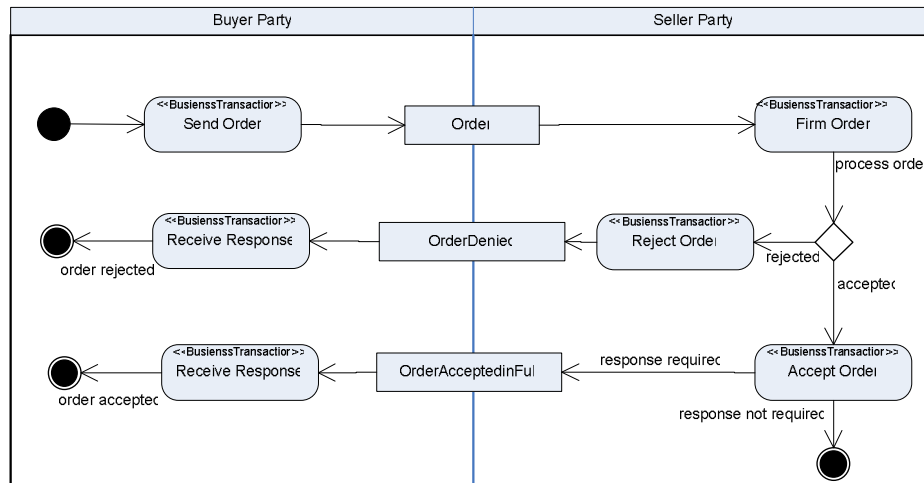


Figure 6-9. Simple Ordering Process (UBL)

With regard to the ontological representation of ebBP, one should differentiate between the representation of the business process modelling language and the representation of a specific process model. Business process modelling language constructs in an ontology can be represented by classes and properties of an ontology, while specific process models are defined as instances of an ontology. In the ebBP ontology, the language constructs are modelled using OWL and the Simple Ordering process itself and its instances are modelled as individuals of the ebBP ontology. To test the completeness and usability of the ontology the above competency questions are answered for this specific process using DL Queries over the ebBP ontology and its individuals.

The Simple Ordering process is defined as individuals of the relevant classes and their relationships in the ebBP ontology. Listing 6-1 shows two individuals of the ontology and depicts their relationship; an instance of a `RequestingBusinessActivity` and an instance of a `CommercialTransaction`.

```

Individual: SendOrder_ReqBA

Types:
  RequestingBusinessActivity,
  owl:Thing

Facts:
  hasAcceptanceAcknowledgement AA2,
  hasDocumentEnvelope Order_DE,
  hasReceiptAcknowledgementException RAE2,
  hasReceiptAcknowledgement RA2,
  hasAcceptanceAcknowledgementException AAE2,
  isIntelligibleCheckRequired "true"^^xsd:boolean,
  hasName "Send Order",
  isAuthorizationRequired "true"^^xsd:boolean,
  isNonRepudiationReceiptRequired "true"^^xsd:boolean,
  hasNameID "SendOrder_ReqBA",
  isNonRepudiationRequired "true"^^xsd:Boolean
=====

Individual: CreateOrder_CT

Types:
  CommercialTransaction,
  owl:Thing

Facts:
  hasRequestingRole OrderInitiator,
  hasRequestingRole OrderResponder,
  hasRespondingBusinessActivity FirmOrder_ResBA,
  hasRequestingBusinessActivity SendOrder_ReqBA,
  hasName "Create Order",
  hasNameID "CreateOrder_CT"

```

Listing 6-1. Two individuals of ordering process in the ebBP Ontology

Competency Questions in the Context of the Simple Ordering Process

With regard to the Simple Ordering process, there are several key 'drill-down' type knowledge questions which are important to answer. In this section, the competency questions are answered in the context of the Simple Ordering process. Each competency question is answered using a DL Query provided in the following.

Competency question 1. Which Business Documents are used in the Simple Ordering process?

```
BusinessDocument and isBusinessDocumentOf some
  (ProcessSpecification and hasNameUuid value
  "bpid:urn:oasis:names:draft:bpss:ubl-2-order-with-simple-
  response-process-2")
```

Listing 6-2. DL Query for competency question 1

The result of this query (Listing 6-2) should be and is: `orderAcceptanceFull_BD`, `Order_BD` and `OrderDenied_BD`. As seen in Listing 6-2 the *isBusinessDocumentOf* Object Property is used for querying the ontology to answer competency question. This property is the inverse property of *hasBusinessDocument*. Without having this inverse property answering this question would not be possible when there is more than one process defined in the knowledge base. This competency question shows how important inverse properties are for drill down queries. Inverse properties are used in most of the competency questions discussed in this scenario.

Competency question 2. Which Business Documents are used in Package "OrderWithSimpleResponse"?

```
BusinessDocument and isBusinessDocumentOf some
  (Package and hasNameID value "OrderWithSimpleResponse")
```

Listing 6-3. DL Query for competency question 2

The result of this query (Listing 6-3) should be and is: `OrderAcceptanceFull_BD`, `Order_BD` and `OrderDenied_BD`.

Competency question 3. Which Signals do the transactions in collaboration "Create Order" use?

```
Signal and isSignalOf some
  (SignalEnvelopeType and isSignalEnvelopeTypeOf some
    (BusinessAction and isBusinessActionOf some
      (CommercialTransaction and isBusinessTransactionOf some
        (BusinessTransactionActivity and
isBusinessTransactionActivityOf some
          (BusinessCollaboration and hasNameID value
"CreateOrder_BC"))))))
```

Listing 6-4. DL Query for competency question 3

The result of this query (Listing 6-4) should be and is: ra2, aa2, aae2 and rae2.

Competency question 4. Which collaborations in the Simple Ordering process use the Business Document whose target namespace is "urn:oasis:names:specification:ubl:schema:xsd:OrderResponseSimple-2"?

```
BusinessCollaboration and
  (hasBusinessTransactionActivity some
    (BusinessTransactionActivity and
      refersToBusinessTransaction some
        (BusinessTransaction and hasBusinessAction some
          (BusinessAction and hasDocumentEnvelope some (DocumentEnvelope
and hasBusinessDocument some (BusinessDocument and
hasSpecification some (Specification and hasTargetNamespace
value
"urn:oasis:names:specification:ubl:schema:xsd:OrderResponseSimple-2" ^^ anyURI ))))))))
and (BusinessCollaboration and
isRealisationOfProcessSpecification some
  (ProcessSpecification and hasUuid value
"bpid:urn:oasis:names:draft:bpss:ubl-2-order-with-simple-
response-process-2")
```

Listing 6-5. Optimised DL Query for competency question 4

The result of this query (Listing 6-5) should be and is CreateOrder_BC. The object property *hasBusinessAction* is defined as a super property of *hasRequestingBusinessActivity* and *hasRespondingBusinessActivity* in the ebBP

Ontology. Furthermore, they all have inverse properties called *isBusinessActionOf*, *isRequestingBusinessActivityOf* and *isRespondingBusinessActivityOf* respectively. This allows the competency questions to be answered. Additionally if the super property did not exist, the query in Listing 6-5 would have been as Listing 6-6.

```

BusinessCollaboration and
  (hasBusinessTransactionActivity some
    (BusinessTransactionActivity and refersToBusinessTransaction
      some (CommercialTransaction and
        (hasRequestingBusinessActivity some
          (RequestingBusinessActivity and hasDocumentEnvelope some
            (DocumentEnvelope and hasBusinessDocument some
              (BusinessDocument and hasSpecification some
                (Specification and hasTargetNamespace value
                  "urn:oasis:names:specification:ubl:schema:xsd:OrderResponseSimple-2" ^^ anyURI )))))))) or
BusinessCollaboration and
  (hasBusinessTransactionActivity some
    (BusinessTransactionActivity and refersToBusinessTransaction
      some (CommercialTransaction and
        (hasRespondingBusinessActivity some
          (RespondingBusinessActivity and hasDocumentEnvelope some
            (DocumentEnvelope and hasBusinessDocument some
              (BusinessDocument and hasSpecification some
                (Specification and hasTargetNamespace value
                  "urn:oasis:names:specification:ubl:schema:xsd:OrderResponseSimple-2" ^^ anyURI )))))))) and
BusinessCollaboration and isRealisationOfProcessSpecification
  some (ProcessSpecification and hasUuid value
    "bpid:urn:oasis:names:draft:bpss:ubl-2-order-with-simple-response-process-2")

```

Listing 6-6. Not Optimised DL Query for competency question 4

It is clear that although it would have been possible to answer the competency question 4 without optimisation, the query would have been longer and less clear. This was achieved with a simple super property added to the ontology.

These added semantics are only achievable through systematic engineering of the ontology and impossible with automatic transformation of a XML schema to an ontology.

Competency question 5. In which transactions in the Simple Ordering process does the "Buyer" party take a requesting role?

```
CommercialTransaction and
  isBusinessTransactionOf some
    (ProcessSpecification and hasUuid value
      "bpid:urn:oasis:names:draft:bpss:ubl-2-order-with-simple-
      response-process-2")
  and
    isRealisedByBusinessTransactionActivity some
      (BusinessTransactionActivity
        and hasPerforms some
          (Performs and hasCurrentRole some
            (Role and hasName value "Buyer")
            and hasPerformsRole some
              (RequestingRole and
                hasNameID value "OrderInitiator"))))
```

Listing 6-7. DL Query for competency question 5

Competency question 5 is basically addressing the relation between Business Transactions and Business Transaction Activities, which are their realisation and their corresponding roles. The result of this query (Listing 6-7) should be and is `CreateOrder_CT`.

In the case of ebBP, and since there is an XML schema for the specification, the above competency questions could be answered using XPath (W3C 1999) and XQuery (W3C 2010), which are techniques for navigating and querying XML documents respectively. These techniques may be considered as an alternative to ontological querying over the XML documents (and not the ontology-based standards). They, however, they do not solve the problem with the lack of semantic support and lower quality conceptual models fro standards as they query the non-ontological models.

Customisability of the Model

Following OntoStanD guidelines, the ebBP ontology is designed with minimum ontological commitment required, which makes it easier to expand and customise.

6.4 Evaluation of the Method

As explained in Chapter 3, the main aim of evaluation in this thesis is to evaluate the practical adequacy of OntoStanD as a design process in terms of its fitness for use. For this, a set of artificial evaluations were conducted by the application of OntoStanD to the TAG Widget specification, presented in Chapter 4 and Chapter 5, and also selected parts of the Greek e-GIF and the ebBP standards as presented and evaluated in this chapter. The practical adequacy of OntoStanD is measured by: (a) Determining if the method meets its intended purpose which is providing methodological approach for achieving a high quality, machine process-able, conceptual model of a standard: this was materialised in Iteration 1 of this research. (b) Ability of OntoStanD to cover normative statements of a standard in an ontological manner and on top of the defined ontological conceptual model: this was also materialised in the Iteration 2 of this research. Chapter 3 determined that if the resulting models of a design method are of high quality, it could be inferred that the method used for modelling itself is a usable and practical one. After applying OntoStanD on three varied standards and evaluating the resulting models as high quality conceptual models, here, it is concluded that OntoStanD itself is fit for its intended use and practically adequate.

6.5 Summary

This chapter evaluated the practical adequacy of OntoStanD by its application on subsections of two existing standards: (a) The Greek e-Government Interoperability Standards, as a non-technical standard and (b) the ebXML Business Process Specification Schema as a technical, XML-based standard. The selected part of the Greek e-GIF standards is ontologised and evaluated as a T-Box model since it is a manually processed data and, therefore, no digital data were available for this standard. Although the Greek e-GIF scenario was considered as a suitable partial evaluation of OntoStanD, a more thorough evaluation, was required and therefore another evaluative iteration was conducted by applying OntoStanD on the OASIS ebBP standard at A-Box level. These two scenarios, in addition to the OASIS TAG Widget scenario presented in the last two chapters, are argued to provide a coherent evaluation of OntoStanD and its application in relation to various types of standard. The three scenarios further demonstrates that not only OntoStanD is capable of capturing the domain knowledge and turning them into a robust ontology-based model, but also it matches very well with the standards communities' best practices and activities.

CHAPTER SEVEN: Conclusion & Future Research Directions

7.1 Research Summary

Information standards play important role in achieving seamless interoperability. These standards, however, are mainly focused on the syntactic interoperability and little attention is paid to the semantic aspects of the concepts and their relationships – which eliminates the power of standards for achieving semantic interoperability. Standards development is a community based activity where consensus between stakeholders is crucial. Standards organisations usually provide a set of guidelines for standards development. As argued in Chapter 2, these guidelines are not comprehensive enough, particularly when it comes to the details of the specification development. In other words, there is lack of practical, technological and semantic support for developing the conceptual model and content of a standard specification. This research proposed that semantic web technologies should be used in the process of standards development and, also, as a basis for defining standards' conceptual models and normative statements. Considering standards and ontology as important factors for achieving interoperability, the work here presented a methodological approach for addressing the shortcomings of current standards and their development processes. The methodology – OntoStanD – is

targeted at developing ontology-based standards, with the aim of achieving higher quality standards and a higher degree of semantic interoperability.

Chapter 2 critically reviewed the three intersecting fields of study necessary for this research: semantic interoperability, standards and ontology. The aim of this review was to attain a deep understanding of the state of the art in the above fields and comprehending the ways in which ontology may facilitate semantic interoperability in conjunction with standards. The literature review provided an insight into the ways ontology and standards have been used for achieving inter-organisational interoperability alongside work that has used ontology for formalising standards in one way or another. The review uncovered a lack of robust guidelines for designing the conceptual models and normative statements of standards and, in a broader sense, for designing high quality interoperability information standards. It also led to an understanding of the potential benefits of ontology in achieving inter-organisational semantic interoperability and their usefulness in the process of standards development and conceptual modelling.

Chapter 3 set out the means for achieving the aim and objectives of this research by adopting the design science research method. DSR provides a means for engaging in design problems via proposing guidelines for designing novel solutions to the defined problems. The chapter presented design science research, the rationale behind its selection, its steps, artefacts and evaluation methods. To choose the most suitable design research method, various DSR methods were studied, analysed and compared. Following DSR guidelines, this research was undertaken in an iterative manner, where each iteration built upon the outcome of its previous iteration. Accordingly, this research was composed of two build iterations and an evaluation iteration. The ultimate artefact of this research is the OntoStanD methodology and the result of the application of OntoStanD on various domains is a set of ontological models. A utility theory was also defined as an underpinning rationale behind designing the method.

Chapter 4 presented the first iteration of this research: It started by investigating the benefits of ontology-based standards for semantic interoperability. Those benefits formed the foundation of a utility theory, proposing that ontology-based standards are a desirable approach for achieving semantic interoperability and that ontology should be utilised in the process of standards development. The chapter further introduced OntoStanD V1.0 as a methodological approach for designing and developing ontology-based standards, with respect to the utility theory. OntoStanD V1.0 was further applied to the OASIS TAG Widget specification extract (OASIS TAG TC 2011), as a working example. The main artefact of the Iteration 1 of this research presented in this chapter were the OntoStanD V1.0 and the ontological conceptual model (ontology base layer) of the OASIS TAG Widget specification, which were evaluated accordingly.

Furthermore, the community based nature of standards development was discussed as an important aspect in Chapter 4, which should be taken into account while proposing a new standards development methodology. Thus, the DOGMA ontology engineering methodology (Spyns et al. 2008), which is a community-based ontology engineering method, was selected as the basis for conceptual modelling in OntoStanD and was discussed in detail. Using DOGMA as a basis for OntoStanD, to a good degree, guarantees meeting the six principles of business semantics management in knowledge intensive communities as proposed by De Leenheer (2009).

Chapter 5 demonstrated the second iteration of this research. This iteration extended and refined OntoStanD V1.0 presented in Chapter 4 by adding a set of guidelines for defining the normative statements of standards on top of the defined ontological conceptual model. In other words it provided detailed guidelines for axiomatising the ontological model previously defined. The Open World Assumption was also discussed as a characteristic which could affect, and in some cases cause, unwanted behaviour in the process of standards

development and consistency checking. A set of guidelines were proposed for overcoming the associated problems in necessary circumstances.

The extended OntoStanD was applied on the TAG Widget specification for adding the normative statements of the specification on top of the developed conceptual model. The artefacts of this iteration were the final version of OntoStanD and the complete ontological model of the TAG widget, including its ontology base and commitment layer, which were evaluated accordingly.

Chapter 6 presented the evaluation iteration of this research with the aim of demonstrating the utility and practical adequacy of OntoStanD. This was achieved by applying OntoStanD on two other scenarios: (1) A subsection of the Greek e-Government Interoperability framework (Greek e-GIF 2010) and (2) a subsection of the OASIS ebBP specification (OASIS 2006). The main focus of this chapter was on the *Domain Conceptualisation* and *Axiomatisation* phases of OntoStanD, which are its key phases and could commonly be used for developing new standards, new versions of standards and also formalising existing standards. The reason for choosing these two cases were (a) good access to the standards' specifications and also standards developers and that (b) they are two completely different types of standards, while both target interoperability. These two scenarios, in addition to the OASIS TAG Widget scenario, provided a thorough evaluation of OntoStanD and demonstrated that it could be utilised for developing ontological models for various types of standards. They also demonstrated that OntoStanD matches well with the standards activities and practices in place.

Table 7-1 presents how various chapters of this thesis addressed the objectives of the research defined in Chapter 1.

Objective	Chapter	Outcome
Obj. 1 - Investigate the structure of standards and the existing standards development processes.	CH2	Studying the characteristics of various standards, their structure and various standards development processes.
Obj. 2 - Investigate various ontology engineering methods	CH2	Studying various ontology engineering methods and comparing them in a chronological manner.
Obj. 3 - Identify the requirements for a robust standards development process.	CH2, CH3 and CH4	This objective is fulfilled by reviewing the literature in the following related fields: standards development, ontology engineering, semantic interoperability and design science research method (since OntoStanD itself is aimed at designing some artefacts).
Obj. 4 - Develop a methodological approach for ontology-based standards development.	CH4 and CH5	OntoStanD V1.0 developed in Chapter 4, which covered defining ontological conceptual models of standards. It was extended and refined in Chapter 5 for covering the normative statements of standards.
Obj. 5 - Evaluate and demonstrate the practical adequacy of the proposed method.	CH4, CH5 and CH6	OntoStanD was evaluated for its utility and practical adequacy by its application on three scenarios: (1) OASIS TAG Widget Specification, (2) a subsection of the Greek e-GIF standards and (3) OASIS ebBP standards.

Table 7-1. How the objectives of the research are addressed in various chapters.

