INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI[®]

Carleton University

**Evaluating Data Interoperability:
A Policy-Driven Ontological Analysis Model**

A dissertation submitted in partial fulfillment of the
requirement for the degree of

Doctor of Philosophy

By

Robert John Garigue

B.A.A Méthodes Quantitatives, Ecole Des Hautes Etudes Commerciales (HEC) Montréal 1976

M.Sc. Computer and Information Systems. Claremont Graduate University California 1992

4 April 2005

©

Library and
Archives Canada

Bibliothèque et
Archives Canada

0-494-06735-7

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Acknowledgements

*Ne demande pas de miracle, mais fait l'effort
de comprendre et de faire ce qui est nécessaire ;
garde toi ouvert au nouveau,
car si tous les humains portent des chaînes
c'est qu'ils les ont largement construites par eux-mêmes,
et peuvent sortir de leurs souffrances
s'ils acceptent de choisir ce qui est vrai.*

Le quotidien de l'existence

Philippe Garigue – 2004

Many people have provided me with inspiration and given me support over the course of this research. Some of them fired my thoughts, other offered clues, and some made sure I watched out for the traps. Therefore, it was fortunate that I had three supervisors. Through the discussions, challenges, and sound advice of Professor Mike Hine and Professor Alex Ramirez, I was able to operate in a more systematic and disciplined manner. I will forever be grateful to Professor Wojtek Michalowski for his continuous end to end focus, logical and critical appreciation of the work. His constant support and encouragement were invaluable. A special thanks to Dr. Natasha Noy and Dr. Bob Smith for all their advice and comments.

There are also numerous friends and colleagues who have helped me in many ways as I struggled along. I would like to acknowledge Dr. Leroy Pearce, Omar Petralis, Randy Alward, Kalev Ruberg, Carolee Birchall, Karen Metakos, Barry Gilmore, Lloyd Darlington, Jim Robbins, Bain McKay and Wayne Ronhaar for their support and encouragement as well as my long term PhD traveling companions, Duncan Bailey and Doug Orendorff, for their constant optimism and joyful dispositions.

To my mother and father, as well as my brother and sister who have forever supported my work and objectives – Merci pour tout. Finally, to my wife Carol and daughters Alexandra and Francesca – thank you; I could never have done this without your endless patience and boundless love.

Table of Content

Acknowledgements.....	2
Table of Content	3
List of Tables	5
List of Figures	5
 ABSTRACT.....	 7
 CHAPTER 1	 8
INTRODUCTION	8
Introduction	8
Background	9
Context of the research.....	10
Focus of the Research Questions	10
Proposed Model – Policy-Driven Ontological Analysis Model (PDOAM)	14
Contribution to the Body of Knowledge	16
Descriptions of the Research by Chapters.....	16
 CHAPTER 2	 19
CONCEPTUAL FOUNDATIONS.....	19
The Data interoperability Challenge	19
Concepts and Ontologies as Basis for Interoperability	22
Literature Review	23
Defining Ontologies	24
Evaluating Ontologies	27
Concept Interoperability	29
Interoperability as Concept Coherence.....	32
From Concept Interoperability to System Interoperability.....	34
Ontologies as Standards of Core Business Concepts	35
The Use of Ontologies in Data Interoperability	37
 CHAPTER 3	 41
POLICY-DRIVEN ONTOLOGICAL ANALYSIS MODEL	41
Justification for a Policy-Driven Ontological Analysis Model	42
The Motivation for Policy-Driven Ontologies	43
The Efficacy of Policy-Driven Ontologies.....	44
Ontology Heterogeneity and Semantic Integration across Multiple Systems	45
Ontologies Integration and Description Logic	47
Ontological Engineering Communities of Practices	50
 CHAPTER 4	 52
RESEARCH METHODOLOGY.....	52

The Dynamics of a Policy-Driven Ontology.....	52
Comparing Domain Ontologies and Policy-Driven Ontologies.....	54
Proposed Policy-Driven Ontological Analysis Model	55
Value of the Model.....	55
Analytical Approach: Process and Structure.....	56
Considerations for Dataset Level Ontology Modeling and Creation	58
Policy-Driven ontology modeling and creation.....	59
Comparing Domain ontologies and Policy-Driven Ontologies.....	60
Constructing, Verifying and Validating the Model	61
Conditions for Asserting Interoperability.....	61
Using Apriori Knowledge to Empirically Validate the Model	63
Testing for Model Coherence through a theoretical problem.....	63
Testing for Model Correspondence Using a Real World Problem.....	63
Ontological Engineering Environments.....	64
Ontology Analysis Environment: Protégé.....	65
Comparing Ontologies.....	66
Comparing Ontologies within Protégé: Using Prompt.....	67
Comparing Two Ontologies using Prompt.....	70
Setting Up the PDOAM Model.....	71
Ontology Mismatches: Generic Issues.....	75
Semantic Matches and Mismatches: Determining Alignment	77
Determining Interoperability.....	82
From Ontology Alignment to Ontology Interoperability	83
CHAPTER 5	86
TRIALS AND ASSESSMENT	86
Empirical Comparative Analysis Framework	86
Testing the Policy-Driven Ontological Analysis Model	90
Expressing the PDOAM with a Theoretical Example.....	90
Description of Model Concepts and Test Challenges	90
Constructing an Ontology of the Concept of Employee.....	91
Constructing an Ontology of the Concept of Customer	94
Ontology Alignment Analysis with a PDO	98
Comparing a Domain Ontology against a Policy-Driven Ontology.....	99
From Alignment Analysis to Interoperability Analysis.....	101
Term Analysis.....	101
Class Analysis.....	102
Testing the Model with Theoretical Data.....	102
Testing the PDOAM with Organizational Data	105
Sources of Organizational Policy-Driven Ontologies	106
Constructing the Policy-Driven Ontology of the Concept of Customer	108
Constructing Domain Ontologies using Organizational Data Sets	111
Validating the Model using the apriori Knowledge of Outcomes.....	113
Application A - Domain Ontology - Involved Party	113
Alignment Analysis of the Involved Party Domain Ontology.....	115
Application B – Domain Ontology – Individual	116

Alignment Analysis of the Individual Domain Ontology.....	118
Application C – Domain Ontology – Party	119
Alignment Analysis of the Party Domain Ontology.....	120
Assessment of Potential Interoperability between Application A B and C.....	120
CHAPTER 6	122
ANALYSIS AND DISCUSSION.....	122
Analysis of the Results.....	122
Discussions and Observations	124
Significance and Contributions of the Model	125
Limitations of the Model.....	128
Future Research: Alignment and Interoperability Metrics.....	132
CHAPTER 7	135
CONCLUSIONS.....	135
Appendix A – Employee Ontology.....	138
Appendix B – Customer Ontology	139
Appendix C – Bank Branch Ontology	141
Appendix D : Financial Products and Services Ontology	143
Appendix E: Enterprise PDO Ontology for Customer	146
Appendix F: Application A Involved_Party	161
Appendix G: Application B - Individual	171
Appendix H: Application C - Party	179
Bibliography	186

List of Tables

Table 1- Ontological Alignment Assessment Framework.....	77
Table 2- Data Interoperability as Outcomes of Ontology Alignment Analysis.....	82
Table 3- Interoperability decision criteria between Two Domain Ontologies	85
Table 4- Interoperability analysis of two Domain Ontologies.....	105
Table 5-Apriori Interoperability outcomes	122
Table 6- PDOAM Outcomes	123
Table 7- Ontology Alignment Metrics.....	134

List of Figures

Figure 1- Formal Ontologies.....	12
Figure 2- PDOAM Generalized Model.....	15
Figure 3- Generalized ontological analysis framework	57
Figure 4- Visualization of a portion of a Policy-Driven Customer Ontology in Protégé. 69	
Figure 5 – Example of Comparing Policy-Driven Ontology and a Domain Ontology	69

Figure 6- Legend.....	70
Figure 7 - Policy defined definition of the concept of customer	73
Figure 8 - Equivalent Ontologies.....	78
Figure - 9 Equivalent and synonymous	78
Figure - 10 – Subset.....	79
Figure 11 - Synonymous Subset.....	79
Figure 12 - Overlap.....	80
Figure 13 - Synonymous overlap.....	80
Figure 14 - Homonymy.....	81
Figure 15 - Disjointed.....	81
Figure 16 – Phases in Comparing two Domain Ontologies.....	83
Figure 17- Policy-Driven Ontological Analysis Model.....	86
Figure 18- Employee ERD.....	92
Figure 19- Employee Ontology	94
Figure 20- ERD of Customer.....	95
Figure 21- Customer Ontology	95
Figure 22 - Comparative analysis of the two ontologies	96
Figure 23- Comparison of the concept of employee and the concept of customer.....	97
Figure 24- Policy-Driven Ontology Comparison	99
Figure 25 - Domain Ontologies- Bank and Products and Services	103
Figure 26- Alignment of two domain ontologies against the PDO of Customer.....	104
Figure 27- Holistic view of the customer	110
Figure 28 – Main Entities of the Policy-Driven Ontology for the Concept of Customer.....	111
Figure 29- Ontology of Involved Party from Application A – Involved Party	115
Figure 30 - Alignment of DO for application A with the PDO	116
Figure 31 - Ontology of Application B – Individual	117
Figure 32- Alignment of DO -B with PDO	118
Figure 33 - Ontology of Party from Application C -Party.....	119
Figure 34 - Alignment DO C with the PDO	120

ABSTRACT

This dissertation will describe the problem of semantic data interoperability and propose a model to evaluate the Policy-Driven Ontological Analysis Model (PDOAM) (Garigue 2004). This model is used to determine the potential for semantic data interoperability between different applications. Interoperability is the ability to exchange and use information and makes possible the integration of various datasets and applications into more complex and intricate enterprise-wide applications. Interoperability requires that the original concepts represented in each of the various databases are equivalent. The PDOAM uses ontological analysis as a way to assess and identify potential interoperability issues between datasets that come from differences or inconsistencies in underlying concepts. Without conceptual equivalence across multiple datasets, the integrity of the datasets is uncertain and the result is invalid assertion statements. Concepts that underlie datasets can be represented through ontologies. This research investigates the use of Ontological Analysis as a way to identify conceptual mismatches that would otherwise prevent interoperability. As part of the model, the notion of a Policy-Driven Ontology is introduced in order to establish the legitimacy of an enterprise-wide defined concept. The Policy-Driven Ontology acts as a reference model and is used to compare domain level ontologies against it. The analysis and valuation of the model will be done through a phased review, in two main evaluations. The first phase structures the analytical framework and describes the processes required to construct Domain Ontologies and Policy-Driven Ontologies. Criteria for assessing semantic alignment between two ontologies are also created. The second phase demonstrates the model. Both theoretical and organizational examples of ontologies will be used in support of this review. Observational data required for these constructions will come from extracts from information systems and enterprise policies from a large financial organization. Validating the model is done in both examples using a priori knowledge of outcomes. By reviewing cases where the interoperability outcome is known a priori, the research can justify empirically that the proposed model is valid within the scope of the research. The findings are presented in the evaluation of outcomes.

CHAPTER 1

INTRODUCTION

Introduction

This research investigated the use of ontologies as a way to determine whether semantic interoperability exists between two sets of data. These type of problems are now common across all organizations (McComb 2004; Pollock et al. 2004). There is, however no standard way to determine whether or not applications and their related datasets are based on similar concepts. This puts the integrity of information sets at risk. The danger lies in the assumption that because some of the terms and properties are similar (or the same) within two data sets that both terms and properties represent the exact same concept; conversely, dissimilar terms and properties may represent the same concepts. This risk exists as a consequence of any integration initiative in an organization. All in all, the problem of data interoperability comes down to determining semantic similarity in data and the determination of differences across concepts as they are used by applications.

In order to resolve this problem, this research will review and investigate a set of methodologies and technologies that center on the use of ontologies in organizations. This research will help review and answer a set of questions relating to the contribution of ontological engineering and ontological analysis in support of resolving the data interoperability problem. The research will show that there is a method that is able to examine and analyze data at a semantic level and is also able to evaluate conceptual similarities and differences. By the same token, it will show the processes required to support the extraction, abstraction and evaluation of data at a semantic level. In addition, this research will show how these methods can be performed practically in an organization. In doing so, it will show the challenges and the benefits of ontological analysis and ontological engineering in an organization. To do this the research will develop and validate an ontological analysis model that can be used to investigate concept alignment in support of data interoperability.

Background

All organizations are faced with a critical strategic challenge – how to effectively and efficiently manage the increasing amount of data and information resident in their various repositories (Lesser 2000). This challenge comes as a result of a combination of factors, but overall the lower cost of technology has been one of the main drivers of organizational dysfunction (Boisot 1998; Borghoff et al. 1998b; Botkin 1999; Tiwana 2001) by localizing datasets across the enterprise. Short-sighted reasoning asserts that (in the short term) it costs less to add storage than it does to put in place effective content governance and management (Stewart 2001). The result is that each organization ends up with a multiplicity of formats, standards and applications that lock data into unique views which are not accessible to each other. From email to databases, each set of application structures content into domains that support end users in their need for sense and decision making (Lloyd et al. 1998; Weick 1995; Weick 2001). The present architecture of information systems within organizations is not conducive to enterprise integration and this means that there is significant re-creation of the same datasets (Pearlson 2001). Despite this, every expectation from organizational governance is of improved performance that comes directly from the integration of data, information and knowledge. A brief illustration of the critical impact of poor concepts interoperability is be seen by looking briefly at Pollock (Pollock et al. 2004) as he describes the problems of semantic integration across technology initiatives such as Web Services, Work Flows, and corporate data repositories.

System Integration or Enterprise Application Integration is consuming an increasing proportion of enterprise resources and development budgets. McComb (McComb 2004) states that 20 to 60% of the annual budget of System Integration is devoted to building and maintaining integration. Most important, however is the estimation that 95% of the cost of System Integration is attributable to resolving semantic interoperability challenges.

Context of the research

Organizations are trying to leverage their investments in systems and information through interoperability (Obrst 2003; O'Sullivan et al. 2003). Interoperability is defined as the ability of two or more systems or components to exchange data and information and the ability to use the data and information that has been exchanged in a meaningful way (O'Sullivan et al. 2003; Renner et al. 1996; Renner 2001; Visser et al. 1997). The reason for this ability is that, notwithstanding the differences in technical protocols, there is an overlaying shared meaning or conceptual frame of reference that comes into play between the systems that ensures interoperability. In the end there is an exchange of information in a useful and meaningful manner enabled by a shared common representation. One way to assess interoperability opportunities is to review the underlying conceptual representations that were used to structure the data that was used by the various applications. Interoperability must also deal with another fundamental problem: in some situations the same business concepts are stored in multiple databases and though they might vary in their representation, class hierarchies and syntax, they may in fact refer to the same real world concept or entity.

The "context of research" is driven by a need for methodological innovation that resolves the data interoperability challenge. The research is justified by proliferation of organizational data sources and a need to operationally utilize the data. This research will have a practical implication but this is not the object of the research contribution. The interoperability of data sources as approached from the theoretical perspective (how to draw conclusions on a certain concept using different sources of data) requires a methodological solution embedded in a model. In this context, the proposed model needs to be both justified and validated. At the present time some practical limitations are discussed but do not invalidate the model.

Focus of the Research Questions

The current state of knowledge and practices identifies gaps in the present approach to ontological engineering models. This leads up to the examination and exploration of how

an organization can maintain its conceptual integrity across datasets in support of an expectation of dataset interoperability. This research proposes a model that relies on the analysis ontologies as the best potential for addressing the interoperability problem. To this effect this research will answer the following questions:

“Can data interoperability be determined through the use of a policy-driven ontological analysis model?”

From a methodological point of view, this research will also have to produce supporting criteria in the use of a new model therefore the following question has also be answered:

“What evaluation criteria are required to support the policy-driven ontological analytical model?”

The proposed Policy-Driven Ontological Analysis model (PDOAM) is based on an ontological analysis framework that will use domain ontologies and an enterprise-wide Policy-Driven Ontology. This approach is novel. The model also addresses the methodology, the supporting evaluation criteria as well as the ontological analysis process itself. This proposal will put forward both the model process and methodology used to assess interoperability.