7.2 Research Contributions

This research follows the design science research guidelines (March, Smith 1995, Peffers et al. 2007, Vaishnavi, Kuechler 2007, Hevner et al. 2004), where, like any other research paradigm, contribution is significantly important. The main contributions of design research are one or more artefacts taking the form of method, model and/or instantiation (Vaishnavi, Kuechler 2007, Hevner et al. 2004). Utility theory (Walls et al. 1992) is also considered as an output of design science research in this research parallel with DSR artefacts as explained in Chapter 3. Proposing the use of ontology as a basis for standards and their development process is the core utility theory in this research, which the other artefacts are built around. This research has a set of artefacts, which are summarised as follows:

OntoStanD: A methodology for designing ontology-based standards, which takes into account the collaborative aspects of standards development as well as the requirements for effective semantic interoperability. *OntoStanD* pays special attention to the process of conceptual modelling of standards (existing or to be developed) and also the ability to define the normative statements of standards on top of the developed robust conceptual model. This is considered to be an artefact of type *method* in DSR terms. The utility and practical adequacy of *OntoStanD* are demonstrated by its application on three existing standards specifications. *OntoStanD* is the main contribution of this research and is presented in Figure 7-1.

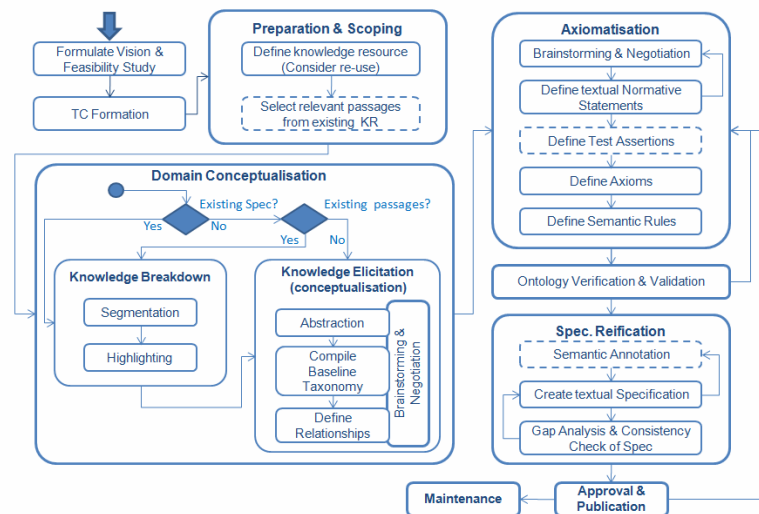


Figure 7-1. Overview of OntoStanD Methodology

OASIS TAG Widget ontological model: *OntoStanD* is primarily applied on a small, but rather comprehensive, specification extract. The outcome of this application is a robust ontological conceptual model for the OASIS TAG Widget specification extract and also its associated normative statements.

The Greek e-GIF (Selected part) ontological model: *OntoStanD* was further applied on a subsection of the Greek e-Government Interoperability Framework standards. The outcome of this application was an ontology for the selected part

of the standard, which is re-using an existing, general purpose, taxonomy for the Greek e-Government (Sourouni et al. 2008). This ontology was, however, not instantiated due to the lack of available digital data at the time of this research.

OASIS ebBP ontological model: Finally an ontology for the OASIS Business Process Specification Schema (ebBP) was developed using OntoStanD, resulting in a cohesive ontology for the standard, which is submitted to the ebBP TC for its further application. This, and the last two ontological models are considered as *models* in DSR terms, which were evaluated accordingly.

Table 7-2, provides a summary of the artefacts in this thesis.

Artefact/Output	DSR Facet	Presented in Chapter
Ontological approach – view that clear articulation of things, relationships and axioms improves quality.	Utility theory	Chapter 4
OntoStanD methodology	Method	Chapter 4 and Chapter 5
Ontological model of the OASIS TAG Widget specification	Model	Chapter 4 and Chapter 5
Ontological model of the selected part of the Greek e-GIF standards	Model	Chapter 6
Ontological model of the OASIS ebBP standard	Model	Chapter 6

Table 7-2. DSR artefacts produced in this thesis

7.3 Implications of the Work

The primary value and relevant benefits of this research for theory and practice are as follows:

Semantic interoperability is a significant challenge for effective inter-organisational interoperations. Many organisations spend a considerable amount of time and effort contributing to standards development activities to make sure that the developed standards meet their requirements. In such circumstances, a substantial amount of time is normally wasted due to misunderstanding of concepts and relationships between members of a TC and

also for changes to be made to their non-methodologically developed/non-semantic conceptual models and normative statements. This research provided a methodological approach for developing ontology-based standards, examples of which demonstrate the potential to achieve higher quality standards and consequently semantic interoperability, which provide a means to save time and money for larger organisations. This approach could also bring economic benefits for smaller companies (who use the standards) by providing more effective interoperability, less ambiguity between trading partners and smoother transactions.

Using OntoStanD and getting familiar with the semantic aspects, may facilitate tools development as a robust conceptual model is defined for each standard, which could directly be used during the process of implementation. Furthermore, the TC members would have a better and more detailed insight on the formal aspects of a standard, which would potentially help them in better communication with the tools developers. The current research provides practitioners in industry with valuable, systematic, and customisable means to design, implement, analyse, evaluate and test standards and their implementations and also formalise and change new and existing standards for achieving higher quality standards.

The above improvements would benefit: (1) Standards developers, analysts, modellers and implementers by allowing them to achieve their strategic goals and objectives through better utilisation of modelling and reuse; (2) organisations which conform to a specific standard by providing them with a robust, formal and semantic-based basis for the standards they use, which helps them with achieving their initial goal for conforming to standards – interoperability; (3) the standards community in large, by providing them with a methodological approach for developing standards in a more robust way, formalising and reusing them; and (4) academia, which will benefit from the

cross-disciplinary research in the four inter-related fields of design research, standards, conceptual modelling, semantic web and ontologies.

7.4 Limitations and Challenges

Although this research has made a number of valuable contributions, like any other research, it has a set of limitations and challenges associated with it. The following provides a set of challenges faced during the process of this research:

DSR Evaluation: One of the significant challenges in conducting this research was its evaluation. This research followed DSR guidelines which, given their breadth, do not provide straightforward and clear guidelines for evaluating design artefacts. Therefore, this research had to formulate a suitable method for evaluating its artefacts. To do so, various DSR publications were studied and scrutinised in combination with evaluation methods for standards, conceptual models as well as ontologies. On the basis of this analysis, the researcher sketched out a set of evaluation factors for evaluating the ontological conceptual models which are the result of application of OntoStanD.

In addition, this research took a more theoretical position for evaluating design methods. This view suggested that some approaches will relate better to the structure and practice of the world, rather than being true or false, or right and wrong. Consequently, a pragmatic view was taken based around the notion of practical adequacy of methods (Sayer 1992). This notion proposes that knowledge should be judged for its usefulness rather than being true or false and must generate expectations of the world that are: (a) actually realised; and (b) intersubjectively intelligible and acceptable. This research took the view that if the models resulting from application of a method are of high quality, it could be inferred that the method used for modelling is practically adequate. A more thorough evaluation of OntoStanD would however materialise when it is

adopted and tested by its main target audience: the standards developers and groups.

Evaluating the application of OntoStanD only for formalising existing standards: OntoStanD is a comprehensive methodology for developing and formalising ontology-based standards. To demonstrate utility and practical adequacy of OntoStanD, it was applied to three existing standards. OntoStanD is however, designed to be suitable for developing new standards as well as formalising existing standards. It was not possible to demonstrate and evidence new standards, however, as no standard could be developed within the scope of this research. OntoStanD is currently being disseminated and introduced to the research and practice communities through relevant publications. Furthermore, the author is a member of OASIS and is in the process of introducing OntoStanD to the standards body for the purpose of further evaluation. In addition, the sponsors of this research are actively participating in various standardisation activities and intend to make use of OntoStanD (in part to persuade standards bodies to use it for further validation of the method and improvements).

OWA and conformance checking: Ontology languages, and in particular OWL, provide great expressive power for conceptual modelling and formalising concepts and relationships in a domain, which is of considerable benefit for the standards communities. Conformance checking is an important matter for standards and it is desirable to use OWL, or other similar languages, for this purpose as well. However, the Open World Assumption (OWA) of OWL and its lack of Unique Name Assumption (UNA) (Sirin et al. 2008) make it difficult, if not impossible, to use OWL for consistency checking. They are, however, two characteristics which significantly add to OWL's expressive power. Under the OWA it is assumed that the information in the knowledge base is incomplete by default, which means a statement cannot be inferred to be

false on the basis of a failure to prove it (Sirin et al. 2008). Under the Closed World Assumption (CWA), however, it is assumed that the information in knowledge base is complete and failure to proving a statement means that statement is false.

As explained in Chapter 5, each of OWA and the CWA are useful for certain circumstances. This was a challenging task in this research since one of the main objectives of ontologies developed for standards is to be used for consistency checking, which needs to be done under the CWA. To be able to use OWL as a powerful knowledge representation language for standards as well as a constraint language for consistency check and data validation, it was necessary to combine open world reasoning and closed world constraint checking (Sirin et al. 2008). To address the above issue, this research proposed a set of guidelines for closing the world when needed in the context of standards. Although this approach needs a considerable amount of manual work, it is considered to be practical. This thesis suggests the utilisation of the Pellet ICV reasoner, in combination with the proposed approach, when it is released.

7.5 Future Research Directions

Based on the findings herein, the following areas are considered to be important for the future of ontology-based information standards development:

- At the moment OntoStanD does not provide any guidelines for semantic annotation and semi automatic specification generation – leaving it to standards developers to choose the annotation the method if they want to do so at all. However, integrating an ontology annotation tool/method with OntoStanD is an interesting aspect for future research. As a point of note, work is being undertaken by the sponsors of this research,

which remains confidential at this point. It is intended that their work will be integrated with OntoStanD in the near future.

- A reference ontology for standards is beneficial to be developed as a complementary addition to OntoStanD. This would ground the developed ontological models in a foundational ontology which would further improve the interoperability of standards.
- Development of a new, specialised, Ontology based representation language for standards is an interesting approach for representing the conceptual models and normative standards of standards. Ideally this language could be used as a schema type language and may even replace XML or provide a common presentation language for all XML-based standards.
- A tighter integration of standards development activities with Web 2.0 and social web technologies is a promising direction for future of standards development, which needs to be investigated. As an example, OntoStanD could be integrated in social network type standards development platforms, such as KAVI Workspace (KAVI 2011).
- The economic benefits of using ontologies as a basis for standards development should be further analysed and studied. Typical large IT organisations will employ around 1% of their global workforce in standards related activities (source confidential 2011). Adopting an ontology-based standards development, such as OntoStanD, would potentially to some extent reduce the time spent and cost for organisations. However, paying for the IPs (patents) owned by other companies, which are incorporated into a specific under development standards and also other politics and commercial interests have always formed the bulk of the actual cost of a standard. Therefore, one may

argue that although OntoStanD (and similar approaches) would improve the quality of standards, facilitate achieving interoperability and reduce the time for reaching consensus by providing a firmer basis for arguing the technical issues, they do not necessarily reduce the cost of standards development by a considerable amount.

- To the best of our knowledge, OntoStanD is the first and only ontology-based standards development methodology (at this time) and therefore cannot be compared to any similar methods for further comparative analysis and evaluation. A comparative analysis of the application of OntoStanD with existing non-ontology-based standards development methods would however be of interest, which would only become relevant when the proposition of OntoStanD is supported by standards development organisations. Thus, there is the need for the standards technical committees to consider how the use of ontologies, and in particular OntoStanD, as a basis for their specifications affects the development of the standards, conformance checking, the implementers and end users of their standard.

REFERENCES

- Al-Debei, M.M. & Fitzgerald, G. (2009) "OntoEng: A Design Method for Ontology Engineering in Information Systems", *ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications*.
- Aldin, L., de Cesare, S. & Lycett, M. (2009) "A Semantic-Based Framework for Discovering Business Process Pattern", pp. 26.
- Anicic, N., Ivezic, N. & Jones, A. (2006) "An Architecture for Semantic Enterprise Application Integration Standards", *Interoperability of Enterprise Software and Applications*, pp. 25-34.
- Antoniou, G. & Harmelen, F.V. (2008) *A Semantic Web Primer*, 2nd edn, The MIT Press.
- Benbasat, I. & Zmud, R.W. (1999) "Empirical research in information systems: The practice of relevance", *MIS Quarterly: Management Information Systems*, vol. 23, no. 1, pp. 3-16.
- Bera, P. & Wand, Y. (2010) "Using ontology languages for conceptual modeling", *Journal of Database Management*, vol. 21, no. 1, pp. 1-28.
- Berndt, D.J., Hevner, A.R. & Studnicki, J. (2003) "The Catch data warehouse: Support for community health care decision-making", *Decision Support Systems*, vol. 35, no. 3, pp. 367-384.
- Berndt, D.J., Hevner, A.R. & Studnicki, J. (2001) "Data warehouse dissemination strategies for community health assessments", *Upgrade*, vol. 2, no. 1, pp. 48-54.
- Berners-Lee, T. (2010) *Long Live the Web: A Call for Continued Open Standards and Neutrality*, Scientific American.
- Berners-Lee, T., Hendler, J. & Lassila, O. (2001) "The Semantic Web", *Scientific American*, vol. 284(5), pp. 34-43.
- Bikakis, N., Giannopoulos, G., Dalamagas, T. & Sellis, T. (2010) "Integrating keywords and semantics on document annotation and search", In Proceedings of the International Conference on On the Move to Meaningful Internet Systems:

- Part II (OTM'10), Meersman, R., Dillon, T. and Herrero, P. (Eds.), Springer-Verlag, Berlin, Heidelberg, pp. 921-938.
- Bird, G. (1998) "The business benefit of standards", *Standard View*, vol. 6, no. 2, pp. 76-80.
- Breslin, J., Passant, A. & Decker, S. (2009) *The Social Semantic Web*, Springer Verlag, Berlin Heidelberg.
- BSI (2011) *The British Standards Institution*. Available:
<http://www.bsigroup.com/en/Standards-and-Publications/About-standards/How-are-standards-made/What-is-the-development-process/> [2011, 02/25].
- Buitelaar, P., Cimiano, P. & Magnini, B. (2004) *Ontology learning from text: methods, evaluation and applications*, ISO Press.
- CEN (2009) *European Committee for Standardisation*. Available:
<http://www.cen.eu/cen/products/en/pages/default.aspx> [2011, 12/11].
- Charalabidis, Y., Lampathaki, F., Sarantis, D., Sourouni, A., Mouzakitis, S., Gionis, G., Koussouris, S., Ntanos, C., Tsiakaliaris, C., Tountopoulos, V. & Askounis, D. (2008) "The Greek electronic Government Interoperability Framework: Standards and infrastructures for one-stop service provision", *Proceedings - 12th Pan-Hellenic Conference on Informatics, PCI 2008*, pp. 66.
- Chatterjee, S., Tulu, B., Abhichandani, T. & Li, H. (2005) "SIP-based enterprise converged networks for voice/video-over-IP: Implementation and evaluation of components", *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 10, pp. 1921-1933.
- Churchman, D. (1995) *Negotiation: Process, Tactics, Theory*, University Press of America.
- Cimiano, P. (2006) "Ontology Learning and Population from Text: Algorithms, Evaluation and Applications", Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- Conrad, k., Newman, B. & Smith, B. (2004) *A Case for the Ontological Expression of E-Business Standards* [Homepage of Ontolog Forum]. Available:
http://ontolog.cim3.net/file/resource/presentation/EIDX_Semantic-

Harmonization_20041201/Ontological_Expression_of_E-Business_Standards--KurtConrad-BoNewman-BobSmith_EIDX-20041201.pdf [2011, 12/11].

- Cruz, C. & Nicolle, C. (2006) "Ontology-based integration of XML data: Schematic marks as a bridge between syntax and semantic level", *WEBIST 2006 - 2nd International Conference on Web Information Systems and Technologies, Proceedings*, pp. 30.
- Cruz, I.F., Xiao, H. & Hsu, F. (2004) "An ontology-based framework for XML semantic integration", *Proceedings of the International Database Engineering and Applications Symposium, IDEAS*, pp. 217.
- Cui, Z., Jones, D. & O'Brien, P. (2002) "Semantic B2B integration: Issues in ontology-based approaches", *SIGMOD Record*, vol. 31, no. 1, pp. 43-48.
- Daga, A., De Cesare, S., Lycett, M. & Partridge, C. (2005) "An ontological approach for recovering legacy business content", *Proceedings of the Annual Hawaii International Conference on System Sciences*, pp. 224.
- De Leenheer, P. (2008) "Towards an ontological foundation for evolving agent communities", *Proceedings - International Computer Software and Applications Conference*, pp. 523.
- De Moor, A., De Leenheer, P. & Meersman, R. (2006) *DOGMA-MESS: A meaning evolution support system for interorganizational ontology engineering*.
- De Vries, H.J. (2006) "IT standards typology", *Advanced Topics in Information Technology Standards and Standardization Research*, vol. 1, pp. 1-26.
- Dong, L. & Linpeng, H. (2008) "A framework for ontology-based data integration", *Proceedings ICICSE 2008 - 2008 International Conference on Internet Computing in Science and Engineering*, pp. 207.
- Drummond Group Inc. (2011) *Cloud Computing Technologies Embrace AS4 Standard*. Available: <http://ebxml.xml.org/news/cloud-computing-technologies-embrace-as4-standard> [2011, 12/11].
- Durand, J., Green, S.D., Kulvatunyou, S. & Rutt, T. (2009) "Test Assertions on steroids for XML artifacts. Presented at Balisage: The Markup Conference 2009, Montréal, Canada, August 11 - 14, ", *In Proceedings of Balisage: The Markup Conference 2009. Balisage Series on Markup Technologies*, vol. 3.