Ontological engineering practices create higher-level representational frameworks that offer new levels of analytical capability, but questions still remain as to which approach offer the best options for addressing interoperability issues. Ontological engineering practices have focused mainly on the process to create ontologies, and at present there are no such definitive guiding principles for using ontologies to address interoperability between dataset structures.

In addressing the research question, the investigation will also be able to lead the way in answering questions such as: “how do ontologies enable the sharing of information across applications” and “how ontologies can help determine where interoperability efforts will succeed, and where they will fail?”

Ontologies vary in their ability to express semantics due to the level of formalism used to construct them. The more formal the representation, the more opportunity to infer new knowledge and express meaning (Bruijn 2003; Gomez-Perez et al. 2004). The degree of formality ranges as indicated in Fig -1 from simple taxonomies, to thesaurus to conceptual models, to truly logical domain theories. However, only formal ontologies are of interest to this research.

Uischold, Grueninger - Sigmod 33 no 4

Figure 1- Formal Ontologies

One approach to dataset analysis is through metadata review. Metadata is data about data and is also used as a way to represent concepts. Metadata that comes from modeling processes such as Unified Modeling Language (UML) and Entity Relation Diagrams

(ERD) is also very close to being considered equivalent to formal ontologies (Shanks et al. 2003). Concepts modeled in these representational languages can be transformed into full ontological representations.

Although not yet as widespread, one reason for organizations to use ontologies is to enable content processing at higher levels of abstractions. They can be used to model the organization without having to consider implementation constraints (Fisher et al. 2003). This approach can be useful to validate business models before they get implemented through technology investment programs. Ontologies try to capture the intrinsic conceptual structure of a specific domain but because ontological management is still a nascent subject within the context of information systems research, there is very little empirical evidence that demonstrates which ontology management practices are more efficient than others. In this regards this research is also contributing to the development of best practices in the area of ontology management.

As indicated, the primary objective of the research is to answer if data interoperability can be determined through the use of a policy-driven ontological analysis model as well as developing and proposing evaluation criteria that operationalize the Policy-Driven Ontological Analysis Model (PDOAM) (Garigue 2004).

The analysis and valuation of the model is being done through a phased review; of which there is be two main phases. The first phase declares the processes required to construct Domain Ontologies and a Policy-Driven Ontology as well as establishing the criteria for evaluating ontologies against each other that determines potential interoperability opportunities.

The second phase will uses the PDOAM to show the potential for data interoperability in two experiments; a theoretical one and an organizational one. In order to validate the model, the research uses a priori knowledge in both tests. By comparing the outcomes of the PDOAM to already known outcomes, the research confirms and validates that the proposed model can be used to determine the potential for data interoperability.

The data required for constructing the various ontologies for the experiments is taken from financial applications and related policies from within a large financial organization. The specific findings are presented in the evaluation of outcomes. A final conclusion is made as to the appropriateness and limits of the model based on the findings of the theoretical model in comparison to the a priori knowledge.

Proposed Model – Policy-Driven Ontological Analysis Model (PDOAM)

Semantic equivalence between a Policy-Driven Ontology and several Domain Ontologies is an indicator that the domain ontologies are in fact subsets of the Policy-Driven Ontology (Garigue 2004), which also establishes some semantic equivalence between the domain ontologies themselves. With a determination of semantic equivalence between domain ontologies, there is then a basis for asserting the existence of data interoperability between the domain applications. In the cases where domain ontologies are not subsets of a Policy-Driven Ontology, semantic equivalence cannot be established and there cannot be any interoperability between domain ontologies on that concept.

There is extensive research in this area, especially in terms of creation and testing of the processes that focus on evaluating some kind of ontology alignment (Huges et al. 2002; Kementsietsidis et al. 2003; Noy et al. 2000; Rosemann et al. 2000; Wand et al. 1996). The study by Washe, Voge, Stuckenschmidt, Schuster, Newman and Hubner on Ontology-Based Integration of Information – A survey of Existing Approaches (Wache et al. 2001) states that

"In a typical system, integration is done at the ontology level using either a common ontology (to which) all source ontologies are related or fixed mappings between different ontologies."

In this context the PDOAM is a novel framework that uses ontological analysis as a way to assess and identify potential interoperability issues between datasets that come from differences or inconsistencies in underlying concepts. Without conceptual equivalence

across multiple datasets, the integrity of the datasets is uncertain and the result is invalid assertion statements. Concepts that underlie datasets can be represented through ontologies. This research demonstrates that the use of Ontological Analysis is a new way to identify conceptual mismatches that would otherwise prevent interoperability. As part of the model, the notion of a Policy-Driven Ontology is introduced in order to establish the legitimacy of an enterprise-wide defined concept. The Policy-Driven Ontology acts as a reference model and is used to compare domain level ontologies against it.

The generalized model is shown in figure 2 and is characterized by the following phases.

1. Identification of a common concept
2. Creation of Domain Ontologies for that concept
3. Creation of Policy-Driven Ontology for that concept
4. Applying a set on alignment criteria between Domain ontology and the reference Policy-Driven ontology
5. Determination of potential for data interoperability between domain ontologies

Figure 2- PDOAM Generalized Model

Contribution to the Body of Knowledge

The main contribution of this thesis is the model used to represent and define the complete process of how to use ontologies for determine data interoperability. Here are some more elements of this process with more specific contributions:

- Firstly, using Policy-Driven Ontologies, one can apply engineering and management principles to develop and maintain a coherent and integrated enterprise information architecture that permits interoperability and reduces costs. This approach also introduces a new layer of abstraction independent of the constraints of technology infrastructure or the implementation process. Application development can leverage more efficiently information standards, common practices and an integrative environment.
- Secondly, ontologies help to identify the relevant concepts in the business and describe their relationships in a way that can be executed computationally through software applications across multiple business models. This will improve the effectiveness of business value chains.
- Thirdly, ontologies now become valuable information assets in of themselves as they enable the integration of more sophisticated and complex business models. By identifying critical information assets the organization has more control over its value creation processes.

Descriptions of the Research by Chapters

Chapter 2 presents the background to the research – specifically it presents a detailed description of the research agenda. In addition there is an outlining of the main element of the framework in a literature review of relevant work done in the ontological engineering domain. A description of the main definitions for the research is presented as along with a description of the motivating scenarios.

Chapter 3 describes the theoretical framework by which PDOAM is grounded through subsumption relations between concepts from different hierarchies as presented by Stuckenschmidt (Stuckenschmidt et al. 2004). A cursory description of the supporting ontological engineering environment is presented.

Chapter 4 describes the chosen research methodology. This includes an elaboration of the Policy-Driven Ontological Analysis Model and its scope. It further describes the structure of the experiments used to validate the model. This validation comes from two experiments; one based on theoretically constructed ontologies and the other using real-world datasets. In both cases a priori knowledge of the outcomes is known which enables validation of the capability of the model by confirming that the model is empirically adequate. In order to evaluate the similarities and differences between ontologies, two sets of criteria are developed; one for determining the alignment between ontologies and the other for determining interoperability between ontologies.

In addition to executing the model, Chapter 5 implements and exercises the criteria developed earlier to support the model for evaluating data interoperability via the two sets of trials; one with theoretical data and the other with real world organizational data. Outcomes of the tests are presented and observations with regards to findings in both trials are documented and presented for review and analysis. In both situations there is a priori knowledge as to the nature of potential data interoperability.

In Chapter 6 there is an overall analysis of the ability of the Policy-Driven Ontological Analysis Model to assess data interoperability. Discussion helps to define the consequences of the Policy-Driven ontological analysis model on enterprise ontology construction processes and ontology standardization.

Chapter 7 presents some overall observations and findings with regards to the limitations of the PDOAM and consideration with regards to its use. A specific description of the significance and contribution of the PDOAM to the ontological engineering research

program is also presented. Finally, a section on future research is developed, accompanied by the potential contribution of some aspects of Description Logic to the model along with recommendations for continuing research in Ontology Alignment using Interoperability Metrics.

A final Conclusions chapter gives closure to the dissertation.

CHAPTER 2

CONCEPTUAL FOUNDATIONS

The Data interoperability Challenge

In the past, solutions that enabled two different systems to talk together entailed mapping a transformation from one data set to another at various levels of abstractions (Renner et al. 1996; Renner 2001). There are various methodological approaches to data interoperability but most fall into two main broad categories of resolutions. In either case, the result of these approaches to interoperability results in tightly coupled systems which make them less able to evolve over time.

1. Extraction and reconstruction. This has been called the Extract-Transform-Load (ELT) process. The ELT method extracts data from one system on a scheduled basis by copying batch feeds of data from one data source, transforming the data, and then loading the transformed data into separate systems. Once the data mapping has been defined, any changes to the source file will invalidate the data. ELT has been a mainstream strategy for achieving interoperability. A series of tools and techniques have been constructed over the course of the last 10 years to enable field to field mapping. These tools, however, have not made a big impact on the practice of data integration (Margulius 2002) as their deployment has been limited. The dominant method at present is based on human intervention and resolution (Kementsietsidis et al. 2003). The limitation of this “craftsman” approach requires intimate knowledge of the particularity of the systems.
2. Wrappers and Middleware. This method permits interoperability by creating a common infrastructure and common standards. The Enterprise Application Integration (EAI) or Business to Business Integration focuses on various types of middleware solutions. This means the creation of pre-built adapters that provide bidirectional connectivity to many types of applications and data sources. These

integration solutions work by extracting or inserting data from these various adapter-enabled systems, transforming the data and converting the representation or schema to a different form and then shipping the data to their destination. This is a component-based solution which assumes that the various systems involved are similar in nature from a technical point of view.

These methodological approaches to the problems of data interoperability have been documented by Renner, Rosenthal, and Scarano, (Renner et al. 1996). The following are fundamental observations as to the problems encountered:

“The current approach to the data interoperability problem is to write ad-hoc data interface programs for each pair of communicating systems. Experience shows that development and maintenance of these programs is expensive in terms of both time and money. Worse, the total effort required increases with the square of the number of communicating systems. Finally, these hard-coded interfaces support only the information transfer anticipated during development, and not “pull-on-demand” transfers. It is plainly evident that the current approach cannot be made to support the requirements of the infosphere, or even those of the immediate future”

Renner, Rosenthal, and Scarano (Renner et al. 1996; Renner 2001) and specifically Rosenthal, Renner, Seligman, and Manola (Rosenthal et al. 2001) have suggested that data integration needs a different approach:

“We offer an alternate vision, one in which data integration is transformed into an industrial process through changes inspired by the Industrial Revolution in manufacturing, a process which produces data connections between islands, or between an existing island and a new integrating view”.

In addition Renner (Renner 2001) states that in order to get to that state two main methodological problems need to be solved:

“Two things must happen before two systems or organizations can understand each other’s data. First, the people involved must identify a matching real-world thing of mutual interest. (All data describes some thing of interest. You can’t meaningfully exchange data unless it describes the same thing.) We call this step establishing a semantic match. Second, they must arrange to eliminate or otherwise deal with the differences in the name, structure, and representation in the data models they use to describe this real-world thing. For example, if you measure a certain distance in miles, but I expect distances measured in kilometres, then the appropriate conversion must be applied to the data before I can properly use and understand it. This second step is coping with representation mismatch”.

Towards the end of the 1990’s a series of different methodological approaches to data mapping were proposed to solve interoperability issues. At the root of the problem was the fact that data mapping did not address the differences between the concepts. They identified the different semantics of the various databases as the impediment to data integration. This issue was the linchpin to easy and full data integration. Most recent approaches fall into two categories, Standard Business Taxonomies (for example the standard business reporting format XBRL or Extensible Business Reporting Language) and Formal Ontologies. The two approaches to interoperability today are:

- Standardization of content-based protocols (these are any xml derivatives such as XBRL, ebXML etc.) to promote and develop taxonomies as a way to build consensus on business data integration specifications.
- Creation of Formal Ontologies that act as abstract representations of the world (Gruber 1993; Guarino 1998). An object described in one language may not correspond to the same object in another language and so the formal semantics of

ontologies permits the translation process between the different data sources (Visser et al. 1997). The capacity for data integration and sharing depends on understanding how the concepts were operationalized in the various datasets.

Concepts and Ontologies as Basis for Interoperability

A concept is something that has been classified in a way that gives a general description of several similar instances of an event or object (Guizzardi et al. 2002; Sowa 1984). A concept is usually generalized from the classification of particular instances. In this context, interoperability opportunities must be assessed by working at a higher level of abstraction; one that permits a comparison between various concepts. Assessments can then be made on the opportunities for interoperability between datasets. If semantical commonality exists at a higher level of abstraction then the concepts used to represent the datasets are equivalent. Therefore, mediation through a similar concept at an ontological level means that the datasets will be interoperable.

If the concepts are not equivalent then there will be great difficulty in ensuring that an overall set of services or applications – whether it is a web service, a workflow, or a corporate memory – will continue to have coherence of meaning. Coherence of meaning is a requirement which ensures that, in the new context of interoperable services, the resulting data will remain accurate, consistent, and meaningful.

One way of analyzing concepts for the purpose of assessing interoperability is to represent them as ontologies. An ontology can be described as a model of a particular field of knowledge - the concepts and their attributes, as well as the relationships between the concepts. In the context of information systems research, the generally accepted working definition is that “ontologies are a consensual agreement on the concepts and relations characterizing the way in which knowledge in a domain is expressed” (Mentzas 2002).

A formal ontology is specified by a collection of names for concept and relationship types organized in a partial ordering by the type-subtype relation (Sowa 1984; Sowa 1991; Sowa 2000). A more operational definition would be that an ontology is a formal, explicit specification of a shared representation (Gruber 1989). It is considered formal because ontologies are defined using specific rules and considered explicit because they can be manipulated and extended for use by computational processes.

One of the capabilities of ontologies is the ability to help systematically order data structures to enhance the exchange of information between computers as well as between people. Furthermore, they enable the definition and sharing of domain-specific vocabularies. For that reason, in working with ontologies, one is dealing with the essential representations of an information system. Ontologies are the components of an information system which specifies what “things” exist and what is true about them. In fact, ontologies are starting to play a strategic role in representing information system issues. Already, the ontological engineering discipline is presenting workable solutions in the realms of Knowledge Management, e-Commerce and the Semantic Web which shows that ontologies are an integral part of the way ahead in addressing interoperability (Bruijn 2003; Peat 2003).

Literature Review

This literature review presents several generic background definitions of ontology that are used in the context of information systems interoperability. In addition, there is a review of an ontology’s underlying structure and functionality. The review ultimately results in a proposed working definition of an ontology for the purpose of this research. It is also important to understand that ontologies also have an underlying logic which permits a formal treatment and enable logical functions such as inferencing and classification. It is this formal nature of ontologies that serves as the basis for making interoperability assessments.

Defining Ontologies

Metaphysically speaking, Bunge (Bunge 1979) states that an ontology is the study of what there is; they are an inventory of what exists. Aristotelian logic was able to infer the nature of things by using predicate logic. Haugeland in his essays (Haugeland 1985; Haugeland 1998) shows that for the last forty years ontologies have been used in the context of knowledge engineering, artificial intelligence and computer science as a way of representing knowledge and extending inferences over the various propositions in order to create new assertions. These assertions are called “ontological commitments” (Choo 1998; Wand et al. 1999). An *ontological commitment* is a commitment to a claim of existence (Smith 1996). In order to ensure interoperability, an ontological commitment to “what is” needs to remain consistent. Zhao (Zhao 2003) states that an ontological commitment is the agreement by multiple parties (person and software systems) to adopt a particular ontology when communicating about the domain of interest, even though they do not necessarily have the same experiences, theories, or prescriptions about that domain. It is the structure of these commitments that will permit interoperability since entities remain consistent to their nature as expressed in the form of their compositions and unique structures. Entities that are similar in nature are similarly expressed.

An early definition of ontology is given by Neches, Fikes, Finin, Gruber, Patil, Senator, and Swartout, (Neches et al. 1991). They defined an ontology as:

An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary.

A few years later the now classic central definition of an ontology was given by Gruber (Gruber 1993) :

An ontology is an explicit specification of a conceptualization.

Both these definitions talk to what an ontology is, but do not describe an ontology in terms of a structure. Guarino (Guarino 1998) makes the point that an ontology is essentially :

A set of logical axioms designed to account for intended meaning of a vocabulary.

In the context of information system research the concept of ontology takes on a computational perspective. Swartout, Patil, Knight, and Russ (Swartout et al. 1996) define more specifically the structure of an ontology as:

An ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base.

Finally Noy and Musen (Noy et al. 2002a) talk to the main reason for ontologies within information systems:

"An ontology defines a common vocabulary for researchers who need to share information in a domain.

Most researchers agree that ontologies have a defined structure that can be independent of the content within which it is contained (Fensel 2001b; Gomez-Perez et al. 2004; Noy et al. 2002a; Noy et al. 2002b; Zhong 2002).

For the purpose of this research we will be adopting Gruber's definition of ontology as the main characterization because it talks to the central role of ontologies in the context of information systems as being a formal specification. However it needs to be extended with Noy's et al. obligation for sharing.

An ontology is an explicit specification of a conceptualization that enables sharing of information in a domain of knowledge.

As such, ontologies have the following structural components: classes, properties, relationships, and instances.

- A class typifies a concept. In the banking sector the notion of client, user, branch manager, and employee, are distinct concepts that can be represented by classes because they represent unique entities. Although they are part of a same ontology of people, each has associated differing properties and relationships. Classes can also represent abstract concepts (intentions, beliefs, feelings, etc.) as well as specific physically quantifiable concepts (table, chairs, cups, etc.)
- The properties (and values of the properties) of those classes represent the attributes of the class. The class of “cup” can have color as an attribute with the range of colors to represent all the attribute values. With regards to the class “customer” then one of the properties would be “name” and the property value would be defined over a string of letters. Properties can be constrained further constrained by rules or axioms (statements that are always true).
- Relationships represent a type of association between concepts in the domain of knowledge that is being represented and are usually binary relationships. Relationships can be used to connect different classes together. Specific relationships can also involve a defined functional assignment such as subclass, inclusion, exclusion, etc.
- Instances are used in ontologies to represent specific examples of a class. “Peter’s green cup” is an instance of the class “cup”.

In some cases there is a need for recursivity to come into play, as the instances of certain types of ontologies are, in fact, classes. Most of the definitions are concerned with the notion of ontology in the generic, but ontologies themselves have also gone through differentiation and specialization of functional roles (Gomez-Perez et al. 2004). The most fitting example of this differentiation comes from looking at the level of abstraction at which the ontology is used to conceptualize a domain of knowledge. At the highest level there are Top/Upper level ontologies that deal with meta-level knowledge. Then there are domain ontologies that focus on structuring knowledge within specific areas of understanding such as the natural of physical sciences. These ontologies are used to define business models and medical domains. Finally there are task ontologies that provide the semantics for processes and relationships (Daconta et al. 2003). This tiered structure is derived from the seminal work of Sowa in conceptual graphs and semantic networks (Sowa 1984; Sowa 1991; Sowa 2000). The practices relating to the life cycle of ontologies such as creation, storage, maintenance and evolution is now being gathered together under the practices of ontological engineering and ontology management (Gomez-Perez 2004).

The main focus of ontological engineering is to provide theories, support systems and practices for expressing structured knowledge and for accessing and reasoning with it in a formal and logical way over time.

Evaluating Ontologies

One critical aspect of managing ontologies is to evaluate them. The evaluation of ontologies has been described by Gomez-Perez (Gomez-Perez 2004). Here is her list of criteria that can be used to evaluate ontologies. They are as follows:

- **Consistency:** Refers to whether it is possible to obtain contradictory conclusions from valid statements. A given definition is consistent if and only if the individual definition is consistent and no contradictory sentences can be inferred from other definitions and axioms.

- **Completeness:** Incompleteness is a fundamental problem in ontologies. On facts alone, it is impossible to prove either the completeness of an ontology or the completeness of its definitions. However, the incompleteness of an individual definition can be proven and thus deduce the incompleteness of the ontology and the incompleteness of an ontology if at least one definition is missing in the established frame of reference.
- **Conciseness:** An ontology is concise if it does not store any unnecessary or useless definitions; if explicit redundancies between definitions of terms do not exist and if redundancies cannot be inferred from other definitions and axioms.
- **Expandability** refers to the effort required to add new definitions to an ontology and more knowledge to its definitions without altering the set of well-defined properties already guaranteed.
- **Sensitivity** relates to how small changes in definitions alter the set of well-defined properties already guaranteed.

The PDOAM ensures that the Policy-Driven Ontologies are regulated to be consistent and concise because the creation of a Policy-Driven Ontology is from within a real-world organizational context and therefore are subject to the validity requirement of correspondence.

In addition, consistency is achieved through the use of Policy-Driven ontologies when it acts as a common specifications across various applications for example in the context of a B2B standard. (Uschold et al. 2004), The creation of a Policy-Driven Ontology developed from within a real world organizational context also permits to address the completeness requirement as completeness is determined in relation to an established frame of reference. This means the Policy-Driven ontology act as the complete ontology for an entity to which domain ontologies can be compared.

Overall, when an ontology meets consistency, completeness and conciseness, it is deemed to be considered a “competent” representation of a concept. This is a variation from the previous definition of competency only in that there is no series of questions that are specifically attributable to an ontology during its development.

A competent ontology still “contains necessary and sufficient set of axioms to represent and solve these questions, thus providing a declarative semantics for the system. It is in this sense that we can claim to have a competent ontology” (Mark S. Fox 1998).

Concept Interoperability

The challenge of maintaining coherence of meaning (through data integrity) in a situation of dataset interoperability must be addressed first and foremost at a conceptual model level. Organizations typically manage a diversity of information systems that represent multiple world views such as finance, human resources, sales, inventory etc. Some of these world views are based on similar concepts for which interoperability would present only technical problems. In other cases, notwithstanding that information may be labelled similarly or have some elements in common, applications are using datasets based on representations of fundamentally different concepts (Resnik 1998). In these cases, trying to make them interoperable becomes an impossible task. It is critical that an organization know how to identify conceptual inconsistencies.

So how does an organization faced with growing pressure to interconnect various information assets and repositories give the assurances that these various resources are being appropriately mapped so as to match up with one another? This question is being asked, but at present, most answers and solutions are inadequate. Organizational initiatives to make various applications interoperable have very little theoretical frameworks or models to go on and the present practices do not guarantee success (Tozer 1999).

Gomez-Perez, Fernandez—Lopez and Corcho (Eriksson et al. 1999; Gomez-Perez et al. 2004) and Eriksson, Fergerson, Shahar, and Musen (Eriksson et al. 1999) have determined that the causes for failure of conceptual interoperability can be categorized broadly into two main categories:

1. Incomplete concept classification. This occurs when concepts are classified without accounting for all essential aspects of the concepts.
2. Semantic errors. These occur when the ontologist makes an incorrect semantic classification, that is “s/he classifies a concept as a subclass of a concept to which it does not really belong: for example the concept of plane as a subclass of the concept of house”.

In both these situations, the result is conceptual ambiguity (Garigue 2003; Shanks et al. 2003; Tozer 1999). Conceptual ambiguity occurs when there are two or more simultaneous interpretations of a concept, all of which can be supported by the context. For example the concept of “customer” in addition to be a part of a large entity such as business or market (from a logical perspective) could be a specific individual, a role, or even a relation. Applications based on these kinds of mereological (study of the “parts of”) difference would not be interoperable. Any attempt to force these datasets to work together in a common application would create uncertainty in the organization (March 1999; Simon 1977; Weick 2001; Wiig 1993) which is counter-productive to any organization.

This particular issue was picked up in an ontological analysis of the concept of “customer” done by Partridge (Partridge 2002). He indicates that there are three ontological perspectives: as a Particular, as a Role, and as a Relationship. Each one of these perspectives has a substantial influence in the resulting structure of the ontology. In business information systems the concept of customer as a particular is prevalent and includes most of the personal and descriptive information. The perspective of a customer as a role would include, in addition to information as a particular, information such as

rights and privileges associated with the role of being a customer. Finally, an ontology of a customer as a relationship would also include information regarding the binary relationship between the specifics of the individual, his/her rights and privileges as determined by a relationship context as it pertained to relationship across several businesses with the customer as a link instead of a node.

The presence of an identical concept across multiple applications offers a great opportunity to link data together using this common concept. The conception of customer can act as a common concept that supports interoperability between applications dealing with various services offered to customers. But there is a lot of similarity between the concepts of *customer* with the concept of *client*. But are they two different concepts or are they equivalent? The context will help determine if they are equivalent by looking in more detail as to how the concept is structured as an ontology. By having concepts represented as ontologies we can establish a formal framework for analyzing and comparing them to one another.

Questions arise in designing applications that support business and organizational processes. For example, would the notion of *user* be a better concept than *customer*, if this concept specifically refers to an internal *service* for an *employee*? Did the system developer or requirements analyst misidentify the concept of *client* with concept of *customer*? There could be cases in which the notion of *customer* is similar to the one of *client* but in most contexts, the notion of *customer* and *client* are different concepts. The consequence of linking the datasets of each of these systems would result in a data integrity failure of the resulting service. Analyzing these different concepts through their ontologies will enable the organization to gain insight into the specifics of semantic similarities and differences. This will ultimately enable it to make an assessment of their potential meaning and evaluating their semantic interoperability.

The notion of ontological ambiguity has been proposed for some time as an ontology engineering issue to be deal with (Garigue 2003; Visser et al. 1997). Researching data interoperability, does also permit the study of ontological ambiguity as a corresponding

issue (Nirenburg et al. 2001) But at present this is not the focus of this research. The objective of the proposed ontological modeling methodology is to recognize opportunities for data interoperability through ontological analysis.

There are many cases where applications and their associated datasets have permitted designers to avoid resolving such incompatibilities because the scope of the concept is limited to a specific application and there is no need to address this issue as an enterprise problem. However organizations now need new ways to assess the opportunities/challenges that interoperability posits. Attempts to join or link applications based on inconsistent or ambiguous concepts results in a failure to meet data integrity requirements in an organization. This condition will in effect destroy the value of the resulting information and business services. Without ensuring that all concepts are unique, distinct, and without ambiguity, organizations will also fail in their sense making and decision making activities (Choo 1998).

As mentioned earlier, interoperability is the ability of two or more systems or components to exchange data and information and to use the exchanged information in a meaningful and consistent way. In the context of working with ontologies, this means establishing a correspondence between various ontologies. At the present time there is no definitive approach as to how best to perform this activity. The following literature review therefore focuses on the principal elements of this debate.

Interoperability as Concept Coherence

Concept coherence is used to ground interoperability initiatives (Garigue 2003). As Thagard (Thagard 2000) states:

"Given a large number of elements (propositions, concepts, or whatever) that are coherent or incoherent with each other in various ways, how can we accept some of the elements and reject others in a way that maximize coherence?"

This requirement for ideas to "fit" in the pre-existing cognitive system may be called coherence. Not only is there a need for a concept to have internal coherence, but elements should not have contradictory properties (unless they can be explained at a higher level, which would restore concept coherence). In this context, coherence in the PDOAM is established through an analysis that results in a match/mismatch between a Domain Ontology and the Policy-Driven Ontology. When the two are interoperable the Domain Ontology is a subset of the Policy-Driven Ontology and makes obvious the fact that the concept has internal coherence. Coherence thus ensures validity of the overall information architecture of the enterprise (Kalfoglou 2002). Domain ontologies describing concepts cannot be accepted as valid unless they preserve common elements and relations with respect to what has been described through an enterprise-level policy.

The difficulty remains in the definition of "coherence" in operational terms. Mere consistency (such as the use of common terms) is clearly not sufficient for interoperability, since any collection of facts relating to those common terms could be logically consistent. Coherence derives from the connections and mutual support of the different beliefs. Since learning is based on the strength of associations, ideas that do not connect to existing knowledge simply cannot be learnt (Thagard 1999; Thagard 2000). These connections represent an associative semantic relationship in that information about one idea also gives some information about the other. A concept can furthermore support another concept by providing an explanation, evidence or arguments as to why the latter idea should be true. Concepts are nodes in a network of meanings.

Thagard asks "how can coherence be computed" (Thagard 2000) ? He answers this by referring to a method of resolving semantic ambiguity by Cotterell (Small et al. 1988) that proposes a constraint satisfaction method. This approach relies on a connectionist model of lexical access in which alternative interpretations of an ambiguous word can be evaluated by representing alternative meanings as nodes in a constraint network. In this approach "bank" can be represented, in terms of association. One interpretation of the word is associated to "river" and the other interpretation is associated with "money". The requirement for not just enumerating alternative meanings but for also establishing linked relationships with them creates higher-level structures of meaning. These relationships structure meaning into semantic networks and offer not just analysis of the terms and relationships but also of the structures themselves. The proposed Policy-Driven Ontological Analysis model takes this into consideration as it deals not just with the terminological elements of a concept, but also with structural attributes and properties. This

approach characterize concepts at a greater level of semantic detail (Moen 2001; Noy et al. 2000; Sowa 1991).

From Concept Interoperability to System Interoperability

As acknowledged by Riichiro Mizoguchi, Mitsuru Ikeda, and Katherine Sinita (Riichiro Mizoguchi 1997) the challenge facing the present ontological engineering research program can be explained in one sentence: “Most of the conventional software is built with an implicit conceptualization. The new generation of AI systems should be built based on a conceptualization represented explicitly”. Using ontologies makes explicit some aspects of conceptualization. Systems can then extend these conceptualizations through inferences. In addition, explicit conceptualizations can construct new argumentations through derivation of meaning.

Another reason for making conceptualizations explicit is that they become semantic boundary objects between information systems. Some of these information systems are social in nature (such as belief systems) and therefore conceptualizations become understandings and agreed-upon concepts. But other information systems are technical. In these systems common conceptualizations need to refer to common data and information models whose purpose is to disambiguate in order to achieve overall application interaction and data interoperability. In this context, interoperability is the ability to use data from disparate applications to communicate meaningfully through the use of common conceptualizations.

Conceptualizations can also be embodied in knowledge structures at higher levels of abstractions. Concepts are thus related to one another through various meta-structures and hierarchies. These hierarchies are also conceptualizations that enable data exchanges and act as a common framework for enabling system interoperability. Even at this level, where the focus is on the metadata structures in order to achieve semantic interoperability, there is still a requirement for conceptual coherence.

Ontologies as Standards of Core Business Concepts

There is a lot of research effort underway to establish the basis for interoperability. Some of the work is done within the ontological research initiative via controlled meta-ontologies such as Suggested Upper Merged Ontology (SUMO) and Mid-Level Ontologies (MILO).

The SUMO and MILO initiatives were developed to support the creation of standard ontologies that ensure data interoperability, improved information search and retrieval, access to automated inferencing, and natural language processing. The upper and mid-level ontologies are limited to concepts that are meta, generic, abstract and philosophical, and therefore are general enough to address (at a high level) a very broad range of general domain areas. Concepts specific to specific domains such as finance, business, or government would not be included; however these standards would provide a functional framework and a set of general abstract concepts upon which domain ontologies (e.g. medical, financial, engineering, etc.) can be constructed.

At the level of domain ontologies, there are multiple organizations such as Organization for the Advancement of Structured Information Standards (OASIS) via standards such EbXML, IFX, and XBRL. For example electronic business XML (ebXML) is a project jointly initiated by UN/CEFACT (The United Nations body for Trade Facilitation and Electronic Business) and OASIS to standardize XML business specifications; ebXML is a technical framework that enables XML to be utilized in a consistent manner for the exchange of all electronic business data.

XBRL is a language for the electronic communication of business and financial data which is intended for use in formal business reporting around the world. It provides major benefits in the preparation, analysis and communication of business information. It offers cost savings, greater efficiency and improved accuracy and reliability to all those involved in supplying or using financial data. XBRL stands for Extensible Business Reporting Language. It is one of a growing large population of standard "XML" languages which are becoming common means of communicating domain information

between businesses. In Canada, as in the US and in Europe, XBRL is being used all the time. In Canada the development of the XBRL standards for Canadian GAAP (General Agreed Accounting Procedures) has been led by XBRL Canada, of which the host member is the Canadian Institute of Chartered Accountants.