- ETSI (2010) *European Telecommunications Standards Institute*. Available:
<http://www.etsi.org> [2010, 02/25].
- European Commission (2009) *Modernising ICT Standardisation in the EU - The Way Forward*.
- Faure, C. (2004) "Beyond brainstorming: Effects of different group procedures on selection of ideas and satisfaction with the process", *Journal of Creative Behavior*, vol. 38, no. 1, pp. 13-34.
- Fensel, D. (2001) "Ontologies: Dynamic Networks of Formally Represented Meaning", *Proc.of the 1st Semantic Web Working Symposium, California, USA*.
- Fernandez, M., Gómez-Peréz, A. & Juristo, N. (1997) "Methontology: From Ontological Art towards Ontological Engineering", *Proceedings of the aaai'97 spring symposium series on ontological engineering*.
- Fernández-López, M. & Gómez-Pérez, A. (2002) "Overview and analysis of methodologies for building ontologies", *Knowledge Engineering Review*, vol. 17, no. 2, pp. 129-156.
- Folmer, E. & Putner, M. (2011) *Management and Development Model for Open Standards (BOMOS) version 2*.
- Folmer, E., Oude Luttighuis, P. & van Hillegersberg, J. (2011) "Do semantic standards lack quality? A survey among 34 semantic standards", *Electronic Markets*, pp. 1-13.
- García, R. & Gil, R. (2007) *Facilitating business interoperability from the semantic web*.
- Gavish, B. & Gerdes Jr., J.H. (1998) "Anonymous mechanisms in group decision support systems communication", *Decision Support Systems*, vol. 23, no. 4, pp. 297-328.
- Gemmill, J., Srinivasan, A., Lynn, J., Chatterjee, S., Tulu, B. & Abhichandani, T. (2004) "Middleware for scalable real-time multimedia cyberinfrastructure", *Journal of Internet Technology*, vol. 5, no. 4, pp. 405-420.
- Giannopoulos, G., Bikakis, N., Dalamagas, T. & Sellis, T. (2010) *GoNTogle: A tool for semantic annotation and search*, In Proceedings of ESWC 2010. pp.376-380.
- Glass, R. (1999) "On design", *IEEE Software*, vol. 16, no. 2, pp. 103-104.

- Gomez-Perez, A., Fernandez-Lopez, M. & Corcho, O. (2004) *Ontological Engineering*, Springer Verlag.
- Gomez-Perez, A. (2001) "Evaluation of ontologies", *International Journal of Intelligent Systems*, vol. 16, no. 3, pp. 36.
- Gong, R., Li, Q., Ning, K., Chen, Y. & O'Sullivan, D. (2006) *Business process collaboration using semantic interoperability: Review and framework*.
- Greek e-GIF (2010) *Greek e-Government Interoperability Framework*. Available: <http://www.e-gif.gov.gr/portal/page/portal/egif/> [2011, 10/11].
- Greek Informatics Development Agency (2010). Available: <http://www.yap.gov.gr/> [2011, 10/11].
- Greek Information Society (2009) *Certification Framework for Public Administration Sites and Portals (in Greek)*. Available: <http://www.e-gif.gov.gr/portal/pls/portal/docs/211033.PDF> [2011, 10/11].
- Grenon, P. & De Francisco, D. (2009) "Ontology-strength industry standards: The case of the telecommunication domain", *Formal Ontologies Meet Industry - Proceedings of the Fourth International Workshop (FOMI 2010)*, Ferrario, R. and Oltramari A. (Eds.), *Frontiers in Artificial Intelligence and Applications*, IOS Press.
- Grenon, P., Wimalaratne, S. & De Bono, B. (2011) "Ontology-strength industry standards: The case of the clinical trial domain", *Formal Ontologies Meet Industry - Proceedings of the Fifth International Workshop (FOMI 2011)*, Vermaas, P. E. and Dignum, V (Eds.).
- Gruber, T.R. (1995) "Toward principles for the design of ontologies used for knowledge sharing?", *International Journal of Human - Computer Studies*, vol. 43, no. 5-6, pp. 907-928.
- Gruninger, M. & Fox, M.S. (1995) "Methodology for the design and evaluation of ontologies", *Industrial Engineering*, vol 95, eds. Bernus & L. Nemes, Cornwall, Great Britain, pp. 1-10.
- Guarino, N. (1998) "Formal Ontology in Information Systems", *Proceedings of the 1st International Conference June 6-8, 1998*, IOS Press, Amsterdam.

- Haller, A., Gaaloul, W. & Marmolowski, M. (2008) "Towards an XPD L compliant process ontology", *Services - Part I, IEEE Congress on Services*, pp. 83-86.
- Halpin, T.A. (2001) *Information Modelling and Relational Databases: From Conceptual Analysis to Logical Design*, Morgan Kaufman Publishers, San Francisco, California.
- Heravi, B.R., Bell, D., Lycett, M. & Green, S.D. (2010a) "Towards an Ontology for Automating Collaborative Business Processes", pp. 311.
- Heravi, B.R., Bell, D., Lycett, M., Green, S.D. & Snelling, D. (2010b) "Towards Ontology based e-Business Standards", *ICE-B 2010 - Proceedings of the International Conference on e-Business*, pp. 169.
- Hevner, A.R., March, S.T., Park, J. & Ram, S. (2004) "Design science in information systems research", *MIS Quarterly: Management Information Systems*, vol. 28, no. 1, pp. 75-105.
- Höfferer, P. (2007) "Achieving Business Process Model Interoperability Using Metamodels and Ontologies", *Proc. of the 15th European Conference on Information Systems (ECIS2007)*, pp. 1620-1631.
- Horrige, M. (2011) *A practical guide to building owl ontologies using protégé 4 and co-ode tools edition 1.3*, Available:
http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWL_TutorialP4_v1_3.pdf [2011, 12/11].
- Huma, Z., Rehman, M.J.-. & Iftikhar, N. (2005) "An ontology-based framework for semi-automatic schema integration", *Journal of Computer Science and Technology*, vol. 20, no. 6, pp. 788-796.
- IEEE (2011) *The Standards Development Lifecycle*. Available:
<http://standards.ieee.org/develop/> [2011, 12/11].
- IETF (1997) *Internet Engineering Task Force Key words for use in RFCs to Indicate Requirement Levels*. Available: <http://www.ietf.org/rfc/rfc2119.txt> [2011, 10/11].
- ISO (2011) *How are ISO standards developed?*, Available:
http://www.iso.org/iso/standards_development/processes_and_procedures/how_are_standards_developed.htm [2011, 12/11].

- ISO/IEC. (2001) *ISO/IEC 9126-1 Software Engineering-Product Quality-Part 1: Quality Model*, ISO.
- Jarrar, M. (2005) *Towards Methodological Principles for Ontology Engineering*, PhD thesis, Vrije Universiteit Brussel.
- Jarrar, M. & Meersman, R. (2008) *Ontology engineering - The DOGMA approach*, In *Advances in Web Semantics I*, Tharam S. Dillon, T. S., Chang, E., Meersman, R. and Sycara, K. (Eds.), Lecture Notes In Computer Science, Vol. 4891. Springer-Verlag, Berlin, Heidelberg.
- Jarrar, M. & Meersman, R. (2002) "Formal ontology engineering in the DOGMA approach", *LNCS*, vol. 2519, pp. 1238-1254.
- Jones, D., Bench-Capon, T. & Visser, P. (1998) "Methodologies for Ontology Development", *IT&KNOWS Conference of the 15th IFIP World Computer Congress*.
- Juran, J.M. & Gryna, F.M. (eds) (1988) *Juran's quality control handbook*, 4th edn.
- Kajan, E. & Stoimenov, L. (2005) "Toward an ontology-driven architectural framework for B2B", *Communications of the ACM*, vol. 48, no. 12, pp. 60-66.
- KAVI (2011) . Available: <http://kavi.com/> [2011, 12/11].
- Krechmer, K. (2005) "The meaning of Open Standards", HICSS '05, *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, Jan. 2005.
- Legner, C. & Wende, K. (2007) "The Challenges of inter-organizational Business Process Design - A research Agenda", *15th European Conference on Information Systems*, pp. 106-118.
- Liegl, P., Huemer, C. & Zapletal, M. (2009) "Towards a Global Business Document Reference Ontology", *Proceedings of the 2009 IEEE International Conference on Semantic Computing*, pp. 355-360.
- López, M.F., Gómez-Pérez, A., Sierra, J.P. & Sierra, A.P. (1999) "Building a chemical ontology using methontology and the ontology design environment", *IEEE Intelligent Systems and Their Applications*, vol. 14, no. 1, pp. 37-46.
- Maedche, A. & Staab, S. (2000) "Mining ontologies from text", In *Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and*

Management (EKAW '00), Dieng, R. and Corby, O. (Eds.), Springer-Verlag, London, UK, pp. 189-202.

- March, S.T. & Smith, G.F. (1995) "Design and Natural Science Research on Information Technology", *Decision Support Systems*, vol. 15, no. 4, pp. 251-266.
- Markus, M.L., Majchrzak, A. & Gasser, L. (2002) "A design theory for systems that support emergent knowledge processes", *MIS Quarterly: Management Information Systems*, vol. 26, no. 3, pp. 179-212.
- Motik, B. & Patel-Schneider, P. F., Parsia, B. (2009) 25 March 2011-last update, *OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax*. Available: <http://www.w3.org/TR/owl2-syntax> [2011, 12/11].
- Mylopoulos, J. (1992) "Conceptual modelling and telos", *Conceptual Modelling, Databases and Case: An Integrated View of Information Systems Development*.
- Nitzsche, J., Wutke, D. & Van Lessen, T. (2007) "An ontology for executable business processes", *Workshop SBPM*, vol. 251, pp. 52-63.
- Nunamaker, J.F., Chen, M. & Purdin, T.D.M. (1991) "Systems development in information systems research", *Journal of Management Information Systems*, vol. 7, no. 3, pp. 89-106.
- OASIS (2006) *ebXML Business Process Specification Schema Technical Specification (version 2.0.4)*. Available: <http://docs.oasis-open.org/ebxmlbp/2.0.4/OS/spec/ebxmlbp-v2.0.4-Spec-os-en.pdf> [2011, 12/11].
- OASIS (2005) *OASIS ebXML RegRep Version 3.0*. Available: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=regrep [2011, 12/11].
- OASIS (2007) *OASIS Guidelines to Writing Conformance Clauses* [Homepage of OASIS]. Available: <http://docs.oasis-open.org/templates/TCHandbook/ConformanceGuidelines.html> [2007, 04/09].
- OASIS (2007) *Web services business process execution language version 2.0 (WS-BPEL 2.0)*. Available: <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html> [2011, 12/11].
- OASIS (2010) *OASIS Technical Committee Process* [Homepage of OASIS]. Available: <http://www.oasis-open.org/committees/process.php> [2011, 02/ 30].

- OASIS ebXML Registry Technical Committee (2006) 25 September 2006-last update, *ebXML Registry Profile for Web Ontology Language (OWL) Version 1.5*. Available: <http://xml.coverpages.org/OASIS-regrepOWL-Profile-22611.pdf> [2011, 12/11].
- OASIS QUOMOS TC, Available: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=quomos [2011, 12/11].
- OASIS SET TC (2009), *Semantic Support for Electronic Business Document Interoperability (SET) TC*. Available: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=set [2011, 12/11].
- OASIS TAB (2011) 17 January 2011-last update, *Conformance, Interoperability and Portability Testing: Proposed Procedures and Practices* [Homepage of OASIS Technical Advisory Board]. Available: <http://wiki.oasis-open.org/tab/TestingPolicy>.
- OASIS TAG TC (2010) 13 February-last update, *Test Assertions Guidelines, Committee Draft 04*. Available: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tag [2010, 10 July].
- OASIS UBL TC (2008), *Universal Business Language v2.0*. Available: <http://docs.oasis-open.org/ubl/cs-UBL-2.0/UBL-2.0.html>.
- Obrst, L. (2003) "Ontologies for semantically interoperable systems", *International Conference on Information and Knowledge Management, Proceedings*, pp. 366.
- Obrst, L., Liu, H., Wray, R. & Wilson, L. (2002) "Ontologies for semantically interoperable electronic commerce", *Proceedings of ICEIMT'02, International Conference on Enterprise Modelling and Enterprise Integration Technologies (ICEIMT), and the Conference of the EI3-IC Initiative (Enterprise Inter- and Intra-organisational Integration - International Consensus)*, .
- Obrst, L., Wray, R.E. & Liu, H. (2001) "Ontological engineering for B2B e-commerce", *Formal Ontology in Information Systems: Collected Papers from the Second International Conference*, pp. 117.
- O'Reilly, T. (2005) *What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software*. Available: <http://oreilly.com/web2/archive/what-is-web-20.html> [2011, 10/12].

- Osborn, A.F. (1993) *Applied Imagination: Principles and Procedures of Creative Problem-Solving*, 3rd Rev edition, Creative Education Foundation.
- Peppers, K., Gengler, C.E. & Tuunanen, T. (2003) "Extending critical success factors methodology to facilitate broadly participative information systems planning", *Journal of Management Information Systems*, vol. 20, no. 1, pp. 51-85.
- Peppers, K. & Tuunanen, T. (2005) "Planning for IS applications: A practical, information theoretical method and case study in mobile financial services", *Information and Management*, vol. 42, no. 3, pp. 483-501.
- Peppers, K., Tuunanen, T., Rothenberger, M.A. & Chatterjee, S. (2007) "A design science research methodology for information systems research", *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45-77.
- Popov, B., Kiryakov, A., Ognyanoff, D., Manov, D. & Kirilov, A. (2004) "KIM - A semantic platform for information extraction and retrieval", *Natural Language Engineering*, vol. 10, no. 3-4, pp. 375-392.
- Pries-Heje, J., Baskerville, R. & Venable, J. (2008) "Strategies for Design Science Research Evaluation", *Proceedings of the 16th European Conference on Information Systems*, pp. 255-266.
- Purao, S. (2002) "Design Research in the Technology of Information Systems: Truth or Dare", *Design Research in the Technology of Information Systems: Truth or Dare*.
- Rebstock, M., Fengel, J. & Paulheim, H. (eds) (2008) *Ontologies-based business integration*, Springer Berlin Heidelberg.
- Rothenberger, M.A. & Hershauer, J.C. (1999) "A software reuse measure: Monitoring an enterprise-level model driven development process", *Information and Management*, vol. 35, no. 5, pp. 283-293.
- Sánchez, D.M., Cavero, J. & Martínez, E. (2007) "The Road Toward Ontologies" in *Ontologies*, R. Sharman, R. Kishore & R. Ramesh (Eds.), Springer, US, pp. 3-20.
- Sayer, A. (1992) *Method in Social Science: a Realist Approach*, 2nd edition edn, Routledge, London.
- Seng, J.-. & Lin, W. (2007) "An ontology-assisted analysis in aligning business process with e-commerce standards", *Industrial Management and Data Systems*, vol. 107, no. 3, pp. 415-437.

- Shah, R. & Kesan, J.P. (2008) "An empirical examination of open standards development", *Proceeding of the 41st Hawaii International Conference on System Sciences*, IEEE Computer Society Conference Publishing Services, pp. 212 (1-10) .
- Shanks, G., Tansley, E. & Weber, R. (2003) "Using ontology to validate conceptual models", *Communications of the ACM*, vol. 46, no. 10, pp. 85-89.
- Sherif, M.H., Egyedi, T.M. & Jakobs, K. (2005) "Standards of quality and quality of standards for telecommunications and information technologies", *Proceedings of the 4th International Conference on Standardization and Innovation in Information Technology, 2005*, pp. 221.
- Simon, H.A. (1996) *The Sciences of the Artificial (3rd ed.)*.
- Simperl, E.P.B. & Tempich, C. (2006) "Ontology engineering: A reality check", *OTM (2006)*, vol 4275:6, Springer, pp. 836-854.
- Singh R., Iyer L.R., Salam A.F. (2005) "The semantic E-business vision", *Communications of the ACM*, vol. 48, no. 12, pp. 38-41.
- Singh, R., Iyer, L.S. & Salam, A.F. (2005) "Semantic eBusiness", *International Journal on Semantic Web & Information Systems*, vol. 1, no. 1, pp. 19-35.
- Sirin, E., Smith, M. & Wallace, E. (2008) "Opening, closing worlds-on integrity constraints", *Proceedings of the Fifth OWLED Workshop on OWL: Experiences and Directions*.
- Sourouni, A., Lampathaki, F., Mouzakitis, S., Charalabidis, Y. & Askounis, D. (2008) "Paving the way to eGovernment transformation: Interoperability registry infrastructure development", *Proceeding of the 7th International conference on Electronic Government*, Wimmer, M. A., Scholl, H. J., and Ferro, E. (Eds.), Springer-Verlag, Berlin, Heidelberg, pp. 340-351.
- Spyns, P. (2005) "Object role modelling for ontology engineering in the DOGMA framework", *On The Move to Meaningful Internet Systems (OTM) Workshop*, Lecture Notes in Computer Science, vol. 3762/2005, pp. 710-719
- Spyns, P., Meersman, R. & Jarrar, M. (2002) "Data modelling versus ontology engineering", *SIGMOD Record*, vol. 31, no. 4, pp. 12-17.

- Spyns, P., Tang, Y. & Meersman, R. (2008) "An Ontology Engineering Methodology for DOGMA", *Journal of Applied Ontology, special issue on "Ontological Foundations for Conceptual Modeling"* Giancarlo Guizzardi and Terry Halpin (Eds.), vol. 3, no. 1-2, pp. 13-39.
- Staab, S., Studer, R., Schnurr, H.-. & Sure, Y. (2001) "Knowledge processes and ontologies", *IEEE Intelligent Systems and Their Applications*, vol. 16, no. 1, pp. 26-34.
- Sure, Y. & Staab, S.S.,R. (2003) "On-To-Knowledge Methodology" in *Handbook on Ontologies*, pp. 117-132.
- Tang, Y. & Meersman, R. (2009) "SDRule markup language: Towards modeling and interchanging ontological commitments for semantic decision making", *Handbook of Research on Emerging Rule-based Languages and Technologies: Open Solutions and Approaches*.
- Tao, J., Sirin, E., Bao, J. & McGuinness, D.L. (2010) "Integrity Constraints in OWL", *Proceedings of the National Conference on Artificial Intelligence*, pp. 1443.
- Tempich, C., Pinto, H.S. & Staab, S. (2006) "Ontology engineering revisited: An iterative case study", *The Semantic Web Research and Applications*, Springer, pp. 110-124.
- The Ontolog Community (2010). Available: <http://ontolog.cim3.net/>.
- The Ontolog Forum *UBL Ontology*. Available: <http://ontolog.cim3.net/cgi-bin/wiki.pl?UblOntology>.
- UN/CEFACT (2003) *Core Components Technical Specification*. Available: http://www.unece.org/fileadmin/DAM/cefact/ebxml/CCTS_V2-01_Final.pdf [2011, 11/12].
- Uschold, M. & Gruninger, M. (1996) "Ontologies: Principles, methods and applications", *Knowledge Engineering Review*, vol. 11, no. 2, pp. 93-136.
- Uschold, M. & King, M. (1995) "Towards a Methodology for Building Ontologies", *In Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*.