Emerging XBRL applications can enable analysts and investors to track specific data points from company disclosures and to electronically integrate them into spreadsheets and other modeling tools. This is intended to reduce the amount of time it takes to locate the required data, reduce the risk of re-keying errors and increase the speed of integration of data into analytical models. Also, there is now the Interactive Financial eXchange (IFX) Forum for the electronic business-to-business exchange of data among financial service institutions around the world focusing mainly on Automatic Business Models (ABM) and Point of Sales (POS) devices.

In the list of growing standards there is an underling assumption that the entities and concepts that are underpinning for these frameworks are themselves interoperable. As these frameworks have not been harmonized, there is still the requirement to have a method to evaluate semantic similarity to ensure data interoperability. At best, there will be a gap between these various standards frameworks and at worst these frameworks will not offer the semantic integration expected because they cannot ensure that all concepts will semantically equivalent

The Universal Data Element Framework (UDEF) seeks to ensure the use of common referent ontologies that span these various business frameworks. This work seeks to ensure interoperability among business standards. The objective of the UDEF is to provide a means of real-time identification for semantic equivalency as an attribute to data elements within e-business documents and integration formats. Given that system integration needs common semantics the UDEF initiative promise is to act as an integrative support environment for the emerging Semantic Web.

Even after a review of standards groups' present efforts to resolve the problem of data interoperability, it is clear that there is still a need to be able to compare the various underlying ontologies to one another in order to determine which ones are semantically equivalent to each other. Ultimately, it is still fundamental to have a working model that will help resolve which concepts and their ontologies are semantically similar and which ones are dissimilar.

Ontological modeling within information systems research is still nascent, but increasing in scope and applicability. Domain experts will benefit from this investigation. Additionally research in ontological modeling will also impact the quality of many knowledge artifacts as well as offer an improved knowledge management process.

The Use of Ontologies in Data Interoperability

The main focus of the present broader ontology research program is on how to integrate or align ontologies so that they can work together across broader automated contexts. Gomez-Perez, Fernandez--Lopez, and Corcho (Gomez-Perez et al. 2004) make the essential point that "most of the methods and methodologies for building ontologies are focused on the development activities, especially on the ontology conceptualization and ontology implementation, and they do not pay too much attention to other important aspects related to management, learning, merge, integration, evolution and evaluation of ontologies".

Doan, Madhavan, Domingos, and Halevy (Doan et al. 2002) also work on this problem as part of their review of how learning processes fail when ontologies are expected to categorize instances from misidentified concepts. Finally Hameed, Sleeman, and Preece, (Hameed et al. 2002) have been able to qualify these mismatches into an initial classification of mismatches. Here they are following on the first attempt of classification done by Visser, Jones, Bench-Capon, and Shave, (Visser et al. 1997) that were looking for root causes of mismatches. Additionally, work by Maedche, and Staab (Maedche et al. 2002) focuses efforts on looking at measures for evaluating similarities and differences between ontologies as a basis for interoperability. Kashyap, and Sheth

(Kashyap et al. 2002) talk to the same issue but in terms of conceptual closeness. Ram and Park, (Ram et al. 2001) go to the next step and put forth some of the requirements for concept conflict resolution. With regards to the issues of dealing with interoperability across multiple ontologies, the literature review (Ram et al. 2001; Rosemann et al. 2000) indicates that there is limited capability.

As seen there are a lot of discussions but no agreed-upon or proven methodology that definitively addresses interoperability issues at the dataset level. Therefore, identifying interoperability issues and concerns first at an ontological level will enable more productive work. In their example about the concept of profit in financial institutions, Firat, Madnick, and Grosz, (Firat et al. 2002) shows how the same concepts are used differently depending on the context and how concept ambiguity becomes the source of multiple data corruption and application misalignments.

At the present time, approaches to data interoperability using ontologies mostly focus on developing a way to map and match the various common elements such as class and attributes of the different ontologies (Kalfoglou et al. 2003). However it can be said that the authors agree that an overall complete mapping and matching between two ontologies would mean that the concepts represented in each ontology are equivalent to one another. In other words if the classes of an ontology match the classes of another ontology, then these two ontologies can be considered equivalent within those classes. Therefore interoperability assessment initiatives have to focus on resolving the various mapping functions between one ontology and another.

A review of the present literature, mostly on ontology matching processes (Gomez-Perez et al. 2004; Kalfoglou et al. 2003), indicates that there are three main accepted theoretical models. These are: the use of a shared structured and formal vocabulary; the use of a unifying top/upper-level ontology; or the construction of a common ontology based on common shared instances to provide a basis for matching functions.

1. Matching ontologies using a shared vocabulary.

This approach is the most common within well-structured knowledge domains such as engineering and medicine. They are considered well-structured because they have common and controlled vocabularies. The use of controlled vocabularies is important for consistency, precision, and completeness. Common formal vocabularies also provide the foundation for the construction of metadata structures and associated query systems. The use of a shared formal and structured vocabulary becomes a unifying control mechanism and enables the resolution of interoperability issues.

2. Matching ontologies through the use of a Top or Upper-Level Ontology.

A Top or Upper-Level Ontology is limited to concepts that are considered meta-level, generic, abstract and philosophical, and therefore are too general to address (at a high level) a broad range of specific domain areas. Concepts that are specific to given domains cannot be included; however this standard-based approach provides a structure and a set of general concepts upon which domain ontologies (e.g. medical, financial, engineering, etc.) can be constructed. The use of an overarching framework permits conceptual integrity independent of a specific application domain.

3. Matching ontologies through common instances:

A set of common instances is also a method for integrating multiple ontologies through a common categorization. This method provides a mapping function between ontologies and can be used to transfer specific instances from one domain ontology to another.

Maedche and Staab (Maedche et al. 2002) and Ram and Park (Ram et al. 2001) have presented approaches that have had some successes in matching up ontologies to one another. Their matching process does establish some basis for evaluating interoperability at the conceptual level; however this research takes it further by adding a requirement to verify that the concept is also interoperability within the legitimate enterprise definition for that concept.

In the broader context of an organization, there exists the problem of having to deal with multiple ontologies while at the same time have to resolve the legitimacy of that concept. The question therefore now becomes, not just of matching ontologies to one another, but also of determining the source of authority from which a concept can assert a dominance that will also align ontologies within organizational definitions and expectations.

To accomplish this, a different entity called a Policy-Driven Ontology will be created to resolve this issue. Further details will be presented in the coming sections but a Policy-Driven Ontology can be categorized as having some aspects of an upper level ontology, because it would play the role of a global structural reference, as well as having some aspect of a domain ontology. As presented by Weinstein and Birmingham, (Weinstein et al. 1999), when talking about differentiated ontologies, a Policy-Driven Ontology will also help by establishing a shared common vocabulary. This because, a Policy-Driven Ontology of a concept also contains the full vocabulary that can be used to describes the domain of knowledge under analysis. This is discussed in further detail in following sections.

CHAPTER 3

POLICY-DRIVEN ONTOLOGICAL ANALYSIS MODEL

The main focus of the research is to determine how data interoperability can be determined through the use of a Policy-Driven Ontological Analysis Model (PDOAM). In support of this objective there is a requirement to determine a set of criteria that need to be applied in order to make that determination (Davies et al. 2003; Obrst 2003; Obrst et al. 2001). This chapter describes the overall process of constructing, abstracting and comparing ontologies in order to evaluate potential for data interoperability across applications.

This chapter will establish the process and criteria that will be used later to demonstrate capability. This will be done through the use of a tests. In addition to testing theoretical data sets the model will also be applied in the context of a real world problem. In this practical example domain ontologies of a common concept are constructed from three different information system applications. The domain ontologies are analyzed and compared to a standard ontology called a Policy-Driven Ontology (PDO) which is developed from organizational requirements (as opposed to a system). A priori knowledge of outcomes is used to show that the model can identify interoperability between data sets.

As mentioned, the Policy-Driven Ontology (PDO) acts as the standard or reference model against which domain ontologies are mapped, compared and analyzed. The semantic mapping between the concepts is evaluated through a process of comparative analysis. This process uses a set of concept alignment criteria that derives from the expectation that the final interoperable system meets an overall requirement of concept coherence (Garigue 2004).

This chapter describes the framework and criteria by which ontologies are compared and evaluated in order to determine potential data interoperability.

Justification for a Policy-Driven Ontological Analysis Model

The objective of the proposed Policy-Driven Ontological Analysis Model (PDOAM) is to identify conceptual matches and mismatches that are part of the interoperability challenge inherent in all applications. Trying to establish some kind of framework to evaluate the equivalences of concepts at the data level requires a capability to do semantic mapping and analysis.

This semantic analysis model needs to evaluate terms and structures in relation to each other in order to determine meaning. This stage of the research focuses on developing a set of criteria that will be used to assess data interoperability at a semantic level. This means that there is a requirement for our model to compare the concepts that are embedded in each of the applications and represented implicitly in the data structure. In order to be of benefit a PDOAM will have to also address questions of ontology extraction and abstraction. This is examined later in the research.

There are a lot of ways to represent concepts. Most involve structuring a network of meaning by categorizing entities and their properties (Smith 1996). The best approach is to use ontologies as a way to represent and structure concepts. Ontological engineering enables the construction of concepts in a way that they can be analyzed as well as leveraged as part of broader organizational strategies relating to information and knowledge management (Butler et al. 2003). This approach is similar as what is done in the context of other information system research methodologies. However the present state of ontological engineering research indicates that is not possible at the present time to compare ontologies in a fully automated way.

Interoperability initiatives try to integrate various applications, but without ontological engineering support or the benefit of any ontological analysis. At present interoperability initiatives is simply done in an ad hoc manner or through unstructured conceptual

analysis (Rosenthal et al. 2001). If the application interoperability challenge is small and encompasses only a few applications then it is possible to address the semantic analyses in an intuitive way, but in large organizations where there are hundreds of applications it is important to have a structured method that scales. And yet in all cases, especially prior to any integration initiative across data sets or applications, there needs to be a way to determine semantic matches or mismatches. This is the role of the Policy-Driven Ontological Analysis Model (PDOAM).

The Motivation for Policy-Driven Ontologies

The choice of the term *Driven* in the context of Policy-Driven Ontologies expresses the need for organizational control and design over data. The term is used as a warrant for application interoperability. This might be a small point of argument but when corporate ontologies are created and legitimized, these constructs factually “drive” the business models and shape the structure and the interoperability efforts (Firat et al. 2002; Kalfoglou 2002; Kim et al. 2002). Eventually Policy Driven Ontologies structure of the organization itself will simply be instantiations of the various types of ontologies within an organization. As form follows function, policy-driven ontologies determine the evolution of the information architecture in terms of the how the integration of legacy systems is done within the context of new applications (Garigue 2003; Resnikoff 1989). This is also supported by analysing the expectancies for consistency, transparency and traceability that derive from IT Governance regulations such as Sarbanes-Oxley (SOX) in the US and the Personal Information Protection and Electronic Document Act (PIPEDA) in Canada. Ensuring consistency, transparency and traceability of information across multiple applications requires some type of organizational control over content structure, and ontologies enable organizations to meet these requirements.

Policy-Driven Ontologies change the economics of information management in organizations. As in the case of data reuse, there is a perennial tension that exists between optimizing local systems, which are developed at lower cost vs. optimizing enterprise systems, which tend to be developed at higher cost (Boisot 1998; Borghoff et al. 1998a; Cohen et al. 1998; Davies et al. 2002; Fensel 2001b). These economic realities change when reference ontologies – Policy-Driven Ontologies - are used as standard information architecture components. The use of standard policy based ontologies will “drive” applications development costs down. This is now evident from research

in the structure of B2B markets. (Fensel 2001a; Fensel 2002; Firat et al. 2002; Gomez-Perez et al. 2004; Obrst et al. 2001; Zhao 2003)

The Efficacy of Policy-Driven Ontologies

As presented in Garigue (Garigue 2004) and (Garigue 2003), the construction of a Policy-Driven Ontology also permits the reduction of complexity of data mapping which is inherent in all application interoperability and integration projects. When a large number of applications are play, the most common approach to mapping data sets from one application to another is to set up various types of functional translation tables. This, however, does not take into consideration the semantic dimension that a data interoperability initiative is trying to resolve. If the semantic level is not taken into account and a semantic mapping between the data sets is not ensured, then it is very probable that the initiative will result in data corruption. Essentially, a conceptualization mismatch would result in a data integrity failure. A Policy-Driven Ontology Analytical Model is put forth specifically to resolve this problem.

As mentioned, previous attempts to enable interoperability of data required a mapping of each of the applications dataset to each other as well as to each of the other application datasets for which interoperability was made a requirement. This approach quickly result in a worse situation as the

organization has to now manage a C^2 mapping problem. Every time one of the applications changes a data set then a complete re-evaluation of each of the relationship has to be conducted. Eventually, if the interoperability requirements keep increasing, the organization becomes unable to maintain overall conceptual coherence and correspondence across all the data translations elements of the entire set of applications.

By introducing Policy-Driven Ontologies into the problematic, an organization can reduce the complexity of interoperability challenge to a n mapping and comparisons problem. This is one of the benefits of the model as organizations try to leverage more and more of the datasets found in the various data repositories and warehouses. As presented in the background section, current approaches to interoperability have not been very effective.

In terms of process, ontologies will be constructed from the datasets themselves. Domain Ontologies (DO) can be constructed from the ERD of the applications themselves and other ontologies can be derived from an enterprise wide policy and become Policy-Driven Ontologies (PDO). In order to evaluate these different classes of ontologies, a set of criteria for analyzing the similarities and differences between ontologies are proposed and incorporated into the model.

The PDOAM methodology uses an ontological analysis model that compares Domain Ontologies based on the datasets themselves with Policy -Driven Ontologies based on enterprise declared and defined concepts that comes the enterprise policies themselves.

Ontology Heterogeneity and Semantic Integration across Multiple Systems

Stuckenschmidt and Harmelen (Stuckenschmidt et al. 2004) present a framework for addressing Semantic Integration across multiple systems in light of ontology heterogeneity. This framework leverages subsumption relationships to determine concept membership. This is the generic theoretical framework and the basis for the PDOAM.

"An abstract definition of an information source is given in terms of a set of information items that are classified according to a source ontology [...]. In the case of a conventional data base, we can think of a single row in a database table as an information Item. The connection to a source ontology can be given by a corresponding reference in the data dictionary that may also contain the ontology itself".

In this context an information source in a domain ontology is a tuple $\langle O, I, M \rangle$, where $O = \langle S, C, d \rangle$ is a source ontology with a shared ontology S , a set of classes names C and a mapping d that assigns a class definition over terms from S to every class name from C , I is a set of information items and $M: I \times C$ is a membership relation that relates information items classes of the source ontologies

Based on this abstraction, the problem on integrating the classifications across two different information systems is then defined in terms of an extension of the M_1 membership relation to and information source: IS_1 by an additional relation M' that relates information items to classes of the second information source IS_2 according to the source ontology of IS_1 .

“In order to reason about instances of classes as well, we extend the semantics of a terminological language in a straightforward way, by assuming that the assignment mapping A does not only apply to class names but also to instances. We define that”:

$$x^I \in W \text{ for every } x \in I$$

“the notion of an individual relation to class $x : C$ is defined as follows:

$$x : C \Leftrightarrow x^I \in C^I$$

In this we can state the following definition with regards to the interoperability problem. It is resolved under these conditions:

Let $IS_1 = \langle \langle S_1, C_1, d_1 \rangle, I_1, M_1 \rangle$ and $IS_2 = \langle \langle S_2, C_2, d_2 \rangle, I_2, M_2 \rangle$ be information sources; then a bilateral interoperability is equivalent to finding a membership relation $M : I_1 \times I_2 \times C_1$ such that for all $x \in I_1 \cup I_2$ and $c_i \in C_1$:

$$(x, c_i) \in M \text{ iff } x : d_1(c_i)$$

So in the context of this theoretical framework for interoperability – to generate this new relation M' they have to rely on the semantics of both information sources that are given by a common source ontology.

As it cannot be assumed that both information sources use the same source ontology or that they are even comparable in the first place then these assertions are to be made in

regards to the relation of the ontologies of the two sources of information. There needs a way to reason across these various ontologies. This is given in turn if both source ontologies share the interpretation $\mathfrak{I}$ of concept terms. This can be done if both source ontologies are based on the same shared ontology.

From this theoretical framework it can be concluded that the source information thus declares the Domain Ontologies and a shared ontology declares a Policy-Driven Ontology.

Ontologies Integration and Description Logic

As reviewed earlier, ontologies are a formal structure of explicit knowledge. In order to operationalize an ontology we will use Gomez-Perez, Fernandez-Lopez and Corcho (Gomez-Perez et al. 2004) approach that adopt Gruber's (Gruber 1993) proposition to model ontologies using frames supported by first order logic.

Because this formalism is grounded in a specific class of logic, certain types of assertions can now be inferred across some types of formal ontologies. Ontologies can be extended using a class of logic called Description Logic (DL) that supports logical inference and categorization. Description logics are sometimes called terminological logic, classification logic, or concept logic. Description logic is a knowledge representation language that expresses knowledge about concepts and concept hierarchies.

Ontologies are usually given in a Tarski (Choo 1998; Tarski 1941; Tarski et al. 1994) declarative semantic style - the notion of truth, as applied to closed sentences - which dictates that ontological commitments are treated in the same way as a sub-language of predicate logic. Declarative Logic is considered an important formalism that gives a logical basis to the well known "frame-based" systems, semantic networks, and other knowledge representations such as object-oriented representations and type systems. Daconta (Daconta et al. 2003) indicates that frame-based languages are also aligned with

the object-oriented modeling paradigms because the knowledge of a domain is centered around entities in a domain.

The basic building blocks of DL are concepts, roles, and individuals. Concepts describe the common properties of a collection of individuals and can be considered as unary predicates which are interpreted as sets of objects. Roles are interpreted as binary relations between objects. Description logic also defines a number of language constructs (such as intersection, union, role quantification, etc.) that can be used to define new concepts and roles.

The main reasoning tasks are classification and inference, subsumption, and instance checking. Subsumption represents the “is-a” relation. Classification is the computation of a concept hierarchy based on subsumption. Description logic systems have been used for building a variety of applications including conceptual modeling, information integration, query mechanisms, view maintenance, software management systems, planning systems, configuration systems, and natural language understanding.

A description in the context of DL is an expression in a formal language that defines a set of instances or tuples. A frame-based knowledge representation language uses the data structures of a frame to structure knowledge in a predefined, formal way.

In accordance with Artale, Franconi, Guarino, and Pazzi, (Artale et al. 1996) and Gomez-Perez, Fernandez--Lopez, and Corcho (Gomez-Perez et al. 2004) it is possible to state that DL is based on declarative formalism that permits the defining of concepts in multiple hierarchies that follow a theory of parts which is a formal way of expressing an “is-a” relationship. Theories of parts and wholes have been studied since early philosophers started to investigate the essence of the physical world. The first formal accounts appeared only at the beginning of our century, dating back to the independent works on Mereology by the polish logician Stanislaw Lesniewski and on the Calculus of Individuals by Leonard and Goodman (Leonard et al. 1940). The most important of them may be informally summarized as follows:

- Extensionality: Two individuals are identical if and only if they have the same parts.
- Principle of Sum: There always exists the individual composed by any two individuals of the theory (i.e., the mereological sum)
- Supplementation: If an individual x is a proper-part-of an individual y , then a different individual z exists which is the missing part from y .

Description Logic does provide the theoretical basis for comparing and inferencing over the domain of knowledge represented in ontologies. Interoperability problems now become a determination of the equivalence between various parts of the domain ontology as it is compared to a common ontology:

1. If elements A and B are parts of the same thing (C) through extensionality;
2. and elements E and F are parts of (D)
3. then $(A$ and $B)$ and $(E$ and $F)$ are equivalent if C and D refer to the same entity in Z .

Another way of expressing this is through the following mapping:

- $O = (S, T)$
 - O = Ontology
 - S = Structured relations of concepts – Linked definitions
 - T = Terms - Labels of concepts
- Ontology Interoperability as a Mapping Function $O_1 = (S1, T1); O_2 = (S2, T2)$
 - Full Interoperability via a full mapping : $f : O_1 = O_2$
 - Partial Interoperability via Partial mapping: $f : O_1 \subseteq O_2$ OR $f : O_2 \subseteq O_1$
 - Non Interoperability via Disjoint mapping : $f : O_1 \cap O_2 = \emptyset$

This assures that classes and attributes from different ontologies are equivalent to each other if they are also equivalent to a similar class and attributes across a common ontology.

In the past, knowledge-based applications were restricted by the fact that they used a distinctive ontology language such as Knowledge Interchange Format (KIF) or the DARPA Agent Markup Language extended with the Ontology Inference Language (DAML+OIL). However with the emergence over the past few years of standards such as Web Ontology Language (OWL), Resource Description Format (RDF) and Topic Maps, which are logically equivalent to a frame-based language, the capacity to build and deploy ontologies on which there can be formal reasoning is now starting to be commonplace within ontological engineering communities of practices.

Ontological Engineering Communities of Practices

The present state of the research also shows that there is no real dispute about the theoretical grounding of ontologies. Most all major reviews and research indicate a broad acceptance of the practice of ontological engineering (Gomez-Perez et al. 2004; Knublauch et al. 2003).

It is interesting to note that the community of researchers in ontological engineering itself is fairly well defined because it is very small. Europeans – mostly the German, Italian and Spanish universities - dominate the theoretical landscape and North Americans dominate the technical and toolset landscape. The US and UK in particular are now driving the second wave of products supporting information system analysis at the ontological level. In fact more and more organizations are implementing ontology-enabled applications and leveraging the advances in ontological engineering environments (Gruninger et al. 2002) (Kalfoglou et al. 2003).

Here again there is also agreement within these communities that the next challenge information systems and organizations will face is to have semantically aligned ontologies to interoperate – as a way to share and reason over a broader knowledge landscape that is accessible (across the web) and across organizations. Hameed, Sleeman, and Preece (Hameed et al. 2002) make the point that most people do not realize that they are using and leveraging multiple ontologies that are deeply embedded within each enterprise application every day. At present time, even with the presence of DL within ontologies there is no formal process to determine which ontologies should and should not interoperate. The Policy-Driven Ontological Analysis Model (PDOAM) will resolve this problem.

CHAPTER 4

RESEARCH METHODOLOGY

The Dynamics of a Policy-Driven Ontology

Addressing the problem of identifying which ontological representation of a concept is more complete and comprehensive is a necessity, because there can be several representations of a same concept. Organizations have to know that systems are operating based on shared concepts that are clearly understood and accessible. There is also the danger that if not managed, the meaning of the concepts will drift from their original intended sense, as mentioned in Firat, Madnick, and Grosz (Firat et al. 2002). There is no benefit in having interoperable systems if concepts themselves are not consistent over time.

All organizations define enterprise-level policies independent of any implementation. Within these policies, concepts are defined operationally and explicitly (Bruijn 2003; Fensel 2001b; Firat et al. 2002; Gangemi et al. 2000; Gomez-Perez et al. 2004). For example, the concept of *customer* is defined by the organization as a critical element within a business model or an information standard. In addition, new regulatory expectations and audit processes are demanding to see these specifications explicitly declared as part of governance and due diligence practices (Tozer 1999). In effect, these policies define concepts that become data/information design requirements for the enterprise. It is important to remember that these specifications are independent of any database or software development implementation; they are considered to be operational and formal business models. These specifications also act as control mechanisms that are part of the enterprise quality assurance process for databases and enterprise-wide applications.

The integrity of a concept can be compromised through equivocation and ambiguity when two different ontologies appear to reference the same concept (Visser et al. 1997).

In reality, in this situation there could be two different concepts at play. For example, a concept called *customer* is found commonly across different applications. However, in one case the application using the concept of *customer* supports only internal services. A *customer* in this context represents an *employee* as users of an internal system. In another application the context for the concept of *customer* is one of a buyer of goods/services of the organizations itself and not an employee of the organization in question.

As an example, if these two different applications were to be integrated to support some governance or decision-making requirement and had to determine how many real customers the organization supports, it is certain that although interoperable, the new system would have irrevocably compromised the concept of *customer*. All responses to queries as well as all inferences using the corrupted dataset would not be a truthful or meaningful representation of the actual state of affairs. The availability of more and more enterprise-wide well defined critical concepts will further improve the ontology analysis process. These fundamental organizational concepts must be used in the context of interoperability assessment and can only be done by creating policy-based and organizationally-driven ontologies and further verified across many instances.

There has been debate as to which is the best approach for creating ontologies. In most cases some kind of meta-reference ontology is supported. This approach has been proposed by Ram and Park in their working paper (Ram et al. 2001) and by Davies, Green, Milton, and Rosemann, (Davies et al. 2003) but in each case the approach to the creation of what is called a common general domain ontology is based on a construction using merged domain ontologies. This approach is different from the one proposed here, as the Policy-Driven Ontology is generated from a top-down method and is an organizational dependent process rather than a technological one.

A stated Policy-Driven ontology acts as an enterprise benchmark of a concept. As these Policy Driven ontologies have a stronger organizational legitimacy they will help resolve situations where there is ambiguity or equivocation between two domain ontologies. In

effect, this new frame of reference permits an integration of both top-down and bottom-up analytical process into a single integrated analytical framework.

As mentioned, a Policy-Driven Ontology methodology permits the maintenance of the integrity of the concept over time (Zhao 2003). A Policy-Driven Ontology ensures that there is no diffusion or drift of the meaning and it maintains the overall structure of the concept over time and across implementations.

In addition, this solution addresses important efficacy considerations. In evaluating concepts for interoperability, the number of concepts that would have to be compared in pair-wise comparisons creates an exponential growth curve which makes it extremely difficult to scale. The use of a single Policy-Driven ontology reduces the comparative analysis workload to a linear growth, which makes the solution more computationally tractable.

Comparing Domain Ontologies and Policy-Driven Ontologies

As mentioned, one of the main benefit of constructing Policy-Driven Ontologies is to support of data interoperability evaluations (Firat et al. 2002; Obrst 2003). Other benefits would be increased quality of organizations and opportunities for inferencing. In our case, the objective of all interoperability initiatives is not just to ensure syntactical equivalence but to ensure semantic equivalence. By its construction, a Policy-Driven Ontology would be the most important ontology in the compared set. This assertion is based on the fact that a policy is based not just on systems requirements, but on external organizational requirements. Hence the establishment of a Policy-Driven Ontology sanctions an expectation of data interoperability (Visser et al. 1997) across systems at an ontological level (Firat et al. 2002). Naturally, there are cases where domain ontologies are more expressive than the Policy-Driven one. This situation could eventually result in the promotion of the most expressive one to the role of standard (Renner et al. 1996). This can also be done on an extended basis through the process of ontology revisions and versioning (Noy et al. 2000).

Proposed Policy-Driven Ontological Analysis Model

The proposed PDOAM model grounded in part in the research from Resnik on semantic similarity (Resnik 1998) and Visser research on ontological mismatch (Resnik 1998; Visser et al. 1997). In the proposed model, semantic equivalence is determined via an ontological analysis where domain ontologies are compared to standard Policy-Driven Ontologies.

In cases where there is semantic equivalence between a Policy-Driven Ontology and several Domain Ontologies this is an indicator that the domain ontologies are in fact subsets of the Policy-Driven Ontology. This also establishes some semantic equivalence between the domain ontologies themselves. With semantic equivalence determined between domain ontologies, there is a basis for asserting the existence of data interoperability between the domain applications that rely on those data sets. In the cases where domain ontologies are not mapped to a Policy-Driven Ontology and thus semantic equivalence cannot be established, then there cannot be any claim of interoperability between any domain ontologies on that concept or any claim of semantic interoperability between their data sets.

Value of the Model

The PDOAM reduces the cost and effort required to ensure data interoperability across multiple applications by identifying potential interoperability outcomes prior to integration efforts initiatives. The present risk associated with application development and integration is that most projects fail to recognize the danger that, although some syntactical level mapping could still take place, without some assurance of semantic equivalence the overall result would be data corruption. This event would compromise any expectation of integrity with regards to the validity of the resulting information sets. In turn, this will also invalidate any decision made based on these datasets (Fensel 2002; Fisher et al. 2003; Kalfoglou et al. 2003; Kim 2000; Obrst 2003). Demonstrating that there is a semantic overlap of the Domain Ontology into the Policy-Driven Ontology ensures that both are in the same universe of semantic discourse within a problem space. As long as Domain Ontologies are regulated by an alignment with a Policy-Driven

Ontology then they are within the same universe of discourse and have the same semantic scope. The semantic bridge between applications would mediate and bind domain ontologies. Such an overlap has to be present between various Domain Ontologies and the Policy-Driven Ontology to have any semantic interoperability.

Analytical Approach: Process and Structure

The research methodology for evaluating the Policy-Driven Ontology Analysis model will be to create a model where there is a requirement to assess whether or not interoperability is possible between known applications. The assessment is based on respective application datasets and their metadata. The model evaluates this approach by analyzing how well concepts extracted from domain applications and transformed into Domain Ontologies can be processed by the model to determine if there are elements that are semantically equivalent and interoperable.

As noted earlier, the PDOAM enables a combination of both a top-down and bottom-up concept reference model. The proposed model creates domain ontologies from organizational metadata assets. These ontologies are then used in a comparative analysis. On the basis of this comparison, an evaluation of interoperability issues is determined. As indicated, this is a mixed approach that compares application-level ontologies, which can be considered a bottom-up approach, with policy-derived ontologies, which can be considered a top down approach.

There are three discreet modeling transformations required:

1. The bottom-up modeling phase can uses ERD schemas extracted from the application's datasets. From these ERD schemas, concepts are identified. These concepts are then transformed into representative domain ontologies. For example, if there is a requirement for a sales and a billing application to be interoperable, then common concepts such as clients and sales would be found in each of the application and would be investigated to ensure that they have the same meaning.

2. The Policy-Driven ontology, because it is essentially a top-down modeling phase, is constructed by deriving from enterprise definitions of the same concepts as found in the domain ontologies. Policy-Driven Ontologies are created by using the enterprise wide operational definitions of concepts such as “client” or “customer”.
3. Finally within the PDOAM, the various domain ontologies are compared against the Policy-Driven ontology and evaluated for their interoperability potential. More specific detail is discussed in the following sections, but the overall assessment process looks at determining whether or not the domain ontologies are concept subsets of the Policy-Driven concept.

The generalized analysis model is illustrated below in Fig 3.

Figure 3- Generalized ontological analysis framework

Considerations for Dataset Level Ontology Modeling and Creation

In every application, there are representational schemas that support the software development process. These representations are essentially models of the various concepts. These descriptions can also be used to create ontologies. Both the Unified Modeling Language (UML) used for modeling entities and behaviours during the software development and the Entity Relation Diagrams (ERD) used to create database schemas can be leveraged to create ontologies, provided that certain transformation rules are respected (Fettke et al. 2003). Papers by and Loos (Fettke et al. 2003), and Shanks, Tansley, and Weber (Shanks et al. 2003) show that in both cases the metadata can be transformed into ontologies. This transformation from metadata to ontology is called a Bunge-Wang-Weber (BWW) transform (Shanks et al. 2003). This transform process means that we can now compare domain ontologies with each other, independent of their initial representation as a UML schema or an ERD (Rosemann et al. 2000). This transform process has also been discussed by Kashyap and Sheth (Kashyap et al. 1996; Kashyap et al. 2002).

It is also during this phase that an ontology normalization activity takes place (Aussenac-Gilles et al. 2000) (Gomez-Perez et al. 2004). Normalization includes two steps, a linguistic step and a conceptual modeling step.

During the linguistic step the analyst will review relevant terms and their lexical relationships to the domain. If the terms have several meanings then the relevant senses are taken to match with the domain. There needs to be a way to recognise and address to the following semantic conditions:

- **Polysemy:** This happens when two concepts which have overlapping or disjointed meanings have the same name. (Flight can be in relation to a specific plane or a scheduled trip). A different concept for each meaning of flight has to be created.
- **Homonymy:** This happens when two disjoint concepts have the same name. (Flight in relation to a plane or can be in relation to a threat). This is resolved by

using different names for each concepts or preventing the homonym to be included in the same namespace.

- Synonymy: This is the converse of homonymy, and happens when two concepts with different names have the same meaning. (Flight and trip).

During the conceptual step, terms and lexical relations are transformed into class and properties which are defined using the terms associated with the concepts and relations.