- Vaishnavi, V. & Kuechler, W. (2007) June 29, 2007-last update,
Design Research in Information Systems [Homepage of isworld]. Available:
<http://www.isworld.org/Researchdesign/drISworld.htm> [2008, 05/27].
- Van der Aalst, W.M.P. & Kumar, A. (2003) "XML-based schema definition for support of interorganizational workflow", *Information Systems Research*, vol. 14, no. 1, pp. 23-46.
- Van der Eijk, P., Nickull, D., Dubray, J., Evans, C., Chappell, D., Harvey, B., Noordzij, M., Vegt, J., McGrath, T., Chopra, V. & Peat, B. (2001) *Professional EbXML Foundations*, WROX Press Ltd.
- Van der Veer, H. & Wiles, A. (2008) "Achieving Technical Interoperability - The ETSI Approach", *ETSI White Paper No.3*, pp. 1-29.
- Vargas-Vera, M., Motta, E. & Domingue, J. (2001) "Knowledge Extraction by using an Ontology-based Annotation Tool", *K-CAP 2001 workshop on Knowledge Markup and Semantic (K-CAP '01)*.
- Venable, J.R. (2006a) "A Framework for Design Science Research Activities", *Proceedings of the Information Resource Management Association Conference, Washington, DC, USA, May 21-24 (2006)*.
- Venable, J.R. (2006b) "The Role of Theory and Theorising in Design Science Research", *Proceedings of the First International Conference on Design Science Research in Information Systems and Technology*.
- Vujasinovic, M., Ivezic, N., Kulvatunyou, B., Barkmeyer, E., Missikoff, M., Taglino, F., Marjanovic, Z. & Miletic, I. (2010) "Semantic mediation for standard-based B2B interoperability", *IEEE Internet Computing*, vol. 14, no. 1, pp. 52-63.
- W3C (2011) *W3C Semantic Web Activity* [Homepage of W3C], Available:
<http://www.w3.org/2001/sw/> [2011, 12/11].
- W3C (2010) *XQuery 1.0: An XML Query Language (Second Ed.)*. Available:
<http://www.w3.org/TR/xquery/> [2011, 12/11].
- W3C (1999) *XML Path Language (XPath), Version 1.0*. Available:
<http://www.w3.org/TR/xpath/> [2011, 12/11].

- W3C COPRAS (2007) *Standardization guidelines for IST research projects interfacing with ICT standards organizations* [Homepage of W3C]. Available: <http://www.w3.org/2004/copras/docu/D15.html> [2010, 07/05].
- Walls, J.G., Widmeyer, G.R. & El Sawy, O.A. (1992) "Building an information system design theory for vigilant EIS", *Information Systems Research*, vol. 3, no. 1, pp. 36-59.
- Wand, Y. & Weber, R. (1993) "On the ontological expressiveness of information systems analysis and design grammars", *Journal of Information Systems*, vol. 3, no. 4, pp. 217-237.
- Weber, R. (2003) "Conceptual modelling and ontology: Possibilities and pitfalls", *Journal of Database Management*, vol. 14, no. 3, pp. 1-20.
- Wong, W.Y. (2009) *Learning Lightweight Ontologies from Text across Different Domains Using the Web as Background Knowledge*, PhD thesis, The University of Western Australia, School of Computer Science and Software Engineering.
- Wu, B., Li, L. & Yang, Y. (2006) "Ontological approach towards E-business process automation", *In Proceedings of the IEEE International Conference on e-Business Engineering(ICEBE '06)*, IEEE Computer Society, Washington, DC, USA, pp. 154-161.
- Xiao, H. & Cruz, I.F. (2006) "Integrating and exchanging XML data using ontologies", *Journal on Data Semantics VI: Special Issue on Emergent Semantics*, Spaccapietra, S., Aberer, K., Cudré-Mauroux, P. (Eds.), Lecture Notes in Computer Science, Springer, pp. 67-89.
- Zhao, G. (2004) "Application Semiotics Engineering Pprocess", *in the Proceedings of the 14th International Conference of Software Engineering and Knowledge Engineering (SEKE04)*, pp. 354.
- Zhao, G. & Meersman, R. (2005) *Architecting ontology for scalability and versatility*, in "On the Move to Meaningful Internet Systems", Meersman R. et al. (Eds.), Lecture Notes in Computer Science, vol. 2, Springer Berlin / Heidelberg, pp.1605-1164.

APPENDIX I: Letter of Support from Fujitsu



Registration Number: 4153469

FUJITSU LABORATORIES OF EUROPE LIMITED

Hayes Park Central, Hayes End Road, Hayes, Middlesex, UB4 8FE, UK.
Tel: (London) +44 (0)20 8606 4485. Fax: +44 (0)20 8606 4539.

December 13, 2011

To Whom It May Concern,

I am writing in general support of the work undertaken by Bahareh Rahmazadeh Heravi in conjunction with researching and writing her PhD.

The work looked at a number of technologies and ways in which they might be applied within the industry. Lighting on the use of ontologies in support of the standardisation process fits our interests well. Fujitsu is engaged in a very wide range of standardisation activities around the world and we understand the importance of clean models to both the quality of the eventual standard, but also to the improved communication they provide.

We support the goals of this work and will continue to encourage the uptake of this technology in the standardisation process. We also believe that moving on to the next step (integrated tool support) would facilitate this uptake.

We also wish her luck with the final submission and future endeavours.

Yours,

A handwritten signature in black ink, appearing to read "David Snelling".

Dr. David Snelling
Assistant Division Manager
Fujitsu Laboratories of Europe

APPENDIX II: Certification Framework for Public Administration Sites and Portals

Guidelines on the Structure and Content of a web site

Section 3.3.3 of the Certification Framework

The Greek e-Government Interoperability Framework

The Greek e-Government Interoperability Framework Version 4.0 (May 2009) defines standards, specifications and rules for the development and deployment of web-based front and back office systems for the Greek Public Administration, at National and Local level, which will accelerate the development of electronic collaboration among public agencies, for the delivery of high quality and secure one-stop e-Government services to businesses, citizens and other public bodies. This framework consists of the following components:

- 1) *The Certification Framework for Public Administration Sites and Portals*
- 2) The Interoperability and Electronic Services Provision Framework
- 3) The Digital Authentication Framework
- 4) The Documentation Model for Public Administration Processes and Data
- 5) The Interoperability Registry
- 6) The XML schema repository

Certification Framework for Public Administration Sites and Portals

The Certification Framework for Public Administration Sites and Portals specifies the guidelines and standards to be followed by the public agencies at central or local level, when designing, developing and deploying e-Government portals and supporting e-Government services. The purpose of the Certification Framework is to contribute to the prospective homogenisation (aesthetical and functional) of the Greek Public Administration web presence.

The guidelines emerging from the Certification Framework can be classified in the following distinct categories:

- **General Principles**, which describe the basic principles to be followed by public agencies when designing, developing or operating e-Government portals. According to the Certification Framework, all government portals should adhere to the following general principles: Principle of Equality and Isonomy, Principle of Completeness and Credibility, Principle of Trustfulness, Principle of proper Use of Public Resources.
- **Website Management**, which describes the organizational procedures for proper development and maintenance of the Public Sites, as well as the

proposed organizational schema of the website, roles and responsibilities for managing its content and overall functionality.

- **Website Structure and Organization**, in terms of standards and specifications for the structure, the categorization and the presentation of the content based on multi-faceted classification schemes, such as: a) the Greek Government Category List (with the basic categories and subcategories for classifying content and services) that accompanies the Framework, and b) the life events and business episodes list, which allows citizens and businesses locate information and services without needing to remember which public administration is responsible for them. This section includes guidelines regarding:
 - Domain Naming, setting out the national rules and principles for registering and managing a .gov.gr domain name, i.e. the website's name, language, writing style and name size.
 - Ergonomics and Visual Styles, including colours and colour combinations, text fonts, icons, links, writing styles, web pages' length and size, content presentation and special presentation formats usage, such as multimedia content or attached documents.
 - Structure and Content of the first and the rest of the pages in a website. It specifies an indicative content structure with the necessary components, such as organisation logo and contact details, search engine, site map, browsing capabilities.
 - Accessibility, which requires compliance with the Web Content Accessibility Guidelines version 1.0 Level A (obligatory), Level AA (recommended) and Level AAA (under consideration) as issued by the W3C Web Accessibility Initiative.
 - Management of Multiple websites, which sets the guidelines that a public organisation should follow when managing more than one websites or portals.

- **Electronic Services and Interoperability** with back-office systems encompassing standards and specifications that fall within the scope of the Interoperability and Services Framework:
 - Electronic Services: defining the necessary information to appear in a website for each service as well as the requirements for electronic transactions progress monitoring and —offlinel capabilities (i.e. printing, downloading and local storage).
 - Added Value Services, such as newsletters, forum and RSS feeds available to users with appropriate mechanisms.
 - Interoperability between the website, the supporting back-office systems and other websites for content retrieval, aggregation, and syndication.

- **Assessment of a Website's Content, Electronic Services and Functionality**, aiming at collecting and exploiting data from the site's everyday operation in order to continuously improve the provided services and the overall user's experience.

- **Security & Privacy** that discusses issues around Security policies, Access and Authentication to electronic services and classified content, Secure data storage, Transaction integrity, System availability and performance, Physical

security. Service Level Agreements are also introduced here in case a government portal or website is hosted in a private sector organization's premises.

- **Legal Issues**, aiming at ensuring users' privacy, and notifying them about limitations that may exist regarding the use of the website's content and services, in terms of copyright, terms of use and disclaimers.
- **Website Dissemination**, indicating an overall dissemination strategy that may extend from the registration of the website to search engines and web directories and sharing links with other relevant websites to further dissemination actions, such as the organization of workshops, and the preparation of printed advertising material, television and radio spots.

Greek e-Government Interoperability Framework → Certification Framework for Public Administration Sites and Portals → Website Structure and Organization → Structure and Content

Guidelines on the Structure and Content of public web sites

Rules are marked as <i>mandatory (KY)</i> or <i>optional (KII)</i> .
--

1. Homepage

[KY.29] The homepage of a public institution's web site OWES TO include:

- the public institution's logo and official name
- a short welcoming message
- the postal address, telephone and fax numbers and the e-mail address of the public institution.

Alternatively, the institution's contact details may appear in a separate web page, entitled "Contact", that OWES TO be accessible from a visible area of the web site's homepage.

[KY.30] The homepage of a public institution's web site OWES TO include links to the rest of the web site's content:

- Informational Content
- Services
- Tools and mechanisms for search, navigation and communication

as well as links to the web sites of similar institutions or institutions providing similar services.

The former categories for content are indicative and IT IS SUGGESTED that they are analyzed at a greater level of detail.

Alternatively or additionally to the above guidelines, IT IS SUGGESTED that a public web site's homepage includes links to the rest of the web site's content according to the target audience of each type of content or services, e.g.: Citizens, Enterprises, Institutions/Organizations.

[KY.31] The web site's visitor MUST be able to access the homepage from all other pages of the public web site.

2. Web pages' Content

[KY.32] Each public web site OWES TO include:

- the public institution's logo and official name
- the public institution's services flow chart
- a short presentation of the public institution
- information on the institution's administration
- the legislation with which the institution's operation and competences must comply
- information on the services offered by the public institution and the way they are provided
- news and announcements

[KII.19] It is SUGGESTED that the web sites of Public Administration institutions include:

- the terms and conditions of their use
- the most frequently asked questions that are submitted with regard to the services provided by the institution and the corresponding answers (FAQs)
- links redirecting the user to public web sites of relevant content
- information with regard to the number of times the web site is viewed as well as with regard to the use of the electronic services it provides

3. Content Structure

[KY.33] The structure of the web site's content MUST follow a tree catalogue structure instead of a flat file structure.

[KII.20] It is SUGGESTED that the tree structures that enable content management within a public web site are extended in width.

[KII.21] It is SUGGESTED that accessing the content and services of a public web site does not require more than three "clicks".

[KII.22] It is SUGGESTED that Public Administration institutions apply "bread crumbing techniques", so that users can access intermediary levels of the web

site structure in just one click, without having to access them indirectly through the web site's content categories and subcategories.

4. Content Management

4.1 Content Creation and Updating

[KY.34] The content of a public web site OWES TO:

- serve the policy of the institution that owns it
- have an explicit structure that facilitates user navigation
- be reliable in terms of its style and level of accuracy
- effectively communicate the institution's goals and to satisfy the users' needs

[KY.35] The institution that owns a web site OWES TO have a well determined procedure for content and services' provision through the public web site. This procedure OWES TO be followed during the initial publication of new content as well as whenever the web site's content and services are updated. The person assigned the role to be responsible for the web site's Content and Services, is responsible for the smooth application of this procedure as well.

[KY.36] The content authors ("content owners") on behalf of the public institution MUST ensure that:

- the content is correct, up-to-date and serves its purpose
- the content's style complies with the style defined for the specific web site
- the contact points, included within (e-mail address, telephone numbers), are valid
- the content is filed at the specific points, defined by the corresponding procedures
- the content complies with all publishing standards, defined by the person that is in charge of the web site's Content and Services.

4.2 Content Storage

[KII.23] IT IS SUGGESTED that the content appearing in the web site of a public institution is stored in relational databases. Older versions of this content (no longer in use but maintained as "history") may be stored in other means/systems.

IT IS SUGGESTED that the maintenance of the history file of a public web site's content is examined in combination with the content backup policy applied by the institution.

[KY.37] The updating or addition of new content in the public web site and each action that modifies the content and services provided by the latter in general **MUST** be recorded in the public web site's log files.

[KY.38] The institution that owns a public web site **OWES TO** maintain a file with the content that has been available at the web site at each specific moment in time (for a reasonable time period).

4.3 Content Organization

4.3.1 Government Category List

[KY.39] Public Administration institutions **OWE TO** organize the information and the services they provide through the public web site in representative content categories. These categories **MUST** be included in the Government Category List (GCL) that is in effect.

4.3.2 Metadata

[KY.40] The institution that owns a public web site **OWES TO** register the appropriate metadata for the informative material and the services that provides through its web site. These metadata **HAVE TO** comply with the policies, the standards and the specifications of the Interoperability and Electronic Services Provision Framework.

[KY.41] The institutions that create original content to be published in their web sites **OWE TO** document the latter through the use of appropriate metadata, so that the content can be exploited by other public web sites as well, and content aggregation and content syndication options can be supported.

[KY.42] The metadata kept with regard to the public web site's content and services **HAVE TO** be able to be exported in XML format.

[KY.43] Public Administration institutions **OWE TO** register metadata with regard to their web sites and as well as with regard to the content published in the latter according to the requirements of Table 4-2 [Section 4 (Metadata Model) of the Documentation Model for Public Administration Processes and Data].

5. Content correctness – completeness and up-to-datedness

[KY.44] The content that appears in a public web site **HAS TO** be checked for spelling and syntax errors, inaccuracies and ambiguities before its publication.

[KY.45] The content that appears in a public web site **MUST** reflect the current status of the issue to which it is related.

[KY.46] The institution **OWES TO** ensure that the updating of the public web site's content involves a specific updating process as well as suitable roles.

[KII.24] IT IS SUGGESTED that for each topic or web page of a public web site the date of the last content update is published.

6. Language of Content

[KY.47] All content of the public web site **MUST** be available in the Greek language. If part of the content is required to be available in an additional language, this **MUST** be the English language.

[KII.25] In case a third or fourth (etc.) language has to be selected for the content of a public web site, IT IS SUGGESTED that the relevant decision is made according to the target audience to which the public institutions' services are addressed, e.g. economic immigrants, tourists, foreign institutions etc.

7. Search engine

[KY.48] Public Administration institutions **OWE TO** incorporate in their web sites a search engine.

[KY.49] The search engine **MUST** at least offer the visitor the capability to perform a simple search in the public web site's content.

[KY.50] The search engine **MUST** be accessible from all web pages of the public web site.

[KII.26] It is **SUGGESTED** that the search engine offers the visitor the following capabilities:

- Keyword search in the web site content (full-text) with any words or phrases.
- Composite search, performed through combination of elements, such as date, category, topic, words or phrases.

8. Web site map

[KY.51] The web sites of Public Administration institutions **HAVE TO** include a navigation map on their contents.

[KY.52] The navigation map of a public web site **HAS TO** be accessible from all pages of the public web site.

9. Navigation

[KII.27] IT IS SUGGESTED that the thematic categories in which the public web site's content and services are organized, are accessible from all pages of the public web site.

[KY.53] None of the public web site's pages is allowed to include "under construction" messages. In case a web page is under construction, the fact HAS TO be reported from the very first moment and the corresponding link HAS TO be inactive.

[KY.54] The links included in a public web site, pointing to other areas of the web site MUST be regularly checked, so that it is ensured that they lead to the appropriate content.

10. Communication with the institution

[KY.55] The web site of a public administration institution HAS TO include explicit information on the procedures foreseen for communication among the visitors and the institution's services.

[KII.28] It is SUGGESTED that the web site of a public administration institution includes information on the names, telephone and fax numbers of the contact points, responsible for specific issues or services of the institution.

[KY.56] The web site of a public administration institution HAS TO include e-mail addresses and contact forms, through which web site visitors are able to contact the institution in order to submit queries with regard to the issues covered by the latter.

[KII.29] It is SUGGESTED that the web site of a public administration institution includes information on the ways and procedures available for reporting complaints with regard to the services provided by the institution.

APPENDIX III: The OASIS TAG Widget Ontology and Rules

Namespace: dc <<http://purl.org/dc/elements/1.1/>>
Namespace: <<http://www.semantic-b2bi.eu/Widget.owl#>>
Namespace: rdfs <<http://www.w3.org/2000/01/rdf-schema#>>
Namespace: swrl <<http://www.w3.org/2003/11/swrl#>>
Namespace: owl2xml <<http://www.w3.org/2006/12/owl2-xml#>>
Namespace: Widget <<http://www.semantic-b2bi.eu/Widget.owl#>>
Namespace: owl <<http://www.w3.org/2002/07/owl#>>
Namespace: xsd <<http://www.w3.org/2001/XMLSchema#>>
Namespace: swrlb <<http://www.w3.org/2003/11/swrlb#>>
Namespace: rdf <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>>
Ontology: <<http://www.semantic-b2bi.eu/Widget.owl>>

Annotations:

rdfs:comment "The Widget Ontology based on the Test Assertion Guidelines, Committee Draft 04, 13 February 2010."

Developed by Bahareh Rahmazadeh Heravi

Bahareh.Heravi@brunel.ac.uk

dc:creator "Bahareh Rahmazadeh Heravi

Fluidity research Group

Department of Information Systems and Computing

Brunel University

Bahareh.Heravi@brunel.ac.uk",

dc:rights "Copyright © 2010 Fujitsu Laboratories of Europe Limited."