The result of normalization ensures that for any given concept, the following information is made explicit (Gomez-Perez et al. 2004):

- The concept must have at least one common attribute or relation which its parent concepts – generally an inherited attribute or relationship.
- The concept must have at least one specific attribute or relation that makes it different from the parent concept.
- The concept must have at least one property that makes it different from its sibling concepts.

Policy-Driven ontology modeling and creation

In this phase of the process, an enterprise-based concept definition is used to create a Policy-Driven ontology. This policy-based transformation phase is a top-down modeling exercise that also supports broader knowledge management activities such as versioning as indicated by Kim (Kim 2000). There are two main approaches to ontology creation:

- Top Down: The Top Down stage-based models focus on the conceptual modeling through theoretical research (philosophical, psychological, sociological, economic, political etc.), and articulate a formal schema. This

approach is appropriate when purposes and requirements of the ontology are well defined and structured through the use of well established vocabularies and taxonomies (law medicine for example)

- Bottom Up: The evolving prototype models are derived using automatic or semiautomatic extraction of categories, taxonomies, partonomies, and dependency graphs from a large number of instances. They are more useful when the environment is dynamic and difficult to understand.

Taking into consideration these aspects, all constructed ontologies are time-consuming, labour intensive and error-prone but they are necessary in defining a domain.

For example, a Policy-Driven ontology can become a design template or even a requirement specification for future enterprise-wide application development. Policy-Driven ontologies (being free from any implementation considerations) can act as reusable components for applications that take advantage of existing data and information already available in the organization. Eventually, the organization can use these Policy-Driven ontologies as building blocks for developing an enterprise-wide integrated information architecture. This architecture can be used to support the integration of datasets that support databases, web services, and corporate workflows (Osterwalder⁷ 2001; Schlenoff et al. 1999). In this context the use of policy-driven ontologies acts as the primary design activity that precedes any development or implementation effort (Daconta et al. 2003).

Comparing Domain ontologies and Policy-Driven Ontologies

During this step, the various domain ontologies that have been created using the metadata from the different databases are compared with the Policy-Driven ontology. Since the Policy-Driven ontology is considered the main reference model, a class analysis can determine if any entity in the domain ontology is also a subclass in the Policy-Driven Ontology. Work in this area has already been performed by Weinstein and Birmingham (Weinstein et al. 1999) focusing on how to compare concepts definitions across

ontologies. They describe how ontologies inherit specific properties and structures from these concept definitions. It is this unique structure of relations and properties that provides the means for identifying and analyzing correspondence between the elements of the different ontologies.

Constructing, Verifying and Validating the Model

Using a theoretical situation will permit a comparative analysis for evaluating organizational reference models and applications. In all testing situations, apriori knowledge of outcomes will be used to validate the overall model. In this first theoretical exercise, the ontological analysis of different data structures will lead to recommendations on semantic interoperability. The grounding for this claim is based on the capacity of the proposed model to evaluate concepts, terms and structure based on both the labels and their structural relationships to the various concepts that are part of the ontology (Noy et al. 2000).

It is important to understand the methodological approach and the validation process of the proposed model. To be valid, the model has to be both internally coherent and it must be able to explain an external real world reality (Creswell 1994; Lakatos et al. 1970; Popper 1989b).

Conditions for Asserting Interoperability

The main criteria for establishing the opportunity for interoperability is to determine if a domain ontology is a subset of the Policy-Driven ontology or not.

Considering the following:

- Concepts are defined by the structure of their properties and relations. They can be represented through ontologies in a way that facilitates the use of description logic for inference and

- Local concepts inherit definitional structures from more global and generic concepts. This implies that domain ontologies inherit structural similarities (and are subsets) of more generic and global concepts (e.g. they inherit context).

Therefore, based on these considerations, interoperability between two datasets is possible if the following propositions are satisfied at the ontological level:

- Proposition 1: If Domain Ontologies, when compared to the Policy-Driven ontology, show a similarity in structure at class and properties and relationship level then they are considered equivalent on that class.
- Proposition 2: If classes in Domain Ontologies are also equivalent to classes the Policy-Driven ontology then the domain ontologies and the Policy-Driven ontology are interoperable on that class, through their common context.
- Proposition 3: If two Domains Ontologies have a common class that is also a subset of a Policy-Driven Ontology then these classes in both Domain Ontologies are interoperable on that class, through their common context in that ontology.

Using these propositions, interoperability can be asserted over Domain Ontologies on specific classes across a variety of situations. Another way of presenting these propositions for evaluating interoperability between datasets would be to look at the concept (i.e. customer) when it is represented in the various domain ontologies with consideration to its specific attributes, properties and relationships (its context), and identify which one of these are the same as those found in a Policy-Driven Ontology. In these cases, the applications are interoperable across those datasets with a confidence interval based on the degree of similarity of their context.

Those considerations also mean the converse, that datasets of a domain ontology which is not a subset of the Policy-Driven ontology are not interoperable with other data sets within the scope of the ontology. The issue then is to look for root causes and to

determine why and where the discrepancies lie. This analysis process can be supported by an ontological engineering environment.

Using Apriori Knowledge to Empirically Validate the Model

By conducting a Policy-Driven Ontological Analysis, interoperability between applications can be evaluated and assessed. To empirically validate the model, the approach taken is to reuse sets of data sets and concepts that have resulted in identified types of (desirable) interoperability outcomes. These sets of outcomes describe the apriori knowledge of interoperability to which we can compare the results of the model. In this way, the model can be validated and confirmed as being empirically adequate (Psillos 1999) as the model can be said to explain “adequately” the observable data. This is a pragmatic approach for validating models within the context of constructive empiricism as demonstrated by Van Fraassen (Van Fraassen et al. 1985). By proving that the model works on apriori outcomes, future data interoperability opportunities can be adequately predicted by the model when working on projects initiatives that require application integration. Overall all models have to be able to show how they meet the requirements for internal coherence and external correspondence (Popper 1968; Popper 1989a).

Testing for Model Coherence through a theoretical problem

Validation of the model using a theoretical problem is done in several steps. Initially a small theoretical problem is used to demonstrate how the model operates in theory. Then a constructed theoretical comparative analysis of two ontologies is done to show how the model operates in principle and in practice. During this evaluation phase the focus is on the model as a process and how it ensures that the outcomes are explicitly predetermined by the structure of the problem set. In this phase, the requirement for internal coherence, which is an essential criterion of any model validation, is addressed.

Testing for Model Correspondence Using a Real World Problem

By using observed data sets for which there is a priori knowledge of data interoperability outcomes, the model reproduces known outcomes and derives the same conclusions as

real world outcomes. This ability ensures that the model meets the criteria for external correspondence.

This criterion can, in fact, be a difficult one to meet as there can be a multitude of other real world factors involved that can lead to the same outcomes as the model but which that are not part of the model. Suffice to say that in this case the ability of the model to recreate known outcomes does not prove that the model works in all situations but rather that it is able to show equivalent results.

If in evaluating the model from both criteria - of internal coherence and an external correspondence, the results are comparative to the ones expected, then one can say that the model is valid and empirically adequate (Van Fraassen 2002). This approach offers a way to compare the results of the Policy-Driven ontological analysis model against exiting interoperability initiatives with their specific outcomes from both a theoretical and a practical perspective.

Ontological Engineering Environments

The development and analysis of ontologies can be done in one of the many ontological engineering environments. At present the main four environments they are: WebOde from the Ontology Group at the Universidad Politecnica de Madrid (UPM), OntoEdit of the Institute AIFB at the University of Karlsruhe (now being commercialized by Ontoprise GmbH), KAON (Karlsruhe Ontology) tool suite is being developed by the institutes AIFB and FZI at the University of Karlsruhe, and Protégé, being Developed by the Stanford Medical Informatics (SMI) group at Stanford University. An in depth review and comparison of ontology mapping and merging environments can be found in reviews by Gomez-Perez, Fernandez--Lopez, and Corcho (Gomez-Perez et al. 2004; Noy et al. 2002b) and Noy and Musen (Noy et al. 2002b).

They classify tools that deal with finding relationship and correspondence between two ontologies in the following way

- Tools for merging ontologies to create new ones such as Protégé, Chimera, and Onto Merge.

- Tools for defining a transformation function from one ontology to another such as Ontomorph
- Tools for mapping between concepts in two ontologies by finding pairs or related concepts such as protégé. Glue, Observer, FCAmerge
- Tools for defining mapping rules to relate only relevant parts of the source ontologies such as Protégé.

Ontology Analysis Environment: Protégé

The analysis phase of the research was done using Protégé. Protégé (Noy et al. 2002a), is a ontology editor as well as a knowledge-base editor. Protégé is also an open-source, Java based tool that provides an extensible architecture for the creation of customized knowledge-based applications. It has become a de-facto industry standard for supporting ontologies that are written in Ontology Web Language (OWL). Protégé is the ontological engineering environment of choice and is generally used to do the following:

- Class modeling – Protégé provides a graphical user interface that models class (the domain concepts) and their attributes and relationships.
- Instance editing. From the classes, Protégé automatically generates interactive forms that enable the domain experts to enter valid instances.
- Model processing. Protégé has a library of plug-ins that help defining semantics, ask queries and define local behaviour. Specifically it has several plug-ins that support ontology comparisons and evaluations.
- Model exchange. The resulting models (classes and instances) can be loaded and saved in various formats, including, OWL, XML, UML, and RDF (Resource Description Framework). Protégé also provides a very scalable back end database.

Most knowledge engineering environments are structured like a classical database system, except they differentiate the content from the schema. In the case of ontologies Protégé, defines classes of information (i.e. the schema) and stored instances of these

classes in the same environment. This makes it an integrated environment and permits users to work with both classes and instances. This means that specific instances can be used on various abstraction levels such as in a top/upper level ontology where a class can be stored as an instance. Similarly, slots, which earlier were employed only within classes, can now be elevated to the same level as classes. Within this knowledge modeling tool there is also the provision to use open knowledge database protocol for accessing knowledge bases stored in knowledge representation systems.

Ontologies can then be analyzed if there is an underlying Description Logic built into the ontology via a theorem prover capability such as RACER developed by Haarslev and Möller (Haarslev et al. 2001) of Concordia University. This inferencing capability can also permit ontologies to be used for even more possibilities not just for resolving semantic interoperability but also organizational sense making and knowledge inferencing.

Comparing Ontologies

The causes that lead to non-interoperability, such as concept incompatibilities need to be categorized. Kashyap and Shep (Kashyap et al. 2002) have shown that there are three types of incompatibilities: domain definition incompatibilities, entity definition incompatibilities, and schematic incompatibilities. Another more general categorization is offered by Visser, Jones, Bench-Capon, and Shave (Visser et al. 1997) that showed two broad categories of mismatch.

- Conceptualization mismatches that result from differences in the categories or data structures used and
- Explication mismatches that result from the definitions used, which may differ in their terminology, their formulae, or the concepts which they are defining.

Hameed, Sleeman, and Preece (Hameed et al. 2002) shows that there exists an “ontology of ontology mismatches” that acts as a list of root causes but, because his work is still in progress and has yet to be accepted by the community, there is a risk associated with using with this referent model. In lieu of this, we will adopt the Gomez-Perez and

Eriksson et al (Eriksson et al. 1999; Gomez-Perez et al. 2004) classification of the causes for failure of interoperability as one of:

- Incomplete concept. This occurs when concepts are classified without accounting for all essential aspects of the concepts.
- Semantic error. These occur when the ontologist makes an incorrect semantic classification, that is “s/he classifies a concept as a subclass of a concept to which it does not really belong: for example the concept of plane as a subclass of the concept of house”.

Once the incompatibilities between the Policy-Driven ontologies and the Domain ontologies have been identified then organizations can start addressing specific barriers to interoperability.

However, the processes and issues of removing barriers to interoperability are out of scope for this research. Those activities would bring us closer to what is being touted as the ontological re-engineering of an organization (Obrst et al. 2001). As this area is still novel and the diffusion of ontological engineering practices are barely emerging within organization there are no assured conclusions on this subject. Findings from this research will help in the exploration of this problem.

Comparing Ontologies within Protégé: Using Prompt

To be able to execute ontology comparison, an ontological engineering environment is required. Protégé will be used as the construction and analytical engineering support environment for the research and will act as the analytical framework in which the comparison of ontologies will be done. It is not the only one; there are a multitude of ontological engineering environments that would support these activities as outlined by Denny (Denny 2004). Prompt supports not just the construction of ontologies also but has some analytical capabilities such as versioning analysis, which the others do not have.

The Prompt plug-in within Protégé is a ontology management capability that will be used to support the comparative analysis of ontologies. The support functions of Prompt deal with the merging and versioning of ontologies as well as the evaluation of overall class alignment (Noy et al. 2002b). Prompt will be used in our context to highlight differences and similarities between two ontologies.

This capability will be leveraged during the comparative analysis processes between the Policy-Driven Ontologies and the Domain Ontologies. Initially indented for versioning, there is an opportunity for use in comparing “anticipated” similar ontologies as different versions of a main ontology. This is a stretch for the objective of Prompt but for the purpose of the research gives us the support required for a comparative analysis.

This use of Prompt is an extension of its initial intention for comparing different versions of a same ontology. However, since the object of the research is the creation and evaluation of a model for assessing data interoperability using an ontological analysis framework, the use of Prompt to compare potentially different ontologies is one of the limitations that need to be taken into consideration. The Protégé environment with Prompt is sufficiently sophisticated to demonstrate that the model works. It is expected that more sophisticated ontological engineering environments will be created in the coming years that will leverage the model in a more specific ways.

Here, in Figure 4, is an illustration of a generic ontology within the Protégé environment. Figure 5 shows an example of the results from comparing two ontologies. This illustrates the similarity and differences with regards to properties of two different ontologies.

Figure 4- Visualization of a portion of a Policy-Driven Customer Ontology in Protégé

Figure 5 – Example of Comparing Policy-Driven Ontology and a Domain Ontology

Protégé has a legend for reading the different annotation of a concept. The following (Fig 6) is the nomenclature addressing this information within the environment

Figure 6- Legend

Comparing Two Ontologies using Prompt

Prompt allows the comparison of two different versions of the same ontology. The heuristics behind the standard Prompt merge have not been modified in this research. Work by Noy and Musen (Noy et al. 2000; Noy et al. 2003) outlines the inner-working of the PROMPT framework. When two projects have different versions of the same ontology, Protégé can use strong heuristics, suggesting, for example, that a single unmatched sibling of the same parent in the two different versions may be the same class but with a different name. By concentrating on the differences between two similar projects, the comparison can give better results for the type of reconciliation required for simple version control.

Prompt was originally designed as an interactive ontology-merging tool. It guides an analysis of the merge of two ontologies by stepping through a potential integration process while making suggestions, determining conflicts, and proposing conflict-resolution strategies. The suggestions are based first on the linguistic similarity of the frame names. After selection of an operation to establish equivalence of terms, Prompt then determines which conflicts in the merged ontology the last operation may have caused and proposes possible solutions to the conflicts. It then proposes other options, considering the resulting class structure of the ontology. The comparative analysis that goes on in Prompt mirrors the requirements of our ontological assessment framework that

will be elaborated in the next sections where terms and structures are compared to determine the appropriate outcome of an ontology comparison.

In the context of the PDOAM, Prompt is used to solely support the comparative analysis of ontologies. In fact there are already a number of ontology engineering environments and tools that can supported this ontological analysis. Michael Denny's Ontology Tool Comparative Analysis table (Denny 2004) indicates that 12% of all the present ontology tools have some kind of support for ontology alignment and data resource integration. But none of the existing environments supports the PDOAM completely. Prompt is the closes one.

In comparison with other systems such as GLUE, ONION, Chimera, OBSERVER and FCA-Merge (Noy et al. 2002b) Prompt is still considered the best when in the absence of a large number of instances when clustering. Machine learning algorithms can also be applied to supplement the semantic mapping processes in Prompt.

Setting Up the PDOAM Model

In this next section, there will be an explanation of how the modeling process is used in practice, in a planned set of trials. One set will be using theoretically constructed data and the next set will be using organizationally observed data. The intention is to use organizational data to show the methodology and not to analyse any specific organizational findings. The metadata for all the datasets used for analysis will come from within the context of a large financial organization.