ObjectProperty: hasCoating
Range: Coating

ObjectProperty: hasShape
Characteristics: Functional
Domain: Widget
Range: Shape
InverseOf: isShapeOf

ObjectProperty: isBatteryOf
InverseOf: hasBattery

ObjectProperty: hasLabel
Domain: Widget
Range: WarningLabel
InverseOf: isLabelOf

ObjectProperty: isLabelOf
Domain: WarningLabel
InverseOf: hasLabel

ObjectProperty: isButtonOf
InverseOf: hasButton

ObjectProperty: hasBattery
Domain: Widget
Range: Battery
InverseOf: isBatteryOf

ObjectProperty: hasButton
Domain: Widget

Range: Red_Button
InverseOf: isButtonOf

ObjectProperty: isShapeOf
InverseOf: hasShape

ObjectProperty: hasCasing
Characteristics: Functional
Domain: Widget
Range: Casing
InverseOf: isCaseOf
SubPropertyOf: hasCasing

ObjectProperty: isCaseOf
InverseOf: hasCasing

DataProperty: hasButtonFlag
Characteristics: Functional

DataProperty: hasHeight
SubPropertyOf: hasSize

DataProperty: hasSize
Annotations: rdfs:comment "Used just for categorisation."

Domain: Widget
Range: integer

DataProperty: hasLength
SubPropertyOf: hasSize

DataProperty: hasWidth
SubPropertyOf: hasSize

DataProperty: hasLabelFlag
Characteristics: Functional
Range: boolean

DataProperty: hasWaterproofCoating
Characteristics: Functional
Domain: Widget
Range: boolean

DataProperty: hasWeight
Domain: Widget
Range: integer

DataProperty: hasBatteryFlag
Characteristics: Functional
Range: boolean

Class: AA_battery
SubClassOf: Battery
DisjointWith: AAA_battery

Class: NotMetallic
SubClassOf: Casing
DisjointWith: Metallic

Class: Rectangular

SubClassOf: Shape
DisjointWith: Circular

```
Class: WarningLabel
Class: NotWaterproof
  SubClassOf: Coating
  DisjointWith: Waterproof

Class: Large
  SubClassOf: Size
  DisjointWith: Medium, Small

Class: Widget
  Annotations: rdfs:comment "NotWaterproof or Waterproof deleted
because it causes Widget to be a Subclass of Waterprofness."
  SubClassOf:
    hasBattery some Battery,
    hasLabel some WarningLabel,
    hasCasing only Casing,
    hasCoating only Coating,
    hasShape only Rectangular,
    hasBatteryFlag only boolean,
    hasButtonFlag only boolean,
    hasLabelFlag only boolean,
    hasLength only integer,
    hasWaterproofCoating only boolean

Class: AAA_battery
  SubClassOf: Battery
  DisjointWith: AA_battery

Class: Circular
  SubClassOf: Shape
  DisjointWith: Rectangular

Class: Size
  Annotations: rdfs:comment "Used just for categorisation."
  SubClassOf:
    Large or Medium or Small

Class: owl:Thing
Class: Battery
Class: Shape
  SubClassOf: owl:Thing, Circular or Rectangular

Class: Red_Button
  SubClassOf: Button

Class: Waterproof
  SubClassOf: Coating
  DisjointWith: NotWaterproof

Class: Medium
  Annotations: rdfs:comment "Defining hasBattery exactly 1 AA_Battery
causes the Widget to be inferred as a subclass of widget because
hasBattery has Widget as domain. "
  SubClassOf:
    Size,
    hasBattery exactly 1 AA_battery,
    hasButton exactly 1 Red_Button

  DisjointWith:
    Large,
    Small
```

Class: Small
 SubClassOf: Size
 DisjointWith: Medium, Large

Class: Button
Class: Casing
 SubClassOf: Metallic or NotMetallic

Class: Coating
Class: Metallic
 SubClassOf: Casing
 DisjointWith: NotMetallic

Individual: widget-TA102-1
 Types: Waterproof, Widget, owl:Thing
 Facts: hasWaterproofCoating "true"^^xsd:boolean

Individual: widget-TA100-1
 Types: Widget, owl:Thing
 Facts: hasShape Rectangular

Individual: Rectangular
 Types: Rectangular, owl:Thing

Individual: Circular
 Types: Circular, owl:Thing

Individual: Label_This_is_not_Waterproof
 Types: WarningLabel, owl:Thing

Individual: AA_Battery1
 Types: AA_battery, owl:Thing
 DifferentFrom: AA_Battery2

Individual: widget-TA104-1_and_widget-TA104-2
 Types: Widget, owl:Thing
 Facts: hasLength 6, hasWeight 200

Individual: widget-TA101-1b
 Types: Medium, owl:Thing
 Facts: hasButton red-Button

Individual: Button2
 Types: Button, owl:Thing

Individual: AA_Battery2
 Types: AA_battery, owl:Thing
 DifferentFrom: AA_Battery1

Individual: red-Button
 Types: Red_Button, owl:Thing

Individual: widget-TA101-1a
 Types: Medium, Widget, owl:Thing
 Facts: hasBatteryFlag "true"^^xsd:boolean

Individual: widget-TA103-1
 Types: Widget, owl:Thing
 Facts:
 hasCasing MetallicCase,
 hasWaterproofCoating "true"^^xsd:boolean

Individual: widget-TA102-2
 Types: Widget, owl:Thing

```
Facts:
  hasLabelFlag "true"^^xsd:boolean,
  hasWaterproofCoating "false"^^xsd:boolean
```

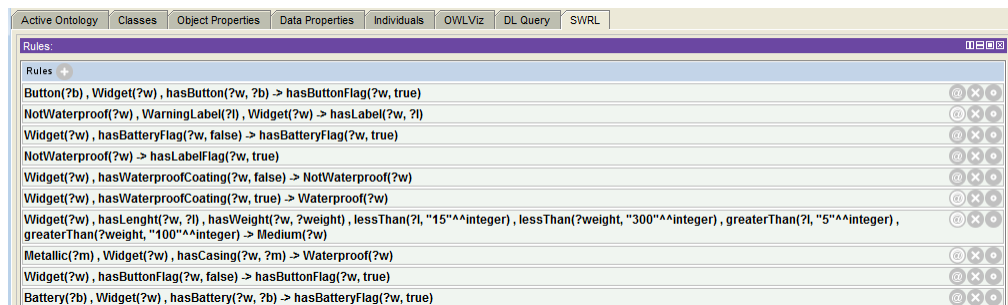
```
Individual: widget-TA104-2
  Types: Widget, owl:Thing
  Facts: hasLength 6
```

```
Individual: MetallicCase
  Types: Metallic, owl:Thing
```

```
Individual: widget-TA103-2
  Types: Widget, owl:Thing
  Facts:
    hasCasing MetallicCase,
    hasWaterproofCoating "true"^^xsd:boolean
```

```
Individual: AAA_Battery
  Types: AAA_battery, owl:Thing
```

```
Individual: widget-TA104-1
  Types: Widget, owl:Thing
  Facts: hasWeight 200
```



APPENDIX IV: The Greek e-GIF Ontology

```
Namespace: <http://www.semanticb2bi.eu/ontologies/GIC.owl#>
Namespace: hasTerms
<http://www.semanticb2bi.eu/ontologies/GIC.owl#hasTerms&
Namespace: Terms <http://www.semanticb2bi.eu/ontologies/GIC.owl#Terms&
Namespace: GIC <http://www.semanticb2bi.eu/ontologies/GIC.owl#>
Namespace: rdfs <http://www.w3.org/2000/01/rdf-schema#>
Namespace: eGIF <eGIF:>
Namespace: owl2xml <http://www.w3.org/2006/12/owl2-xml#>
Namespace: ElectronicServices
<http://www.semanticb2bi.eu/ontologies/GIC.owl#ElectronicServices&
Namespace: Structure
<http://www.semanticb2bi.eu/ontologies/GIC.owl#Structure&
Namespace: xsd <http://www.w3.org/2001/XMLSchema#>
Namespace: owl <http://www.w3.org/2002/07/owl#>
Namespace: rdf <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
Namespace: Ergonomics
http://www.semanticb2bi.eu/ontologies/GIC.owl#Ergonomics&
```

```
Ontology: <http://www.semanticb2bi.eu/ontologies/GIC.owl>
Import: <http://www.e-gif.gov.gr/eGIFOntology.owl>
```

```
Annotations:
rdfs:comment "
Developed by Bahareh Rahmanzadeh Heravi
Bahareh.Heravi@brunel.ac.uk
dc:creator "Bahareh Rahmanzadeh Heravi
Fluidity research Group
Department of Information Systems and Computing
Brunel University
Bahareh.Heravi@brunel.ac.uk"
```

```
ObjectProperty: hasWebsiteItem
```

```
Domain:
WebPage
or Website
```

```
Range:
WebsiteItems
```

```
ObjectProperty: hasStructure
```

```
Domain:
Website

Range:
Structure
```

```
ObjectProperty: hasContactForm
```

```
Domain:
Website

Range:
ContactForm
```

```
ObjectProperty: hasFax
```

Range:
Fax

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasLegislation

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasTerms:Conditions

Range:
Terms:Conditions

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasAddress

Range:
Address

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasAnnouncement

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasLogo

Range:
Logo

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasCounter

Range:
Counter

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasMetadata

Range:
Metadata

ObjectProperty: hasFAQ

Range:

FAQ

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasContactDetails

Domain:
ContactForm

Range:
ContactDetails

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasServicesFlowchart

Range:
ServicesFlowchart

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasName

Range:
OfficialName

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasTelephone

Range:
Telephone

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasWelcomeMessage

Range:
WelcomeMessage

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasMessage

Domain:
WebPage

Range:
Message

ObjectProperty: hasLogFile

Range:
LogFile

SubPropertyOf:
hasWebsiteItem
ObjectProperty: hasLink

Domain:
WebPage

Range:
eGIF:Service
or InformativeContent
or Tools
or WebPage
or Website

SubPropertyOf:
hasWebsiteItem

ObjectProperty: isInLanguage

Range:
ContentLanguage

ObjectProperty: hasComplaintForm

Range:
Complaints

ObjectProperty: hasEmail

Range:
Email

SubPropertyOf:
hasWebsiteItem

ObjectProperty: hasNews

SubPropertyOf:
hasWebsiteItem

Class: SearchEngine

SubClassOf:
Tools

Class: CompositeSearch

SubClassOf:
SearchEngine

Class: WelcomeMessage

SubClassOf:

Message

Class: ContentBackupPolicy

SubClassOf:
Content

Class: DigitalAuthentication

SubClassOf:
Components

Class: ServicesFlowchart

SubClassOf:
RepresentativeInformation

Class: Homepage

SubClassOf:
WebPage

Class: WebsiteManagement

SubClassOf:
Guidelines

Class: Flat

SubClassOf:
Structure

Class: DomainNaming

SubClassOf:
WebsiteStructure

Class: Content

SubClassOf:
Structure:Content

Class: Complaints

SubClassOf:
Content

Class: Email

SubClassOf:
ContactDetails

Class: eGIF:Public_Organization

Class: ContentLanguage

SubClassOf:
TobeCategorised

Class: MultipleWebsitesManagement

SubClassOf:
WebsiteStructure

Class: UnderConstruction

SubClassOf:
Message,
Status

Class: Assessment

SubClassOf:
Guidelines

Class: ClassificationScheme

SubClassOf:
Structure

Class: FAQ

SubClassOf:
InformativeContent

Class: TobeCategorised

SubClassOf:
InteroperabilityFramework

Class: Structure:Content

SubClassOf:
WebsiteStructure

Class: WebsiteDissemination

SubClassOf:
Guidelines

Class: Components

SubClassOf:
InteroperabilityFramework

Class: InformativeContent

SubClassOf:

WebsiteItems

Class: Contact

SubClassOf:
WebPage

Class: DocumentationModel

SubClassOf:
Components

Class: Status

SubClassOf:
ToBeCategorised

Class: DocumentFormat

SubClassOf:
ToBeCategorised

Class: Terms:Conditions

SubClassOf:
InformativeContent

Class: InteroperabilityRegistry

SubClassOf:
Components

Class: Ergonomics:VisualStyle

SubClassOf:
WebsiteStructure

Class: GeneralPrinciples

SubClassOf:
Guidelines

Class: Active

SubClassOf:
Status

Class: Website

SubClassOf:
eGIF:Web_Information_System

Class: WebsiteCounter

SubClassOf:
Counter

Class: InteroperabilityFramework

SubClassOf:
owl:Thing

Class: Address

SubClassOf:
ContactDetails

Class: WebsiteStructure

SubClassOf:
Guidelines

Class: Logo

SubClassOf:
WebsiteItems

Class: InstitutionsAdministrationInformation

SubClassOf:
RepresentativeInformation

Class: OtherStorage

SubClassOf:
ContentStorage

Class: Tree

SubClassOf:
Structure

Class: LogFile

SubClassOf:
WebsiteItems

Class: Fax

SubClassOf:
ContactDetails

Class: ContentOwner

EquivalentTo:
ContentAuthor

SubClassOf:
Role

Class: Legislation

SubClassOf:
RepresentativeInformation

Class: Security

SubClassOf:
Guidelines

Class: SchemaRegisrty

SubClassOf:
Components

Class: eGIF:Service

Class: E-ServiceCounter

SubClassOf:
Counter

Class: GCL

Annotations:
rdfs:comment "Government Category List"

SubClassOf:
ClassificationScheme

Class: KeywordSearch

SubClassOf:
SearchEngine

Class: LegalIssues

SubClassOf:
Guidelines

Class: SiteMap

SubClassOf:
Tools

Class: Counter

SubClassOf:
WebsiteItems

Class: ContentStorage

SubClassOf:
TobeCategorised

Class: Tools
 SubClassOf:
 WebsiteItems

Class: eGIF:Web_Information_System

Class: Guidelines
 SubClassOf:
 CertificationFramework

Class: Announcements
 SubClassOf:
 InformativeContent

Class: Message
 SubClassOf:
 WebsiteItems

Class: Link
 SubClassOf:
 WebsiteItems

Class: Telephone
 SubClassOf:
 ContactDetails

Class: RepresentativeInformation
 SubClassOf:
 WebsiteItems

Class: Accessibility
 SubClassOf:
 WebsiteStructure

Class: News
 SubClassOf:
 InformativeContent

Class: ContentAuthor
 EquivalentTo:
 ContentOwner

 SubClassOf:
 Role

Class: ServiceContactPoint
 SubClassOf:
 Role

Class: XML

```
SubClassOf:
  DocumentFormat

Class: ContactDetails

SubClassOf:
  WebsiteItems

Class: ThematicCategories

SubClassOf:
  ClassificationScheme

Class: ServicesInformation

SubClassOf:
  RepresentativeInformation

Class: WebPage

SubClassOf:
  WebsiteItems

Class: WebsiteItems

SubClassOf:
  Content

Class: Metadata

SubClassOf:
  TobeCategorised

Class: OfficialName

SubClassOf:
  WebsiteItems

Class: ServiceProvision

SubClassOf:
  Components

Class: Introduction

Annotations:
  rdfs:comment "Introduction/ShortPresentation"

SubClassOf:
  RepresentativeInformation

Class: RelationalDB

SubClassOf:
  ContentStorage

Class: owl:Thing

Class: ContactForm

SubClassOf:
  WebsiteItems
```

Class: CertificationFramework

SubClassOf:
Components

Class: OtherWebsites

SubClassOf:
Website

Class: ElectronicServices:Interoperability

SubClassOf:
Guidelines

Class: Institution

SubClassOf:
eGIF:Public_Organization

Class: Role

SubClassOf:
WebsiteManagement

Class: Structure

SubClassOf:
Structure:Content

Class: PublicInstitution

SubClassOf:
Institution

Individual: English

Types:
ContentLanguage,
owl:Thing

Individual: Greek

Types:
ContentLanguage,
owl:Thing

Individual: OtherLanguage

Types:
ContentLanguage,
owl:Thing

APPENDIX V: The OASIS ebBP Ontology

Namespace: <<http://www.semantic-b2bi/ontologies/BH-ebBP.owl#>>
Namespace: rdfs <<http://www.w3.org/2000/01/rdf-schema#>>

Namespace: owl2xml <<http://www.w3.org/2006/12/owl2-xml#>>
Namespace: owl <<http://www.w3.org/2002/07/owl#>>
Namespace: xsd <<http://www.w3.org/2001/XMLSchema#>>
Namespace: rdf <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>>
Namespace: BH-ebBP <<http://www.semantic-b2bi/ontologies/BH-ebBP.owl#>>

Ontology: <<http://www.semantic-b2bi/ontologies/BH-ebBP.owl>>

Annotations:

rdfs:comment "ebXML Business Process (ebBP) Ontology - Version 1.0.0

Developed By Bahareh Rahmazadeh Heravi (Bahareh.Heravi@brunel.ac.uk) working under contract to and with the support of Fujitsu Laboratories of Europe Limited.

All intellectual property and proprietary rights in the Ontology belong to Fujitsu Laboratories of Europe Limited.

The ebBP Ontology provides a semantic formalisation of the structural components of the OASIS ebXML Business Process Specification Schema Technical Specification (ebBP), based on the latest ebBP specifications [ebBP v2.0.4 - December 2006].

This work is undertaken in collaboration with Stephen D. Green from Document Engineering Services.

Copyright © 2010 Fujitsu Laboratories of Europe Limited."

ObjectProperty: hasBusinessPartnerRole

Domain:
ExternalRoles

Range:
BusinessPartnerRole

ObjectProperty: isRealisedByBusinessCollaboration

Domain:
ProcessSpecification

Range:
BusinessCollaboration

InverseOf:
isRealisationOfProcessSpecification

ObjectProperty: hasCollaborationActivity

Domain:
BinaryCollaboration
or BusinessCollaboration
or MultiPartyCollaboration

Range:
CollaborationActivity

ObjectProperty: hasAttributeSubstitution

Domain:
Package
or ProcessSpecification

Range:
AttributeSubstitution

ObjectProperty: hasSpecification

Domain:
BusinessDocument
or Signal

Range:
Specification

ObjectProperty: isReceiptAcknowledgementExceptionOf

Domain:
ReceiptAcknowledgementException

Range:
BusinessAction
or RequestingBusinessActivity
or RespondingBusinessActivity

InverseOf:
hasReceiptAcknowledgementException

SubPropertyOf:
isSignalEnvelopeTypeOf

ObjectProperty: isAcceptanceAcknowledgementExceptionOf

Domain:
AcceptanceAcknowledgementException

Range:
BusinessAction
or RequestingBusinessActivity
or RespondingBusinessActivity

InverseOf:
hasAcceptanceAcknowledgementException

SubPropertyOf:
isSignalEnvelopeTypeOf

ObjectProperty: hasDocumentation

Range:

Documentation

ObjectProperty: isRequestingBusinessActivityOf

Domain:
RequestingBusinessActivity

Range:
BusinessTransaction

InverseOf:
hasRequestingBusinessActivity

SubPropertyOf:
isBusinessActionOf

ObjectProperty: isPerformsOf

Domain:
Performs

Range:
BusinessPartnerRole
or BusinessTransactionActivity
or CollaborationActivity

ObjectProperty: hasProtocolFailure

SubPropertyOf:
hasFailure

ObjectProperty: refersToBusinessTransaction

Domain:
BusinessTransactionActivity

Range:
BusinessTransaction

InverseOf:
isRealisedByBusinessTransactionActivity

ObjectProperty: hasBusinessAction

Domain:
BusinessTransaction

Range:
RequestingBusinessActivity
or RespondingBusinessActivity

InverseOf:
isBusinessActionOf

ObjectProperty: hasReceiptAcknowledgement

Range:

ReceiptAcknowledgement

InverseOf:
isReceiptAcknowledgementOf

SubPropertyOf:
hasSignal

ObjectProperty: hasPostCondition

Domain:
BinaryCollaboration
or BusinessCollaboration
or BusinessTransactionActivity
or MultiPartyCollaboration

Range:
ConditionExpression

ObjectProperty: isBusinessActionOf

InverseOf:
hasBusinessAction

ObjectProperty: EndsWhen

Domain:
BinaryCollaboration
or BusinessCollaboration
or BusinessTransactionActivity
or MultiPartyCollaboration

Range:
ConditionExpression

ObjectProperty: hasPreCondition

Domain:
BinaryCollaboration
or BusinessCollaboration
or BusinessTransactionActivity
or MultiPartyCollaboration

Range:
ConditionExpression

ObjectProperty: hasExternalRoles

Domain:
Package
or ProcessSpecification

Range:
ExternalRoles

ObjectProperty: hasBusinessTransactionActivity

Domain:
BinaryCollaboration
or BusinessCollaboration
or ComplexBusinessTransactionActivity
or MultiPartyCollaboration

Range:
BusinessTransactionActivity

InverseOf:
isBusinessTransactionActivityOf

ObjectProperty: hasDocumentEnvelope

Domain:
Attachment
or Package
or ProcessSpecification

Range:
DocumentEnvelope

InverseOf:
isDocumentEnvelopeOf

ObjectProperty: hasRespondingRole

Domain:
BusinessTransaction

Range:
RespondingRole

ObjectProperty: hasRole

Domain:
BusinessCollaboration
or OperationMapping

Range:
Role

InverseOf:
isRoleOf

ObjectProperty: hasComplexBusinessTransactionActivity

Domain:
BinaryCollaboration
or BusinessCollaboration
or ComplexBusinessTransactionActivity
or MultiPartyCollaboration

Range:
ComplexBusinessTransactionActivity

ObjectProperty: isBusinessTransactionActivityOf

Domain:

BusinessTransactionActivity

Range:

BinaryCollaboration
or BusinessCollaboration
or ComplexBusinessTransactionActivity
or MultiPartyCollaboration

InverseOf:

hasBusinessTransactionActivity

ObjectProperty: isRealisationOfProcessSpecification

InverseOf:

isRealisedByBusinessCollaboration

ObjectProperty: hasAttachment

Domain:

DocumentEnvelope

Range:

Attachment

InverseOf:

isAttachmentOf

ObjectProperty: hasFailure

Domain:

BinaryCollaboration
or BusinessCollaboration
or MultiPartyCollaboration

Range:

Failure

ObjectProperty: isAttachmentOf

Domain:

Attachment

Range:

DocumentEnvelope

InverseOf:

hasAttachment

ObjectProperty: hasState

Domain:

BusinessCollaboration

Range:

State

ObjectProperty: hasFromBusinessState

Domain:
BusinessActivity
or LinkingConstructs

Range:
FromLink

ObjectProperty: hasRequestingRole

Domain:
BusinessTransaction

Range:
RequestingRole

ObjectProperty: hasMultiPartyCollaboration

Domain:
Package
or ProcessSpecification

Range:
MultiPartyCollaboration

SubPropertyOf:
hasBusinessCollaboration

ObjectProperty: hasTimeToPerform

Domain:
BusinessCollaboration,
BusinessTransactionActivity,
Fork

Range:
TimeToPerform

ObjectProperty: isRealisedByBusinessTransactionActivity

Domain:
BusinessTransaction

Range:
BusinessTransactionActivity

InverseOf:
refersToBusinessTransaction

ObjectProperty: isSignalOf

Domain:
Signal

Range:
Package
or ProcessSpecification

InverseOf:
hasSignal

ObjectProperty: hasPerformsRole

Domain:
Performs

Range:
Role

ObjectProperty: hasToBusinessState

Domain:
BinaryCollaboration
or LinkingConstructs

Range:
BusinessStateLinkType

ObjectProperty: isRespondingBusinessActivityOf

Domain:
RespondingBusinessActivity

Range:
BusinessTransaction

InverseOf:
hasRespondingBusinessActivity

SubPropertyOf:
isBusinessActionOf

ObjectProperty: hasBusinessFailure

SubPropertyOf:
hasFailure

ObjectProperty: hasReceiptAcknowledgementException

Range:
ReceiptAcknowledgementException

InverseOf:
isReceiptAcknowledgementExceptionOf

SubPropertyOf:
hasSignal

ObjectProperty: hasConditionExpression

Domain:
BusinessDocument
or BusinessStateLinkType
or Signal
or Transition
or Variable

Range:

ConditionExpression

ObjectProperty: hasRespondingBusinessActivity

Domain:
BusinessTransaction

Range:
RespondingBusinessActivity

InverseOf:
isRespondingBusinessActivityOf

SubPropertyOf:
hasBusinessAction

ObjectProperty: hasRequestingBusinessActivity

Domain:
BusinessTransaction

Range:
RequestingBusinessActivity

InverseOf:
isRequestingBusinessActivityOf

SubPropertyOf:
hasBusinessAction

ObjectProperty: hasDecision

Domain:
BusinessCollaboration

Range:
Decision

ObjectProperty: isSignalEnvelopeTypeOf

ObjectProperty: hasAcceptanceAcknowledgementException

Range:
AcceptanceAcknowledgementException

InverseOf:
isAcceptanceAcknowledgementExceptionOf

SubPropertyOf:
hasSignal

ObjectProperty: hasVariable

Domain:
Package
or ProcessSpecification
or TimeToPerform

Range:
Variable

ObjectProperty: hasSuccess

Domain:
BinaryCollaboration
or BusinessCollaboration
or MultiPartyCollaboration

Range:
Success

ObjectProperty: hasBusinessTransaction

Domain:
OperationMapping
or Package
or ProcessSpecification

Range:
BusinessTransaction
or CommercialTransaction

InverseOf:
isBusinessTransactionOf

ObjectProperty: isDocumentEnvelopeOf

Domain:
DocumentEnvelope

Range:
Attachment
or Package
or ProcessSpecification

InverseOf:
hasDocumentEnvelope

ObjectProperty: hasParent

Annotations:
owl:priorVersion "@parentRef"

Domain:
Package

Range:
Package

ObjectProperty: isRoleOf

InverseOf:
hasRole

ObjectProperty: hasOperationMapping

Domain:
Package
or ProcessSpecification

Range:
OperationMapping

ObjectProperty: hasCurrentRole

Domain:
Performs

Range:
Role

ObjectProperty: hasBusinessSuccess

SubPropertyOf:
hasSuccess

ObjectProperty: hasSignal

Domain:
BusinessAction

Range:
Signal

InverseOf:
isSignalOf

ObjectProperty: hasPerforms

Domain:
BusinessPartnerRole
or BusinessTransactionActivity
or CollaborationActivity

Range:
Performs

ObjectProperty: hasBusinessCollaboration

Domain:
Package
or ProcessSpecification

Range:
BusinessCollaboration

ObjectProperty: hasStart

Domain:
BinaryCollaboration,
BusinessCollaboration,
MultiPartyCollaboration,
BinaryCollaboration
or BusinessCollaboration

or MultiPartyCollaboration

Range:
Start

ObjectProperty: hasBusinessActivity

Domain:
BusinessCollaboration

Range:
BusinessActivity

ObjectProperty: hasAcceptanceAcknowledgement

Range:
AcceptanceAcknowledgement

InverseOf:
isAcceptanceAcknowledgementOf

SubPropertyOf:
hasSignal

ObjectProperty: hasProtocolSuccess

SubPropertyOf:
hasSuccess

ObjectProperty: hasPackage

Domain:
Package
or ProcessSpecification

Range:
Package

ObjectProperty: isAcceptanceAcknowledgementOf

Domain:
AcceptanceAcknowledgement

Range:
BusinessAction
or RequestingBusinessActivity
or RespondingBusinessActivity

InverseOf:
hasAcceptanceAcknowledgement

SubPropertyOf:
isSignalEnvelopeTypeOf

ObjectProperty: BeginsWhen

Domain:
BinaryCollaboration
or BusinessCollaboration
or BusinessTransactionActivity
or MultiPartyCollaboration

Range:
ConditionExpression

ObjectProperty: hasBinaryCollaboration

Domain:
Package
or ProcessSpecification

Range:
BinaryCollaboration

SubPropertyOf:
hasBusinessCollaboration

ObjectProperty: hasBusinessDocument

Domain:
DocumentEnvelope

Range:
BusinessDocument

InverseOf:
isBusinessDocumentOf

ObjectProperty: isBusinessTransactionOf

Domain:
BusinessTransaction
or CommercialTransaction

Range:
Package
or ProcessSpecification

InverseOf:
hasBusinessTransaction

ObjectProperty: isBusinessDocumentOf

Domain:
BusinessDocument

Range:
DocumentEnvelope

InverseOf:
hasBusinessDocument

ObjectProperty: isReceiptAcknowledgementOf

Domain:

ReceiptAcknowledgement

Range:
BusinessAction
or RequestingBusinessActivity
or RespondingBusinessActivity

InverseOf:
hasReceiptAcknowledgement

SubPropertyOf:
isSignalEnvelopeTypeOf

DataProperty: hasUuid

Annotations:
rdfs:comment "Defines a string identification mechanism for a Process Specification. The uuid is not used for the purpose of versioning, so that even a change introduced by AttributeSubstitution (to business documents' schemas, for example), would be marked by a new uuid."

Domain:
ProcessSpecification

Range:
string

DataProperty: hasBusinessTransactionBaseTypePattern

Range:
anyURI

DataProperty: isPositiveResponse

Annotations:
rdfs:comment "May evaluate to TRUE or FALSE. If TRUE, the DocumentEnvelope is intended as a positive response to a request. The value for this parameter is used to evaluate a Business Success or Failure of the corresponding Business Transaction."

Domain:
AcceptanceAcknowledgement,
AcceptanceAcknowledgementException,
DocumentEnvelope,
ReceiptAcknowledgement,
ReceiptAcknowledgementException

Range:
boolean

DataProperty: hasOperationName

Domain:
MessageMap,
SignalMap

Range:
NMTOKEN

DataProperty: hasExpressionLanguage

Annotations:
 rdfs:comment "Defines the language used for the Condition Expression."

Domain:
 ConditionExpression

Range:
 NMToken

DataProperty: hasInstanceVersion

Domain:
 ProcessSpecification

Range:
 string

DataProperty: hasName

Range:
 string

DataProperty: hasMimeType

Domain:
 Attachment

Range:
 string

DataProperty: hasSpecificationVersion

Annotations:
 rdfs:comment "The technical specification version of the Process Specification."

Domain:
 ProcessSpecification

Range:
 NMToken

DataProperty: isConfidential

Annotations:
 rdfs:comment "Transient confidentiality is provided by a secure network protocol, such as SSL as the document is transferred between two adjacent ebXML Messaging Service or other transport messaging nodes. Persistent confidentiality is intended to preserve the confidentiality of the message such that only the intended party (application) can see it."

Domain:
 Attachment,
 DocumentEnvelope

Range:
NMToken

SubPropertyOf:
DocumentSecurity

DataProperty: isNonRepudiationRequired

Annotations:
rdfs:comment "If non-repudiation of origin and content is required, then the Business Activity stores the business document in its original form for the duration mutually agreed to in an agreement."

Domain:
RequestingBusinessActivity,
RespondingBusinessActivity

Range:
boolean

SubPropertyOf:
Non-Repudiation

DataProperty: Timeouts

DataProperty: hasValue

Annotations:
rdfs:comment "Is the value, which shall replace the current value of the attribute."

Domain:
AttributeSubstitution

Range:
string

DataProperty: hasAttributeName

Annotations:
rdfs:comment "Is the name of an attribute of any element within the scope of the substitution set."

Domain:
AttributeSubstitution

Range:
NMToken

DataProperty: hasDocumentationLang

Domain:
Documentation

Range:
language

DataProperty: isAuthenticated

Annotations:

rdfs:comment "The communications channel used to transport the Message provides transient authentication. The specific method will be determined by the communications protocol used. Persistent authentication means the Business Document signer's identity is verified at the receiving application level.

Authentication assists in verification of role identity of a participating party."

Domain:

Attachment,
DocumentEnvelope

Range:

NMToken

SubPropertyOf:

DocumentSecurity

DataProperty: hasDuration

Domain:

TimeToPerform

Range:

duration

DataProperty: hasType

Domain:

Specification

Range:

NMToken

DataProperty: isNonRepudiationReceiptRequired

Annotations:

rdfs:comment "Both parties agree to mutually verify receipt of a Requesting Business Document and that the receipt is non-repudiable."

Domain:

RequestingBusinessActivity,
RespondingBusinessActivity

Range:

boolean

SubPropertyOf:

Non-Repudiation

DataProperty: Non-Repudiation

DataProperty: hasOperationStep

Domain:
MessageMap,
SignalMap

DataProperty: hasInterfaceName

Domain:
MessageMap,
SignalMap

Range:
NMToken

DataProperty: hasNameID

Annotations:
rdfs:comment "Is the nameID reference to the Documentation
related to a particular element."

Range:
string

DataProperty: waitForAll

Annotations:
rdfs:comment "Indicates that all transitions coming into the
Join are executed in order for the Business Collaboration to reach the
Join state (AND-join). By default, the Join is an AND-join."

Domain:
Join

Range:
boolean

DataProperty: hasDefaultValue

Domain:
ConditionExpression

Range:
string

DataProperty: hasBusinessColaborationPattern

Domain:
BusinessCollaboration

Range:
anyURI

DataProperty: hasConditionGuard

Annotations:
rdfs:comment "The condition that guards the transition from a
Business State."

Domain:
FromLink

Range:
NMToken

DataProperty: hasTargetNamespace

Annotations:
rdfs:comment "The target namespace of the Specification of the particular ebBP element."

Domain:
Specification

Range:
anyURI

DataProperty: hasRetryCount

Annotations:
rdfs:comment "The business retry for a RequestingBusinessActivity identifies the number of retries allowed in addition to the initial request while the TimeToPerform has not been exceeded."

Domain:
BusinessAction

Range:
int

DataProperty: hasURI

Annotations:
rdfs:comment "Defines the address of the Documentation object. A URL can be a URI."

Domain:
Documentation

Range:
anyURI

DataProperty: isGuaranteedDeliveryRequired

Domain:
BusinessTransaction

Range:
boolean

DataProperty: hasExpression

Annotations:
rdfs:comment "Defines the value for the Condition Expression."

Domain:

ConditionExpression

Range:
string

DataProperty: DocumentSecurity

DataProperty: hasLocation

Annotations:
rdfs:comment "The location of the Specification of the particular ebBP element."

Domain:
Specification

Range:
anyURI

DataProperty: isIntelligibleCheckRequired

Annotations:
rdfs:comment "Allows partners to agree that a message is confirmed by a Receipt Acknowledgement only if it is also legible. Legible means that it has passed structure/schema validity check. The content of the receipt and the legibility of a business message (if required) are reviewed prior to the processing of the Business Document or the evaluation of Condition Expressions in the business message's Business Documents or Document Envelope."

Domain:
RequestingBusinessActivity,
RespondingBusinessActivity

Range:
boolean

DataProperty: hasPattern

Domain:
BinaryCollaboration,
BusinessCollaboration,
BusinessTransaction,
MultiPartyCollaboration

Range:
anyURI

DataProperty: isTamperDetectable

Annotations:
rdfs:comment "Transient isTamperDetectable is the ability to detect if the information has been tampered with during transfer between two adjacent Message Service Handler nodes. Persistent isTamperDetectable is the ability to detect if the information has been tampered with after it has been received by messaging node, between the messaging node and the application. Tamper detection assists in verification of content integrity between and within a participating party."

Domain:
Attachment,
DocumentEnvelope

Range:
NMToken

SubPropertyOf:
DocumentSecurity

DataProperty: hasLegalIntent

Annotations:
rdfs:comment "This attribute is optional and means that particular activity that could represents a statement or commitment between trading partners, and their shared intent."

The hasLegalIntent attribute could have widely differing interpretations and enforceability depending on type of business, process, and jurisdiction. No implication of interpretation or enforceability is made by the ebBP specification. The implementer SHOULD NOT assume any particular runtime behavior based on this attribute."

Domain:
BusinessTransactionActivity

Range:
boolean

DataProperty: hasTimeToAcknowledgeAcceptance

Annotations:
rdfs:comment "The time a Responding or Requesting role has to non-substantively acknowledge business acceptance of a Business Document."

Domain:
AcceptanceAcknowledgement

Range:
duration

SubPropertyOf:
Timeouts

DataProperty: hasTimeToAcknowledgeReceipt

Annotations:
rdfs:comment "The time a Responding or Requesting role has to acknowledge receipt of a Business Document."

Domain:
ReceiptAcknowledgement

Range:
duration

SubPropertyOf:
Timeouts

DataProperty: isConcurrent

Annotations:

rdfs:comment "It determines whether at run-time multiple instances of that BTA can be 'open' at the same time within any Business Collaboration instance performed between any parties."

isConcurrent limits the ability to execute multiple BTA of the same BT across Business Collaboration instances (with the same party), or within the same Business Collaboration if multiple paths are open."

Domain:

BusinessTransactionActivity

Range:

boolean

DataProperty: isInnerCollaboration

Annotations:

rdfs:comment "Indicates whether or not this Business Collaboration definition can only be used within a Collaboration Activity (as a sub collaboration) or initiated directly by a party."