Our approach will leverage findings by Weinstein and Birmingham (Weinstein et al. 1999) that compared concepts in differentiated ontologies. Their research showed that the semantic compatibility of two concepts is a function of the intersection of their denotation. This work was further demonstrated by Firat, Madnick, and Grosz (Firat et al. 2002) on financial systems as well as by Schlenoff, Ivester, Libes, Denno, and Szykman (Schlenoff et al. 1999) in healthcare information systems. However, neither of them used a Policy-Driven ontological analysis model as presented here.

The proposed modeling approach is validated through a pair-wise comparative analysis between the domain ontologies and the Policy-Driven ontology, and there are three main phases:

- In the first phase, using the Policy-Driven Ontological Analysis Model, Domain Ontologies are compared with the PDO as the comparative framework.
- The second phase will be a review of the outcome of this comparison based on the alignment and the interoperability criteria. This will be presented in the next chapter.
- In the third phase, a review and discussion of the results and findings based on the Policy-Driven ontological analysis processes is presented. This will be done in the subsequent chapter.

As indicated, datasets from a large financial institution will be used to verify the Policy-Driven Ontological analysis model. Their metadata repositories contain all major business-critical applications as well as ERD's on most of the applications. Concepts that are common across multiple datasets, such as the concept of "Customer" or "Employee", can be identified and extracted from the metadata repositories and used to create the relevant domain ontology. This activity is done following the transformational requirements for ontology construction (Guizzardi et al. 2002; Moody et al. 2003; Shanks et al. 2003).

The identified concepts are also retrieved from various enterprise-wide organizational policy documents and used to create Policy-Driven Ontologies. These ontologies are used as the reference ontologies for the comparative analysis process. For example, in this specific organization, a unified and generic enterprise-wide concept of customer has been developed. This specific policy-grounded definition has 29 different classes that range from the name and address to various types of accounts that a person could have. Each

class has several attributes. This policy is an ideal candidate for transformation into a Policy-Driven Ontology. Such policies that define a concept with all properties and attributes described can also be used as requirements for an information architecture. For example, the listing of the customer concept top classes for the ontology is shown in Figure 7.

Figure 7 - Policy defined definition of the concept of customer

The generic transformations are as follows:

1. Identify a list of target applications and their respective metadata and dataset repositories. A review of the different projects helped to identify the series of applications/information and data repositories that are required to become interoperable. A review of these applications/information and data repositories and

their metadata is done to determine if there sufficient metadata records are available to develop domain ontologies.

2. Determine the main key concepts are common to the various applications. For example concepts such as client, customers, address, account, revenue etc. are common.
3. Perform the impact analysis for potential application interoperability problems if they are not resolved (for example what would happen if there were two application based on the misidentified key concepts such as “customer”). List and describe some of the initial interoperability issues.
4. Conduct a pair-wise review on the concepts in the domain, based on transformed domain ontologies.
5. Create key Policy-Driven Ontologies based on enterprise-wide data definitions on domain concepts such as those shown in the definition of customer ;
6. Apply the Policy-Driven Ontology analysis model, comparing the Domain Ontologies to the Policy-Driven Ontology with regards for their semantic compatibility. Classify the results into one of three categories
 - Matched – class of the domain ontology can be subsumed into the archetype policy based ontology.
 - Partial match in which only certain classes and attributes of a Domain Ontology is found in the Policy-Driven ontology.
 - Non-correspondence in which there are no corresponding classes or structures found in common between the domain and the Policy-Driven Ontology.
7. Analyze and discuss interoperability findings, resulting from each case. Review the effectiveness of the Policy-Driven ontological analysis model.
8. End the review

As declared earlier, the use of an ontological engineering environment provides some support for the creation and management of the various ontologies, and can be presented in various formats such as OWL, RDF, tables or in graphical networks form. This environment will also support ontology lifecycle management requirements such as versioning. Although not discussed here, one of the potential outcomes of matching ontologies can be also merging them, which would be done when two ontologies need to be integrated into a single, more complex ontology – either Policy-Driven or Domain ontologies.

Datasets structured into metadata through the use of modeling languages such as UML and ERD can be used within the ontological engineering environment to create the various domain ontologies. Ontological engineering environments also provide mechanisms to analyse issues of integration, merging, sharing, and reasoning across multiple ontologies.

Recent developments in ontological engineering practices have influenced the way researchers look at an organization's representational problems in information systems. It is clear that this venue also offers the PDOAM approach to help manage enterprise needs for more general information-sharing processes at the application level.

Ontology Mismatches: Generic Issues.

A review of the state of ontology mapping tools (Kalfoglou et al. 2003) notes that every work that focuses on ontology mapping has failed producing a fully automated method for ontology mapping. They quote Sheth and Larson (Sheth et al. 1990) on the following:

Complete automation is not feasible as we need more information than currently provided by the database schemata, the semantics of the data models do not adequately capture the real world and the absence of structural similarity between schemata or absence of instance data in target applications makes their automatic matching or integration difficult.

However, the state of the practice in ontology modeling has recently seen the development of semi-automated support for ontology mapping tools such as Prompt for the Protégé environment (Noy et al. 2000). Prompt uses the hierarchical structures of classes and properties combined with the network of meaning derived from the class labels for initiating its merging or alignment processes. Prompt then uses the underlying ontological class structures (class, attributes and properties) to determine similarity.

To classify the outcomes of the assessment of the mapping between ontologies, it is necessary to enumerate the main outcomes, which determines interoperability opportunities. As stated earlier, Visser, Jones, Bench-Capon, and Shave (Visser et al. 1997) will be the main classification framework, since they identify the main reasons for ontological mismatches. The analysis table will include an evaluation of terms and class definitions.

An alignment criteria table must to be constructed to classify the various cases of matches and mismatches. Exploration and comparative analysis of these overall results is a core part of the research findings.

In each ontology alignment evaluation, the set of Terms and structures of the Class Definitions will be compared. Policy-Directed Ontologies Class Definitions will be compared to Domain Ontologies class Definitions. In addition, Policy-Driven Ontologies Terms will be compared to Domain Ontologies Terms.

The resulting decision table is based on a synthesis of Visser (Visser et al. 1997), Gomez-Perez (Gomez-Perez et al. 2004) and Kavouras in (Kavouras 2003; Kavouras et al. 2000). This decision table combines all required for a term and definition comparison. This process was discussed in Garigue (Garigue 2004). The table is presented in the next section.

Semantic Matches and Mismatches: Determining Alignment

A set of terms and concepts defined as class structures (using hierarchy of entities as class structure) are compared. Policy-Driven Ontologies' class structures, $Pdo(S_n)$, are compared to Domain Ontologies' class structures $DO(S_n)$ and Policy-Driven Ontologies' Terms $Pdo(t_n)$ are compared to Domain Ontologies Terms- $DO(t_n)$. The following table presents the outcome of the comparison of a Domain Ontology (DO) against a Policy-Driven Ontology (PdO). This table determines the alignment between a Domain Ontology and the Policy-Driven Ontology.

Ontology as composed of Class Structures (S_n) + Terms (T)	$PdO(T_n) = DO(T_n)$	$PdO(T_n) \neq DO(T_n)$
$Pdo(S_n) = DO(S_n)$	Equivalent	Equivalent and Synonymous
$Pdo(S_n) > DO(S_n)$	Subset	Synonymous Subset
$Pdo(S_n) \cap DO(S_n)$	Overlap	Synonymous Overlap
$Pdo(S_n) \neq DO(S_n)$	Homonymy	Disjointed

Table 1- Ontological Alignment Assessment Framework

Each one of these outcomes will now be presented graphically for further detail.

Equivalent:

The resulting mapping between the two ontologies – the Policy-Driven Ontology and the Domain Ontology - is isomorphic: The same terms and class structures are used in both ontologies. This situation demonstrates complete interoperability as both applications are in fact using the same ontology. There is semantic equivalence in the form of an equivalent structure of relationships between the entities of the concept as well as syntactical equivalence at the terminological level.

Figure 8 - Equivalent Ontologies

Equivalent and Synonymous:

Overall there is still isomorphic mapping between the two ontologies as the class structures are the same but the terms are different. There are semantic relations that hold true between two synonymous words because they express the same meaning. This situation still permits data interoperability but it requires an intermediate shared vocabulary (shared vocabulary mediation) to help resolve differences in terms.

Figure - 9 Equivalent and synonymous

Subset:

The Policy-Driven Ontology is broader in that it covers more elements than the Domain Ontology. Essentially the DO is a subset of the PdO. Thus, the Policy-Driven Ontology extends the Domain Ontology - more than required by the domain application. There is a limit to the semantic alignment between these two ontologies. In this case, the terms and structure of the classes are the same for only a portion of the PdO.

Figure - 10 – Subset

Synonymous Subset:

This situation is similar to the Subset case but the terms are different for certain of the entities across the two ontologies. The terms are still synonymous and therefore still ensure same semantic alignment

Figure 11 - Synonymous Subset

Overlap:

The Domain Ontology has some overlap with the Policy-Driven Ontology but there are concepts (terms and definitions) that are unknown within the scope of the Policy-Driven Ontology and within the Domain Ontology. These concepts outside the Policy-Driven Ontology but within the scope of the Domain Ontology and are not considered as part of the overlap. In this case the terms used are the same.

Figure 12 - Overlap

Synonymous Overlap

The Domain Ontology has some overlap with the Policy-Driven Ontology but there are concepts (terms and definitions) that are unknown with the scope of the Policy-Driven Ontology as well as within the Domain Ontology. These concepts outside the Policy-Driven Ontology but within the domain ontologies are not considered as part of the overlap. Some of the terms are synonymous.

Figure 13 - Synonymous overlap

Homonymy:

The class definitions are different but the terms used are the same. In this context, the term is the same across both ontologies but it has different meanings. There could be a belief, in this situation, that there are some interoperability opportunities because of the use of same terms. In actual fact, there is no semantic alignment between the two ontologies.

Figure 14 - Homonymy

Disjointedness:

The two ontologies are disjointed since they use different terms and have different class structures. This situation occurs when they have nothing in common, even though both claim to represent the same concept.

Figure 15 - Disjointed

Determining Interoperability

Now that we have established the list of all possible outcomes that result from a comparison of two ontologies, interoperability outcomes can be developed, mapped and enumerated via an overall evaluation and decision table. By looking at the alignment between the two ontologies, it is possible to determine if the ontologies are fully interoperable, partially interoperable or non-interoperable. The outcome resulting from the analysis of matches and mismatches between a Domain Ontology and a Policy-Driven Ontology can be presented by the following table which shows the conditions for opportunities of interoperability, partial interoperability and non-interoperability between two data sets.

The following interoperability outcome (Table 2) is used to classify the various of matches and mismatches and potential interoperability outcomes. Exploration and analysis of how this table supports the model is part of the observation section of this report.

Comparison	Outcome
Equivalence Equivalent and synonymous	Interoperability
Subset Synonymous subset Overlap Synonymous Overlap	Limited Interoperability
Homonymy Disjointedness	Non interoperability

Table 2- Data Interoperability as Outcomes of Ontology Alignment Analysis

From Ontology Alignment to Ontology Interoperability

The next phase presents how final interoperability between domain data sets is achieved. The PDOAM now considers domain ontologies in the context of a Policy-Driven Ontology and resolves the question of their potential interoperability across two domain ontologies.

This phase of the model is broken into two processes:

1. The first process reviews the alignment issues between each of the domain ontologies against the Policy-Driven ontology, and then
2. The second process looks at how the domain ontologies individual alignments are matched up against one another.

The following figure shows the two step process

Figure 16 – Phases in Comparing two Domain Ontologies

The process to determine interoperability between two domain ontologies requires a set of Domain Ontologies with classes and terms that are compared to establish if there are any part that aligns against the Policy-Driven Ontology.

The following interoperability outcome table is derived form the alignment table that shows how the combination of Domain Ontologies interacts to determine domain level interoperability. Theoretically, there are 16 outcomes possible for each set of domain ontologies. It is possible to collapse them into a simple 4*4 table of outcome, in which the 16 combinations produce one of three outcomes.

They are further reduced into three general rules:

1. "Disjoint" entails no potential interoperability – therefore Non Interoperable
2. "Subset" or "Overlap" entails a situation of Limited Interoperability
3. Fully Interoperable domain ontologies only occurs when both domain ontologies are "Equivalent" or "Equivalent and Synonymous" to the Policy Driven Ontology

Alignment	Equivalent	Subset	Overlap	Disjoint
Equivalent	Interoperable	Limited Interoperable	Limited Interoperable	Non Interoperable
Subset	Limited Interoperable	Limited Interoperable	Limited Interoperable	Non Interoperable
Overlap	Limited Interoperable	Limited Interoperable	Limited Interoperable	Non Interoperable
Disjoint	Non Interoperable	Non Interoperable	Non Interoperable	Non Interoperable

Table 3- Interoperability decision criteria between Two Domain Ontologies

CHAPTER 5

TRIALS AND ASSESSMENT

Empirical Comparative Analysis Framework

To empirically validate the PDOAM model as represented in figure 17, a comparative analysis and review is done using controlled as well as observed datasets taken from organizational system integration projects. A review and investigation of the metadata repository that holds all the representations of the various applications will permit the identification of recurrent concepts, terms, and definitions that are used across databases. The projects chosen all have requirements for dataset interoperability. As major projects have been completed, their outcomes as well as the data elements were included in the corporate metadata repository for analysis in support of this research.

Figure 17- Policy-Driven Ontological Analysis Model

The generalized Policy-Driven Ontological Analysis Model is shown in figure 16 and is characterized by the following phases.

1. Identification of common core concepts.
2. Creation of Domain Ontologies for those concepts.
3. Creation of Policy-Driven Ontology for a core concept.
4. Evaluating the alignment between Domain ontology and the reference Policy-Driven ontology using the Alignment Criteria table.
5. Determination of data interoperability between domain ontologies using interoperability criteria table.

Ontologies comparisons are done using the Alignment and Interoperability criteria as per previous tables 3 and 4 found in the previous chapter. Ontologies created from data identified through the metadata repository about the various application interoperability projects will provide the various sets of ontologies required for the comparative analysis process as described in the section on the generalized ontological analysis framework. The research will use the Policy-Driven Ontology principal for the concept of “customer” since this includes a wide range of personal information such as address, age, and sex as well as organizational information such as “accounts” and “branch”.

The PDOAM is considered valid if the results of the comparative analysis between the Policy-Driven Ontological Analysis and Domain Ontologies, done in accordance with the assessment framework, are equivalent to the known results of the trials within the context of the tests using theoretical data, as well as with real-world oriented organizational data. In the context of the theoretical test data, sets will be constructed to demonstrate not just how the model works but also that the model generates and predicts, the same expected results.

With regards to the organizational tests, the results and the datasets from past data integration projects are used as input into the Policy-Driven Ontological Analysis

framework. The interoperability outcomes from the Policy-Driven model will be compared with the real world outcomes. The real world outcomes act as validation outcomes for the Policy-Driven Ontological Analysis framework.

As the outcome of constructed and real-world cases are known in apriori way, the results of the Policy-Driven Ontological Analysis will produce the same results. As presented earlier, this approach will use a set of criteria for the model derived from Visser, Jones, Bench-Capon, and Shave (Visser et al. 1997) and extended by the creation of the Ontology Alignment and Ontology Interoperability decision criteria tables.

In evaluating the model there are two outcomes applicable in both sets of tests – with the theoretical, as well as the organizational data. The model compares known outcomes using theoretical and organizational data sets for which the outcome are a priori known

- The model has the same outcomes as predicted using both the theoretical and organizational collected data sets: therefore model is valid
- The model has outcomes other than the ones predicted using both the theoretical and organizational collected data sets: therefore model is not valid

Analysis of the data is done by the model using the theoretical and organizational data to determine interoperability. Interoperability outcomes can be as follows:

- Dataset interoperability test outcomes indicate interoperability. The results of the Policy-Driven Ontology Analysis Model generate results indicate that interoperability is possible. The two domain ontologies are both sets or subsets of the Policy-Driven Ontology
- Dataset interoperability test outcomes indicate non-interoperability. The analysis of the data sets indicates that there exist a state of disjointedness between the domain ontologies and the Policy-Driven ontology.