Domain:

BinaryCollaboration,
BusinessCollaboration,
MultiPartyCollaboration

Range:

boolean

DataProperty: hasBusinessTansactionHeadPattern

Range:

anyURI

DataProperty: isAuthorizationRequired

Annotations:

rdfs:comment "When a party uses isAuthorizationRequired on a Requesting and/or a Responding activity accordingly, the result that [the activity] will only be processed as valid if the party interpreting it successfully matches the stated identity of the activity's [Role] to a list of allowed values previously supplied by that party. Authorization typically relates to a signed business document and the association to the role identity of the party expected for that activity."

Domain:

RequestingBusinessActivity,
RespondingBusinessActivity

Range:

boolean

DataProperty: hasBinaryCollaborationTypePattern

Domain:

BinaryCollaboration

Range:
anyURI

DataProperty: hasTimeToPerformType

Domain:
TimeToPerform

Range:
NMToken

DataProperty: hasGuard

Domain:
Transition

Range:
string

Class: Success

Annotations:

rdfs:comment "A "Success" end state (Protocol or Business) is dependent on receipt of a Business Document satisfying the associated TimeToPerform."

In order for a BTA instance to reach a "Success" state at run-time, the following things SHOULD be true:

- no timeout would have occurred (signals or response)
- no signal can have a negative content
- the response document sent to the requester MUST be marked as isPositiveResponse = 'true' in the ebBP instance that specifies the Business Collaboration in order to support Business Success

Conversely, if all signals are positive and sent and received on time, the transaction will be successful from a protocol perspective.

In order to achieve a Success state, a BTA MUST complete with both a Protocol and a Business Success."

EquivalentTo:
(hasBusinessSuccess some BusinessSuccess)
and (hasProtocolSuccess some ProtocolSuccess)

SubClassOf:
CompletionState

DisjointWith:
BusinessTransaction

Class: AuthorizationException

Annotations:

rdfs:comment "Roles are not authorized to participate in the BTA. Note that the receiving BSI can only identify this exception.

Informative"

 SubClassOf:
 ReceiptAcknowledgementException

Class: ConditionExpression

 Annotations:
 rdfs:comment "An expression element that can be evaluated and provide a TRUE or FALSE."

 SubClassOf:
 owl:Thing

 DisjointWith:
 BusinessTransaction

Class: QueryResponse

 Annotations:
 rdfs:comment "Used by a Requester for an information query of which the responding party already has."

 SubClassOf:
 BusienssTransactionPattern

Class: SignalMap

 SubClassOf:
 owl:Thing

 DisjointWith:
 AttributeSubstitution,
 BusinessTransaction

Class: BusinessTransaction

 Annotations:
 rdfs:comment "A Business Transaction represents an atomic unit of work that may be associated with a trading arrangement between two business partners.

A Business Transaction is conducted between two parties playing opposite abstract roles in that transaction. Each party, as an abstract partner, assumes an abstract role in a Business Transaction."

 EquivalentTo:
 CommercialTransaction

 SubClassOf:
 BusienssTransactionPattern,
 ((hasState some Failure)
 or (hasState some Success))
 and (hasState some CompletionState),
 (hasBusinessDocument min 1 BusinessDocument)
 or (hasBusinessDocument max 2 BusinessDocument),
 hasSignal only Signal,


```
    hasRequestingBusinessActivity exactly 1
RequestingBusinessActivity,
    hasRequestingRole exactly 1 RequestingRole,
    hasRespondingBusinessActivity exactly 1
RespondingBusinessActivity,
    hasRespondingRole exactly 1 RespondingRole
```

```
DisjointWith:
    MessageMap,
    Success,
    ConditionExpression,
    SignalMap,
    StatusVisibility,
    ToLink,
    OperationMapping,
    RequestingBusinessActivity,
    Role,
    ExternalRoles,
    Fork,
    Performs,
    CollaborationGroup,
    Join,
    BusinessAction,
    Failure,
    BusinessStateLinkType,
    AcceptanceAcknowledgementException,
    MultiPartyCollaboration,
    BusinessPartnerRole,
    AcceptanceAcknowledgement,
    Decision,
    BinaryCollaboration,
    OptName,
    Attachment,
    ReceiptAcknowledgementException,
    ReceiptAcknowledgement,
    BusinessCollaboration,
    ComplexBusinessTransactionActivity,
    CollaborationActivity,
    Documentation,
    Signal,
    FromLink,
    DocumentEnvelope,
    Package,
    Variable,
    AttributeSubstitution,
    BusinessActivity,
    ProcessSpecification,
    Transition,
    RespondingRole,
    BusinessDocument,
    RespondingBusinessActivity,
    Start,
    Specification,
    RequestingRole,
    BusinessTransactionActivity
```

Class: StatusVisibility

Annotations:

rdfs:comment "Information (which can be aggregated) returned by the subparties of an embedded Business Transaction Activity or ComplexBTA for visibility purposes to the outermost ComplexBTA. For example, a subparty (requester in an embedded BTA that is responder in ComplexBTA) returns aggregated supplier information to the ComplexBTA prior to the responder issuing an order response. The Status Visibility element specifies which status values and which Document Envelope events of the embedded processes are considered, if any, when returning the status value to the context of the ComplexBTA. If no status values or DocumentEnvelope events can be monitored, then both BusinessDocumentList and SubstateVisibility are omitted."

 SubClassOf:
 owl:Thing

 DisjointWith:
 BusinessTransaction

Class: ToLink

 Annotations:
 rdfs:comment "A linking construct that indicates states that the current context (containing element) can transition to."

 SubClassOf:
 BusinessStateLinkType,
 hasConditionExpression only ConditionExpression

 DisjointWith:
 BusinessTransaction

Class: OR-Join

 Annotations:
 rdfs:comment "If one or more Business Activities complete, the OR-Join completes."

waitForAll="False"

 SubClassOf:
 Join

Class: ExternalRoles

 Annotations:
 rdfs:comment "External role element maps to the actual roles used in a Business Collaboration."

 SubClassOf:
 owl:Thing

 DisjointWith:
 BusinessTransaction

Class: Role

SubClassOf:
owl:Thing

DisjointWith:
BusinessTransaction

Class: BusinessFailure

SubClassOf:
Failure

Class: Performs

Annotations:
rdfs:comment "Performs elements are required whenever referencing the RequestingBusinessActivity or RespondingBusinessActivity in a BTA or within the BTAs of a ComplexBTA. Also Performs elements are required when the Role values in a referring context differ from or need to be switched between the Role values in the referenced context."

SubClassOf:
owl:Thing,
hasCurrentRole exactly 1 Role,
hasPerformsRole exactly 1 Role

DisjointWith:
BusinessTransaction

Class: CollaborationGroup

Annotations:
rdfs:comment "The group that includes the various types of Collaborations. Note: The Business Collaboration will replace the Binary and MultiParty Collaboration in a future version. Note: This group was added in v2.0."

SubClassOf:
owl:Thing

DisjointWith:
BusinessTransaction

Class: Join

Annotations:
rdfs:comment "A choreography construct that defines the point where one or more forked activities join. Can define that the completion of all state occur."

SubClassOf:
LinkingConstructs

DisjointWith:
BusinessTransaction

Class: Failure

```
Annotations:
  rdfs:comment "Defines a failure completion of a Business
Collaboration as a transition from an activity."

SubClassOf:
  CompletionState,
  (hasBusinessFailure only BusinessFailure)
  or (hasProtocolFailure only AnyProtocolFailure)

DisjointWith:
  BusinessTransaction

Class: ComplexVariable

SubClassOf:
  Variable

Class: SequenceException

Annotations:
  rdfs:comment "The order or type of a Business Document or
Business Signal is incorrect.

Informative"

SubClassOf:
  ReceiptAcknowledgementException

Class: SignatureException

Annotations:
  rdfs:comment "Business Documents are not signed for non-
repudiation when required.

Informative"

SubClassOf:
  ReceiptAcknowledgementException

Class: ResponseTimeout

SubClassOf:
  AnyProtocolFailure

Class: SyntaxException

Annotations:
  rdfs:comment "There is invalid punctuation, vocabulary or
grammar in the Business Document or Business Signal.

Informative"

SubClassOf:
  ReceiptAcknowledgementException

Class: XOR
```

Annotations:

rdfs:comment "An XOR Fork means that only one Business State of the Fork will be allowed to be reached, although all transitions to Business States are possible at the start. Once one of the outgoing transitions attached to the Fork gateway get activated, all the other transitions becomes invalid (e.g. a BTA starts)."

SubClassOf:

Fork

Class: RequestResponse

Annotations:

rdfs:comment "Used when an initiating party requests information that a responding party already has and when the request for business information requires a complex interdependent set of results."

SubClassOf:

BusienssTransactionPattern

Class: BusinessException

Annotations:

rdfs:comment "The business rules of the Responding activity are violated. The application refused to process the incoming Business Document. Most often because it violated some pre-processing business rules.

Informative"

SubClassOf:

AcceptanceAcknowledgementException

Class: Attachment

Annotations:

rdfs:comment "An optional unstructured document associated with a Business Document."

SubClassOf:

owl:Thing

DisjointWith:

BusinessTransaction

Class: CommercialTransaction

Annotations:

rdfs:comment "Formal obligation between parties"

EquivalentTo:

BusinessTransaction

SubClassOf:

BusienssTransactionPattern

Class: BusinessCollaboration

Annotations:
 rdfs:comment "A Business Collaboration is a set of Business Activities executing Business Transactions between business partners or collaborating parties. Each business partner plays one or more abstract partner roles in the Business Collaboration."

SubClassOf:
 owl:Thing,
 hasState some State,
 hasBusinessActivity min 1 BusinessActivity,
 hasRole exactly 2 Role

DisjointWith:
 AttributeSubstitution,
 BusinessTransaction,
 BusinessTransactionActivity

Class: PerformanceException

Annotations:
 rdfs:comment "The requested Business Action cannot be performed. The application MAY NOT be available.

Informative"

SubClassOf:
 AcceptanceAcknowledgementException

Class: DocumentEnvelope

Annotations:
 rdfs:comment "Conveys business information between two roles in a business transaction. One document envelope conveys the request from the Requesting to the Responding role and another the response from the Responding role back to the Requesting one (where applicable)."

SubClassOf:
 owl:Thing,
 hasAttachment only Attachment,
 hasBusinessDocument exactly 1 BusinessDocument

DisjointWith:
 BusinessTransaction

Class: AnyProtocolFailure

Annotations:
 rdfs:comment "AnyProtocolFailure is designed to allow the protocol to catch and handle behavior when the protocol fails because of technical failure."

SubClassOf:
 Failure

Class: Package

Annotations:
 rdfs:comment "Defines a hierarchical name scope containing reusable elements."

SubClassOf:
 owl:Thing

DisjointWith:
 BusinessTransaction,
 BusinessTransactionActivity

Class: Notification

Annotations:
 rdfs:comment "Used for business notifications such as a Notification of Failure Business Transaction in line with a Commercial Transaction pattern. Represents a formal exchange between parties"

SubClassOf:
 BusienssTransactionPattern,
 hasRequestingBusinessActivity exactly 1
RequestingBusinessActivity,
 hasRespondingBsinessActivity exactly 1
RespondingBusinessActivity

Class: BeginsWhen

Annotations:
 rdfs:comment "A description of an event external to this activity that normally causes it to commence (i.e. PreCondition + other variables = BeginsWhen)."

SubClassOf:
 BusinessRule

Class: BusinessActivity

SubClassOf:
 owl:Thing,
 hasPerforms exactly 2 Performs

DisjointWith:
 AttributeSubstitution,
 BusinessTransaction

Class: AND-Join

Annotations:
 rdfs:comment "Via the AND-Join (by default, the Join is an AND-Join), all transitions coming into the Join MUST be executed for the collaboration to reach the Join state that reflects the state movement.

waitForAll="True"

SubClassOf:
 Join

Class: Transition

Annotations:
rdfs:comment "A link between business states in a Business Collaboration. Choreography is expressed as transitions between business states. Transition to the same business state is allowed."

SubClassOf:
LinkingConstructs,
owl:Thing

DisjointWith:
BusinessTransaction

Class: BusinessDocument

Annotations:
rdfs:comment "A generic name of a document."

SubClassOf:
owl:Thing,
hasConditionExpression only ConditionExpression

DisjointWith:
AttributeSubstitution,
BusinessTransaction

Class: ResponseAcceptanceAcknowledgementFailure

SubClassOf:
AnyProtocolFailure

Class: Specification

Annotations:
rdfs:comment "A specification element that can associate many references to a particular ebBP element."

SubClassOf:
owl:Thing

DisjointWith:
AttributeSubstitution,
BusinessTransaction

Class: BusinessSuccess

SubClassOf:
Success

Class: ProtocolSuccess

SubClassOf:
Success

Class: RequestingRole

Annotations:

rdfs:comment "Allows definition of the Requesting declarative role on the Business Transaction. This explicit, yet abstract, role facilitates role mapping."

 SubClassOf:
 Role

 DisjointWith:
 BusinessTransaction

Class: BusinessTransactionActivity

 Annotations:

 rdfs:comment "A Business Transaction Activity is the performance of a Business Transaction within a collaboration. Business Transaction definitions can be associated to any number of BTA elements. This means that the same Business Transaction can be performed by multiple Business Transaction Activities in different collaborations, or by multiple Business Transaction Activities in the same collaboration, sometimes with opposite roles."

 SubClassOf:
 BusinessActivity,
 State,
 hasPerforms min 2 Performs

 DisjointWith:
 AttributeSubstitution,
 ProcessSpecification,
 BusinessTransaction,
 BusinessCollaboration,
 Package

Class: InformationDistribution

 Annotations:

 rdfs:comment "Represents an informal information exchange between parties."

 SubClassOf:
 BusienssTransactionPattern,
 hasRequestingBusinessActivity exactly 1
RequestingBusinessActivity,
 hasRespondingBsinessActivity exactly 1
RespondingBusinessActivity

Class: MessageMap

 SubClassOf:
 owl:Thing

 DisjointWith:
 BusinessTransaction

Class: SignalTimeout

 SubClassOf:
 AnyProtocolFailure

Class: RequestConfirm

Annotations:

 rdfs:comment "Used where an initiating party requests confirmation about their status with respect to previous obligations or a Responder's business rules."

SubClassOf:

 BusienssTransactionPattern

Class: ResponseRecieptAcknowledgementFailure

SubClassOf:

 AnyProtocolFailure

Class: OperationMapping

Annotations:

 rdfs:comment "An Operation Mapping specifies a possible mapping of a BTA to a set of web service operation invocations to enable the participation of a non-ebXML capable party in an ebXML relationship. An ebBP definition does not itself contain a reference to a WSDL file, but rather references to abstract operation names, which can be de-referenced with specific WSDL files, specified at the Collaboration Protocol Profile."

SubClassOf:

 owl:Thing,
 hasBusinessTransaction exactly 1 BusinessTransaction,
 hasRole exactly 1 Role

DisjointWith:

 BusinessTransaction

Class: RequestingBusinessActivity

Annotations:

 rdfs:comment "A Business Action performed by the Requesting role within a Business Transaction."

SubClassOf:

 BusinessAction

DisjointWith:

 AttributeSubstitution,
 BusinessTransaction

Class: GeneralException

SubClassOf:

 AnyProtocolFailure

Class: Fork

Annotations:

 rdfs:comment "A Fork MAY be defined without a corresponding Join. In this case, the TimeToPerform element MUST NOT be used. It MUST only be used in the case where all outgoing transitions from the Fork have incoming transitions to the Join."

As bounded by Fork semantics, multiple joins may be allowed for a fork (multiple dependencies exist)."

SubClassOf:
LinkingConstructs

DisjointWith:
BusinessTransaction

Class: EndsWhen

Annotations:
rdfs:comment "A description of an event external to this activity that normally causes it to conclude (i.e. PostCondition + other variables = EndsWhen)."

SubClassOf:
BusinessRule

Class: NotificationOfAcceptance

SubClassOf:
Notification

Class: BusinessAction

Annotations:
rdfs:comment "An abstract superclass that holds the attributes common to the Requesting and Responding Business Activity."

SubClassOf:
owl:Thing,
hasAcceptanceAcknowledgement exactly 1
AcceptanceAcknowledgement,
isAuthorizationRequired only boolean

DisjointWith:
BusinessTransaction

Class: TimeToPerform

Annotations:
rdfs:comment "The maximum amount of time between the time at which the request is sent and the substantive response is received."

SubClassOf:
owl:Thing,
hasVariable only Variable,
hasDuration some duration,
hasTimeToPerformType only NMToken

Class: BusinessTransactionPattern

Class: BusinessStateLinkType

Annotations:
rdfs:comment "The type related to the linking constructs (TO and FROM)."

```
SubClassOf:
  owl:Thing,
  hasConditionExpression only ConditionExpression
```

```
DisjointWith:
  BusinessTransaction
```

Class: AcceptanceAcknowledgementException

```
Annotations:
  rdfs:comment "An Acceptance Exception signals an error
condition in a Business Activity. This Business Signal is returned to
the initiating role that originated the request. This exception MUST
terminate the Business Transaction."
```

Typically, an Acceptance Exception means that the processing application (usually unknown to the other party) received the corresponding Business Document but was unable to process them."

```
SubClassOf:
  Signal

DisjointWith:
  BusinessTransaction
```

Class: MultiPartyCollaboration

```
Annotations:
  rdfs:comment "Multiparty (Business) collaborations involve more
than two abstract partner roles."
```

```
SubClassOf:
  BusinessCollaboration

DisjointWith:
  BusinessTransaction
```

Class: BusinessRule

Class: BusinessPartnerRole

```
Annotations:
  rdfs:comment "Each business partner plays one or more abstract
partner roles in the Business Collaboration."
```

```
SubClassOf:
  owl:Thing

DisjointWith:
  AttributeSubstitution,
  BusinessTransaction
```

Class: CompletionState

```
EquivalentTo:
  Failure
  or Success
```

```
SubClassOf:
```

State

Class: RequestRecieptAcknowledgementFailure

SubClassOf:
AnyProtocolFailure

Class: AcceptanceAcknowledgement

Annotations:
rdfs:comment "The Acceptance Acknowledgement Business Signal, if used, signals that the message received (Request or Response) has been accepted for business processing and that processing is complete and successful by the receiving application, service or a receiving business application proxy."

SubClassOf:
Signal

DisjointWith:
BusinessTransaction

Class: State

SubClassOf:
owl:Thing

Class: SimpleVariable

Annotations:
rdfs:comment "Simple variable reference a BTA and a Business Document exchanged as part of this BTA."

SubClassOf:
Variable

Class: Decision

Annotations:
rdfs:comment "A Decision selects only one of the possible transitions, and the other(s) is/are automatically disabled. An XOR Fork may be designed to operate like a Decision, but a Decision cannot be an XOR Fork."

SubClassOf:
LinkingConstructs

DisjointWith:
BusinessTransaction

Class: BinaryCollaboration

Annotations:
rdfs:comment "A Binary (Business) Collaboration involves two top-level or abstract partner roles only."

SubClassOf:
BusinessCollaboration

DisjointWith:

BusinessTransaction

Class: OptName

SubClassOf:
owl:Thing

DisjointWith:
BusinessTransaction

Class: OR

Annotations:
rdfs:comment "An OR value mean that one or more Business Activity pointed to by a transition coming from the Fork might be initiated. Several paths are possible although when and which become active is unknown. These Business Activities MAY occur in parallel."

SubClassOf:
Fork

Class: ReceiptAcknowledgementException

Annotations:
rdfs:comment "A Receipt Exception signals an error condition in the management of a Business Transaction. This Business Signal is returned to the initiating activity that originated the request. This exception MUST terminate the Business Transaction."

SubClassOf:
Signal

DisjointWith:
BusinessTransaction

Class: ReceiptAcknowledgement

Annotations:
rdfs:comment "The Receipt Acknowledgement Business Signal, if used, signals that a message (Request or Response) has been properly received by the BSI software component."

SubClassOf:
Signal

DisjointWith:
BusinessTransaction

Class: ComplexBusinessTransactionActivity

Annotations:
rdfs:comment "A Complex Business Transaction Activity (ComplexBTA) allows for nested BTAs to happen in a recursive manner."

SubClassOf:
BusinessActivity,
State,
hasBusinessActivity min 1 BusinessActivity

DisjointWith:
BusinessTransaction

Class: CollaborationActivity

Annotations:
rdfs:comment "A Collaboration Activity is the performance of a Business Collaboration, within another Business Collaboration."

SubClassOf:
BusinessActivity,
State

DisjointWith:
AttributeSubstitution,
BusinessTransaction

Class: Documentation

Annotations:
rdfs:comment "Defines user documentation for any element."

SubClassOf:
owl:Thing

DisjointWith:
AttributeSubstitution,
BusinessTransaction

Class: PostCondition

Annotations:
rdfs:comment "A description of a state external to this activity that is required after the activity concludes (i.e. the state doesn't exist before the execution of this activity but does exist afterwards)."

SubClassOf:
BusinessRule

Class: Signal

Annotations:
rdfs:comment "As a Business Action, this element defines the identification structure for Business Signal messages to be sent to a trading partner."

SubClassOf:
owl:Thing

DisjointWith:
BusinessTransaction

Class: LinkingConstructs

Class: owl:Thing

Class: FromLink

Annotations:
 rdfs:comment "A linking construct that indicates a state that can be transitioned from in the current context (containing element)."

SubClassOf:
 BusinessStateLinkType,
 hasConditionExpression only ConditionExpression

DisjointWith:
 BusinessTransaction

Class: RequestAcceptanceAcknowledgementFailure

SubClassOf:
 AnyProtocolFailure

Class: Variable

Annotations:
 rdfs:comment "Variables are named information elements that are available to bind concepts across Business Transaction. They also serve to make the semantics clear in a condition expression."

SubClassOf:
 owl:Thing

DisjointWith:
 BusinessTransaction

Class: AttributeSubstitution

Annotations:
 rdfs:comment "Attribute or document value should be used in place of some value in an existing Process Specification. Attribute substitution could be used for document substitution. These substitution changes were made in v2.0. AttributeSubstitution allowed to "edit" references (IDREFS) or other attribute values."

SubClassOf:
 owl:Thing

DisjointWith:
 BusinessTransaction,
 SignalMap,
 BusinessCollaboration,
 Documentation,
 CollaborationActivity,
 RequestingBusinessActivity,
 BusinessActivity,
 BusinessDocument,
 RespondingBusinessActivity,
 Start,
 BusinessPartnerRole,
 Specification,
 BusinessTransactionActivity

Class: NotificationOfFailure

SubClassOf:
Notification

Class: PreCondition

Annotations:
rdfs:comment "A description of a state external to this activity that is required before the activity can commence."

SubClassOf:
BusinessRule

Class: ProcessSpecification

Annotations:
rdfs:comment "Root element of a Process Specification document that has a globally unique identity. The Process Specification element can specify the version of the technical specification used and the process instance version related to the target ebBP (schema)."

SubClassOf:
owl:Thing

DisjointWith:
BusinessTransaction,
BusinessTransactionActivity

Class: RespondingRole

Annotations:
rdfs:comment "Allows definition of the the Responding declarative role on the Business Transaction. This explicit, yet abstract, role facilitates role mapping."

SubClassOf:
Role

DisjointWith:
BusinessTransaction

Class: Start

Annotations:
rdfs:comment "The specific Collaboration started with to traverse a path through a graph to a Completion State."

SubClassOf:
LinkingConstructs

DisjointWith:
AttributeSubstitution,
BusinessTransaction

Class: RespondingBusinessActivity

Annotations:
rdfs:comment "A Business Action performed by the Responding role within a Business Transaction."

SubClassOf:
BusinessAction

DisjointWith:
AttributeSubstitution,
BusinessTransaction

Individual: OrderAcceptedInFull_DE

Types:
DocumentEnvelope,
owl:Thing

Facts:
hasBusinessDocument OrderAcceptedInFull_BD,
isPositiveResponse "true"^^xsd:boolean,
isTamperDetectable "transient",
hasName "Accepted In Full",
isAuthenticated "transient",
isConfidential "transient",
hasNameID "OrderAcceptedInFull_DE"

Individual: rabpss2

Types:
Specification,
owl:Thing

Facts:
hasLocation "ebbp-signals-2.0.xsd"^^xsd:anyURI,
hasName "ReceiptAcknowledgement",
hasNameID "raebpss2"

Individual: AA2_B

Types:
AcceptanceAcknowledgement,
owl:Thing

Facts:
hasSignal aa2,
hasName "AA2",
hasNameID "AA2_B"

Individual: aebpss2

Types:
Specification,
owl:Thing

Facts:
hasLocation "ebbp-signals-2.0.xsd"^^xsd:anyURI,
hasName "Exception",
hasNameID "aebpss2"

Individual: RA2

Types:
ReceiptAcknowledgement,
owl:Thing

Facts:
hasSignal ra2,
hasName "RA2",
hasNameID "RA2"

Individual: OrderDenied_DE

Types:
DocumentEnvelope,
owl:Thing

Facts:
hasBusinessDocument OrderDenied_BD,
isPositiveResponse "false"^^xsd:boolean,
isTamperDetectable "transient",
isAuthenticated "transient",
hasName "Denied",
hasNameID "OrderDenied_DE",
isConfidential "transient"

Individual: RA2_B

Types:
ReceiptAcknowledgement,
owl:Thing

Facts:
hasSignal ra2,
hasName "RA2",
hasNameID "RA2_B"

Individual: OrderInitiator

Types:
RequestingRole,
owl:Thing

Facts:
hasName "InitiatorOrderWithSimpleOrComplexResponse",
hasNameID "OrderInitiator"

Individual: raebpss2

Types:
Specification,
owl:Thing

Facts:
hasLocation "ebbp-signals-2.0.xsd"^^xsd:anyURI,
hasName "Exception",
hasNameID "raebpss2"

Individual: ConditionExpression3

Types:
ConditionExpression,
owl:Thing

Facts:
hasExpressionLanguage "DocumentEnvelope",
hasExpression "OrderDenied_DE"

Individual: Performs2

Types:
Performs,
owl:Thing

Facts:
hasCurrentRole Seller,
hasPerformsRole OrderResponder

Individual: aa2

Types:
AcceptanceAcknowledgement,
owl:Thing

Facts:
hasSpecification aabpss2,
hasName "AcceptanceAcknowledgement",
hasNameID "aa2"

Individual: aae2

Types:
AcceptanceAcknowledgementException,
owl:Thing

Facts:
hasSpecification aae2bss2,
hasName "AcceptanceAcknowledgementException",
hasNameID "aae2"

Individual: AAE2_B

Types:
AcceptanceAcknowledgementException,
owl:Thing

Facts:
hasSignal aae2,
hasName "AAE2",
hasNameID "AAE2_B"

Individual: OrderResponseSimpleOrderDenied_SP

Types:
Specification,
owl:Thing

```
Facts:
  hasLocation "http://docs.oasis-open.org/ubl/cd-UBL-
2.0/xsdrt/maindoc/UBL-OrderResponseSimple-2.0.xsd"^^xsd:anyURI,
  hasTargetNamespace
"urn:oasis:names:specification:ubl:schema:xsd:OrderResponseSimple-
2"^^xsd:anyURI,
  hasName "Order Response Simple - Order Denied",
  hasType "schema",
  hasNameID "OrderResponseSimpleOrderDenied_SP"
```

Individual: ra2

```
Types:
  ReceiptAcknowledgement,
  owl:Thing
```

```
Facts:
  hasSpecification rabpss2,
  hasName "ReceiptAcknowledgement",
  hasNameID "ra2"
```

Individual: Ordering

```
Types:
  Package,
  owl:Thing
```

```
Facts:
  hasBusinessTransaction CreateOrder_CT,
  hasBusinessCollaboration CreateOrder_BC,
  hasBusinessDocument OrderAcceptedInFull_BD,
  hasBusinessDocument OrderDenied_BD,
  hasBusinessDocument Order_BD,
  hasName "Ordering",
  hasNameID "Ordering"
```

Individual: ConditionExpression4

```
Types:
  ConditionExpression,
  owl:Thing
```

```
Facts:
  hasExpressionLanguage "DocumentEnvelope",
  hasExpression "OrderAcceptedInFull_DE"
```

Individual: CreateOrder_BTA

```
Types:
  BusinessTransactionActivity,
  owl:Thing
```

```
Facts:
  hasPerforms Performs1,
  hasPerforms Performs2,
  refersToBusinessTransaction CreateOrder_CT,
  hasLegalIntent "true"^^xsd:boolean,
  hasName "Create Order",
  hasNameID "CreateOrder_BTA"
```

Individual: OrderDenied_BD

Types:

BusinessDocument,
owl:Thing

Facts:

hasSpecification OrderResponseSimpleOrderDenied_SP,
hasConditionExpression ConditionExpression1,
hasName "Order Denied",
hasNameID "OrderDenied_BD"

Individual: ConditionExpression1

Types:

ConditionExpression,
owl:Thing

Facts:

hasExpressionLanguage "XPath1",
hasExpression "//AcceptedIndicator='false'"

Individual: OrderWithSimpleResponseProcess_PS

Types:

ProcessSpecification,
owl:Thing

Facts:

hasBusinessTransaction CreateOrder_CT,
hasBusinessDocument OrderAcceptedInFull_BD,
hasBusinessDocument OrderDenied_BD,
hasBusinessDocument Order_BD,
hasUuid "bpid:urn:oasis:names:draft:bpss:ubl-2-order-with-simple-response-process-2",
hasInstanceVersion "0.9.7",
hasName "Order With Simple Response Process",
hasNameID "OrderWithSimpleResponseProcess_PS",
hasSpecificationVersion "2"

Individual: CreateOrder_CT

Types:

CommercialTransaction,
owl:Thing

Facts:

hasRequestingRole OrderInitiator,
hasRequestingRole OrderResponder,
hasRespondingBusinessActivity FirmOrder_ResBA,
hasRequestingBusinessActivity SendOrder_ReqBA,
hasName "Create Order",
isGuaranteedDeliveryRequired "true"^^xsd:boolean,
hasNameID "CreateOrder_CT"

Individual: ge2

Types:

Signal,

owl:Thing

Facts:
hasSpecification gebpss2,
hasName "GeneralException",
hasNameID "ge2"

Individual: RAE2

Types:
ReceiptAcknowledgementException,
owl:Thing

Facts:
hasSignal rae2,
hasName "RAE2",
hasNameID "RAE2"

Individual: aabpss2

Types:
Specification,
owl:Thing

Facts:
hasLocation "ebbp-signals-2.0.xsd"^^xsd:anyURI,
hasName "AcceptanceAcknowledgement",
hasNameID "aabpss2"

Individual: OrderAcceptedInFull_BD

Types:
BusinessDocument,
owl:Thing

Facts:
hasSpecification OrderResponseSimpleOrderAcceptance_SP,
hasConditionExpression ConditionExpression2,
hasName "Order Accepted In Full",
hasNameID "OrderAcceptedInFull_BD"

Individual: Order_BD

Types:
BusinessDocument,
owl:Thing

Facts:
hasSpecification Order_SP,
hasName "Order",
hasNameID "Order_BD"

Individual: rae2

Types:
ReceiptAcknowledgementException,
owl:Thing

Facts:

```

hasSpecification raebpss2,
hasName "ReceiptAcknowledgementException",
hasNameID "rae2"

```

Individual: Buyer

```

Types:
  Role,
  owl:Thing

```

```

Facts:
  hasName "Buyer",
  hasNameID "Buyer"

```

Individual: CreateOrder_Decision

```

Types:
  Decision,
  owl:Thing

```

```

Facts:
  hasFromBusinessState CreateOrder_BTA,
  hasToBusinessState CreateOrder_Failure,
  hasToBusinessState CreateOrder_Success,
  hasName "Create Order",
  hasNameID "CreateOrder_Decision"

```

Individual: OrderResponder

```

Types:
  RespondingRole,
  owl:Thing

```

```

Facts:
  hasName "ResponderOrderWithSimpleOrComplexResponse",
  hasNameID "OrderResponder"

```

Individual: OrderResponseSimpleOrderAcceptance_SP

```

Types:
  Specification,
  owl:Thing

```

```

Facts:
  hasLocation "http://docs.oasis-open.org/ubl/cd-UBL-
2.0/xsdrt/maindoc/UBL-OrderResponseSimple-2.0.xsd"^^xsd:anyURI,
  hasTargetNamespace
"urn:oasis:names:specification:ubl:schema:xsd:OrderResponseSimple-
2"^^xsd:anyURI,
  hasName "Order Response Simple - Order Acceptance",
  hasType "schema",
  hasNameID "OrderResponseSimpleOrderAcceptance_SP"

```

Individual: Order_SP

```

Types:
  Specification,
  owl:Thing

```



```
Facts:
  hasLocation "http://docs.oasis-open.org/ubl/cd-UBL-
2.0/xsdr/maindoc/UBL-Order-2.0.xsd"^^xsd:anyURI,
  hasTargetNamespace
"urn:oasis:names:specification:ubl:schema:xsd:Order-2"^^xsd:anyURI,
  hasName "Order",
  hasType "schema",
  hasNameID "Order_SP"
```

Individual: gebpss2

```
Types:
  Specification,
  owl:Thing
```

```
Facts:
  hasLocation "ebbp-signals-2.0.xsd"^^xsd:anyURI,
  hasName "Exception",
  hasNameID "gebpss2"
```

Individual: RAE2_B

```
Types:
  ReceiptAcknowledgementException,
  owl:Thing
```

```
Facts:
  hasSignal rae2,
  hasName "RAE2",
  hasNameID "RAE2_B"
```

Individual: CreateOrder_Failure

```
Types:
  Failure,
  owl:Thing
```

```
Facts:
  hasConditionExpression ConditionExpression3,
  hasName "Failure",
  hasNameID "CreateOrder_Failure"
```

Individual: Order_DE

```
Types:
  DocumentEnvelope,
  owl:Thing
```

```
Facts:
  hasBusinessDocument Order_BD,
  isTamperDetectable "transient",
  isAuthenticated "transient",
  hasName "UBL 2 Order",
  isConfidential "transient",
  hasNameID "Order_DE"
```

Individual: Seller

Types:
Role,
owl:Thing

Facts:
hasName "Seller",
hasNameID "Seller"

Individual: AA2

Types:
AcceptanceAcknowledgement,
owl:Thing

Facts:
hasSignal aa2,
hasName "AA2",
hasNameID "AA2"

Individual: SendOrder_ReqBA

Types:
RequestingBusinessActivity,
owl:Thing

Facts:
hasAcceptanceAcknowledgement AA2,
hasDocumentEnvelope Order_DE,
hasReceiptAcknowledgementException RAE2,
hasReceiptAcknowledgement RA2,
hasAcceptanceAcknowledgementException AAE2,
isIntelligibleCheckRequired "true"^^xsd:boolean,
hasName "Send Order",
isAuthorizationRequired "true"^^xsd:boolean,
isNonRepudiationReceiptRequired "true"^^xsd:boolean,
hasNameID "SendOrder_ReqBA",
isNonRepudiationRequired "true"^^xsd:boolean

Individual: CreateOrder_BC

Types:
BusinessCollaboration,
owl:Thing

Facts:
hasBusinessTransactionActivity CreateOrder_BTA,
hasRole Buyer,
hasRole Seller,
hasState CreateOrder_Failure,
hasState CreateOrder_Success,
hasStart CreateOrder_ST,
isRealisationOfProcessSpecification
OrderWithSimpleResponseProcess_PS,
hasDecision CreateOrder_Decision,
hasName "Create Order",
hasNameID "CreateOrder_BC"

Individual: ConditionExpression2

Types:
ConditionExpression,
owl:Thing

Facts:
hasExpressionLanguage "XPath1",
hasExpression "//AcceptedIndicator='true'"

Individual: CreateOrder_ST

Types:
Start,
owl:Thing

Facts:
hasName "Start Send Order",
hasNameID "CreateOrder_ST"

Individual: AAE2

Types:
AcceptanceAcknowledgementException,
owl:Thing

Facts:
hasSignal aae2,
hasName "AAE2",
hasNameID "AAE2"

Individual: FirmOrder_ResBA

Types:
RespondingBusinessActivity,
owl:Thing

Facts:
hasAcceptanceAcknowledgement AA2_B,
hasDocumentEnvelope OrderAcceptedInFull_DE,
hasDocumentEnvelope OrderDenied_DE,
hasReceiptAcknowledgementException RAE2_B,
hasReceiptAcknowledgement RA2_B,
hasAcceptanceAcknowledgementException AAE2_B,
isIntelligibleCheckRequired "true"^^xsd:boolean,
hasName "Firm Order",
isAuthorizationRequired "true"^^xsd:boolean,
isNonRepudiationReceiptRequired "true"^^xsd:boolean,
hasNameID "FirmOrder_ResBA",
isNonRepudiationRequired "true"^^xsd:boolean

Individual: CreateOrder_Success

Types:
Success,
owl:Thing

Facts:
hasConditionExpression ConditionExpression4,
hasName "Success",

hasNameID "CreateOrder_Success"

Individual: Performs1

Types:

Performs,
owl:Thing

Facts:

hasCurrentRole Buyer,
hasPerformsRole OrderInitiator