Comparison between known and expected outcomes across multiple tests and projects, for valid model, should generate identical outcomes as the postulated or observed outcomes. The outcomes are to be classified in the accordance with the Assessment Framework Outcomes. The claim that the Policy-Driven Ontological Analysis Model provides empirically sound and valid analysis model, is based on the fact that the model produces the same outcomes as the postulated and observed outcomes.

In order to address the possible influence of context variables such as extra personnel and different levels of financial resources available, with regards to organizational context, all datasets are selected from within one organization. In this way, variability in external factors that could have an influence on projects (such as the use of different development methodologies and different data naming conventions) are reduced. Selecting datasets from within one organization, ensures that the datasets are subject to the same organizational policies and generally within the same working software development methodologies. The selection and description of the data sets in this research was done in a manner to present the data without disclosing any business in confidence information as such contextual information with regards the structure or the importance and criticality of the application to the organization.

Two sets of tests will be performed using the PDOAM: one set of test will be using theoretical data and another set will be using actual, known data. In both situations apriori outcomes are known. An assessment of the model will be based on the final comparison between the known outcomes and generated PDOAM outcomes.

Testing the Policy-Driven Ontological Analysis Model

Expressing the PDOAM with a Theoretical Example

In this phase, a theoretical example will be exercised using two very small general ontologies. They will be evaluated for their potential interoperability. They are compared to one another to demonstrate several dimensions of the alignment criteria table. This theoretically constructed example demonstrates in practice how two ontologies can be compared for terms and class structure. In this theoretical example the results are known a priori in order to demonstrate how the criteria for terms and class structure are applied to determine interoperability.

This first example exercises the model in a comparison of two ontologies. They are non-declared ontologies in term of being either a policy driven ontology or either a domain ontology. In the second part of the example, one ontology will be promoted to the role of a Policy Ontology over the other and this last one will take on the role of a Domain Ontology.

The declaration of a PDO role will changes the semantic context of the analysis and resolves any inherent ontological ambiguity as well as helps in the determination of interoperability.

As first set of data are constructed for the theoretical exercise, the ontologies are structured to show an outcome of limited interoperability. Properties of classes are not required for this demonstration. Operating at a class level is sufficient to validate the model. The present theoretical example presents the most difficult case since extreme examples illustrate more powerfully the model.

Description of Model Concepts and Test Challenges

This first PDOAM test will establish if there are overlaps or subset classes between the two ontologies that have some common conceptual structural elements in the presence of

semantic ambiguity. In practice the theoretical example will compare the concept of Employee with the concept of Customer. These next sections outline the terms and class structures for the two ontologies' and will demonstrate how ontologies are constructed.

Both Employee and Customer are core concepts in all business models. They are fundamental to almost all information systems (Fensel 2002; Geerts et al. 2002; Kim 2000; Obrst et al. 2001; Schlenoff et al. 1999; Zhao 2003). However, there is a fair amount of ambiguity that can arise if it is not clear where the boundaries of those concepts are. In some systems, people paid by the organization, who act as employees are also considered "customers" of internal financial systems. In other cases, customers could be employees that have additional privileges. In many respects it is not about a concept itself but how a concept is related to other concepts within the system being described that will determine its ontological essence. Therefore at a data level, each concept has a unique internal and external structure that determines the boundaries by which a conceptual entity is defined.

In this theoretical example, a deliberate effort is made to create ambiguity at the terminological as well as at the structure level (via the classes) in order to see how the PDOAM resolves this problem and how it picks up on the obfuscation attempt. For example, to create terminological ambiguity the term "Last_Name" used as label for the class Name in one ontology but "Surname" is used as a label in the other. In addition, the term Customer_entity is used to refer to the Employee. This is both a semantic and syntactical obfuscation and a change of case across two labels. In both case the term customer acts as the highest class entity and is placed there as a way to see how the PDOAM process resolves a potential case of false equivalence.

Constructing an Ontology of the Concept of Employee

The following ERD was used to derive the Employee Ontology. Not all the elements of the ERD are used to construct the ontology but a sufficient amount of concepts were mapped to classes to create the Employee ontology.

Figure 18- Employee ERD

An individual's name can be the same as another employee but adding more and more information will reduce the ambiguity and ensures unique instance specificity. Generally employees are part of an organization and the organization specifies which employee is in question by creating unique identifiers such as an employee number which in turn is associated with a set of roles, rights and privileges that legitimizes an individual access to internal services and resources. With that access the individual can participate in the activities of the organization. As indicated, even though this is an ontology of an employee, the highest class in this class hierarchy is defined as a Customer_entity which is a way to create ambiguity of role and a concept misalignment.

This is a classic example (Vogt 2002) where the term is used to refer to a specific system that defines a specialized relation. It defines employees as simply being "customers" of the system but not necessarily a customer to larger information systems such as the bank itself. In effect, this application, because of its limited scope, has misappropriated the term and, as a result, created a situation of semantic ambiguity through different contexts (local vs global).

In this case there needs to be a way through the analysis process that would indicate a semantic misalignment and a homonymy issue at play between the two terms across these two ontologies. This ambiguity of context is in the following ontology with the overall ontology, Customer_entity, constructed from classes that define employee attributes.

The mark-up of this ontology is defined as follows:

```
<rdf:RDF
  </owl:Ontology>
  <owl:Class rdf:ID="Full_name">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Name"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Address">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Customer_entity"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Middle_name">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Name"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Name">
    <rdfs:subClassOf rdf:resource="#Customer_entity"/>
  </owl:Class>
  <owl:Class rdf:ID="Start_date">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Employee_number"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Id_number">
    <rdfs:subClassOf rdf:resource="#Customer_entity"/>
  </owl:Class>
  <owl:Class rdf:ID="First_Name">
    <rdfs:subClassOf rdf:resource="#Name"/>
  </owl:Class>
  <owl:Class rdf:about="#Employee_number">
    <rdfs:subClassOf rdf:resource="#Customer_entity"/>
  </owl:Class>
</rdf:RDF>
```

A more comprehensible way to present this ontology is via the protégé ontological engineering environment. This will be the normal presentation format with the full code of the ontologies in the various annex at the end.

Figure 19- Employee Ontology

Constructing an Ontology of the Concept of Customer

In an organization such as a bank, a customer is situated outside the legal boundary of a firm. The concept of customer has associated with it a set of classes that identifies it as being an individual that has unique set of records of transactions. In this case the following figure -20 would be an ERD that would represent the ontology of a customer with its accounts.

Figure 20- ERD of Customer

In the case of a financial institution a customer ontology describes associate concept of assets as well as the concepts relating to transactional operations that change the status of those assets. In this illustration in figure -21 of a customer ontology derived from the EDR there is a class hierarchy that relates the information about an individual to information that show how the assets are structured.

Figure 21- Customer Ontology

As discussed earlier the alignment analysis of the PDOAM will be done using the Prompt Plug-In for Protégé. It will be used in a non-standard way. The Prompt functions are

optimized for versioning and working with one ontology at a time but in various states of evolution. Prompt therefore looks for similarities and differences in a versioning process both structural and linguistic but based on the assumption that is the same ontology but in a different state of development. In our approach it is assumed that when comparing ontologies for interoperability there is an underlying expectation that there will be common elements either in the class structure or in terms used. The Fig 22 show how Employee and Customer ontologies are presented within the Prompt versioning window.

Figure 22 - Comparative analysis of the two ontologies

The two special cases are when ontologies are compared are found to be either Equivalent or Disjointed. In a case of Equivalence, the Prompt algorithm would show no difference as the two ontologies are the same. This would mean that the versions are the same and that there has been no change in the second ontology. In the second case of Disjoint, the ontologies are now so totally different that they have no elements in common.

In looking for the more difficult situation that are based on other than Disjoint or Equivalence Prompt will identify which classes have changed and which classes are the same so as to identify cases of either subset or overlap.

As seen in the next figure 23 Prompt has now performed a mapping and determined an alignment between the two ontologies.

Table view **Tree view**

Image table

f1	f2	renamed	operation	map level	rename explanation
	● Employee_number	No	Add		<null>
	● Id_number	No	Add		<null>
	● Start_date	No	Add		<null>
● Customer_id		No	Delete		<null>
● Withdrawal		No	Delete		<null>
● Transfer		No	Delete		<null>
● Account_ID		No	Delete		<null>
● Checking_account		No	Delete		<null>
● Deposit		No	Delete		<null>
● Saving_account		No	Delete		<null>
● Transaction		No	Delete		<null>
● Customer	● Customer_entity	Yes	Map	Directly-changed	Same superclass and subclasses
● First_name	● First_Name	Yes	Map	Directly-changed	different delimiters
● Surname	● Full_name	Yes	Map	Directly-changed	multiple unmatched siblings with same slots
● Address	● Address	No	Map	Isomorphic	frame name and type are the same
● Name	● Name	No	Map	Isomorphic	frame name and type are the same
● Middle_name	● Middle_name	No	Map	Unchanged	frame name and type are the same

Figure 23- Comparison of the concept of employee and the concept of customer

In using the Prompt, mapped classes across the two ontologies are indicated in the bottom section where classes that have been evaluated as having the equivalent have the attribute “map” to indicate either Isomorphic Relation (nothing is different) or “Directly Changed” to indicate that there are some differentiations and finally to add or delete to indicate that there is nothing in common.

In using the Prompt process devised by Noy and Musen (Noy et al. 2000), one can see that eventually all these steps will become more automated and should eventually include some calculation of a figure of metric indicating semantic closeness.

Ontology Alignment Analysis with a PDO

Prompt is able to compare the Employee and Customer Ontologies via the versioning capability. The versioning analyses terms and classes in terms of similarity and difference. The Prompt support for versioning offers support for the alignment analysis phase of model but is not able to leverage all of the class structural information that is available across the two ontologies; the versioning is a step-driven process. In using it as a one-step phase analysis, not all the capability of Prompt is leveraged. Notwithstanding this limitation, the Prompt capability does permit an initial alignment analysis capability of sufficient detail to execute the Alignment decision table. In this theoretical test the ontology of Employee is very different to both in the terms and the structure than the ontology of Customer. Direct comparing of these two ontologies is done to exercise the Alignment Criteria. In this case there is no Policy-Driven Ontology at play.

In comparing the ontology of customer to one of employee as a versioning exercise to test what gets added and deleted between potentially similar ontologies, Prompt results showed that it was syntactically right in identifying common labels to entities. It is not, however, able to tell if these alignments are semantically wrong and that the label of Customer and Customer_entity are in fact representing very different concepts.

This test was set up to show one of the main challenges in approaching data interoperability analysis at an ontological level. There needs to be more discriminating structure at play during the comparative analysis phase. Alignment analysis indicates which classes are equivalent but in no preferred order. This means that some discrimination to be done at the syntactical level via the label level and semantic level by the structure of the ontology itself.

Simply comparing one ontology to another, in conditions such as these, where there is no Policy-Driven ontology to act as an additional discriminator, will always result in a situation of semantic ambiguity. The PDOAM approach addressed this type of semantic ambiguity by forcing a referent model to influence the interpretations in the various comparisons.

The next level of analysis is done using a Policy-Driven Ontology to show its role in imposing semantic control and governance.

Comparing a Domain Ontology against a Policy-Driven Ontology

In this example, we will set the Customer Ontology to act as the Policy-Driven Ontology against the Domain Ontology of an employee when labelled as Customer_entity. This will show the contribution of an ontology in the role of a Policy-Driven Ontology to the model. This ensures that there will be priorities established as to which classes will be predominant and can act as part of the conflict resolution process based on its organizational legitimacy. For the purpose of the exercise the “customer” ontology takes on the new role of a Policy-Driven Ontology.

Figure 24- Policy-Driven Ontology Comparison

In comparing Employee and Customer within Prompt the following differences are highlighted. In blue and red are classes unique to each ontology they are disjointed from

one another. The set of common classes are presented in black and shown as the common set of classes across the two entities.

As seen in Fig 24 there is a prioritization of ontologies that now has to occur “one over the other”. Prioritization of one of the ontologies over the other is a critical step highlighted in the Policy-Driven Ontology Analysis Model. In this fashion there is identification of the role of a dominant and a subordinate ontology. This prioritization can now be factored into the analysis process. One can see that, in reference to the Concept of Customer when in its role of Policy-Driven Ontology, some concepts of the employee overlap, such as the class Name, some are removed such as id_number (which is an employee concept) and some are added such as the Customer-id.

In the context of evaluating the class alignments that exist between the employee ontology and the customer ontology, there necessitates a requirement to determine which ontology will be prioritized over the other. When an ontology acts as a Policy-Driven Ontology, the resulting outcome of the comparison is quite different, as shown in the Fig 24.

In this context the additional authority and influence of a Policy-Driven Ontology permits the resolution of some of the semantic ambiguity. The Policy-Driven Ontology defines the core reference model for every other domain ontology and although most of the process has not changed, it is essential to prioritize certain ontologies during the alignment analysis process. A PDO now can act as not just a referent concept with a higher level of legitimacy but also as a design requirement within a governance process.

In our theoretical example the uppermost class “Customer” has been chosen as the policy-driven ontology over the domain ontology of the employee and the alignment of all classes are made in reference to organizationally validated concepts, structures and terms. In the example there is an identification of excluded classes that are not part of the Customer ontology.

From Alignment Analysis to Interoperability Analysis

The purpose of the Policy-Driven Ontology Analysis Model is to identify potential interoperability of data sets based on a comparative analysis of their ontologies. As shown, the present state of knowledge with regards to ontological engineering environments does permit a semi-automated approach to the problem and can show where there is alignment between classes both at the term level as well as the structural level. However, at present, without the creation of an organizationally policy-driven ontology, semantic alignment is impossible.

In comparing the Policy-Driven Ontology of the Customer with the Domain Ontology of the Employee (labelled Customer_entity) an alignment is produced. Based on this outcome and using the interoperability table presented earlier, it is possible to attribute a category of interoperability potential that exists between the two data sets. In this example there is outcome of Synonymous Overlap between the two ontologies which indicates that there is potential for partial semantic interoperability between the two datasets on which these ontologies were created. Based on this result and in the context of a common application in which instances of Name and Address in one application map to Name and Address in the second application they are to be considered semantically equivalent in reference to the notion of Customer.

Term Analysis

In reviewing the table view of the terms and class alignments (seen in the previous chapter) the following are to be noted.

- There are a total of 23 terms used as labels across the two ontologies
- 2 class were found to have identical terms
- 2 classes were identified as identical terms despite the fact that there was a spelling difference (First_Name vs First_name)
- 2 classes were considered to have synonymous terms (Surname and Full_name)

- 2 classes were considered synonymous terms even though it was a composite name where Customer_entity was considered to be terminologically equivalent to Customer
- The other 12 classes had non-recurring terms that appear only once in either ontology.

Class Analysis

Of importance to note is that almost all of the classes that were considered terminologically equivalent were also considered to be structurally equivalent. The false positive with regards to class Customer_entity that was considered equivalent of the class Customer was not picked up in the alignment analysis. Although in the end, the use of a PDO makes this issue secondary as the classes in Customer_entity was in fact subsumed by the broader class of Customer and common classes are aligned appropriately.

Testing the Model with Theoretical Data

The second theoretical case exercising the PDOAM will be between the customer ontology in its role of PDO and two new constructed domain ontologies. To determine if there is potentially data interoperability between the two domain ontologies they will be compared first to the Customer Policy-Driven ontology knowing that these ontologies were constructed to demonstrate interoperability via subset alignment that will overlap within concept of customer.

In figure 25 these two new domain ontologies are represented. One domain ontology will define a Branch (Bank_branch) and the other domain ontology will define banking products and services (Banking Products_and_Services)..

Figure 25 - Domain Ontologies- Bank and Products and Services

The application of the PDOAM identifies the following alignments between the PDO and the two DO as are shown in the figure 26.

The two domain ontologies have been aligned to the Customer PDO, as previously defined (in fig 22), indicating where the classes show similarities. They show both a condition of Synonymous Subset. The resulting interoperability matrix (table 4) indicates that this situation occurs when both domain ontologies are Subset of a Domain Ontology. In turn this results in a situation of Limited Interoperability. This was expected from the construction of the test scenario. As the data sets were developed for this case, it can be said then that the theoretical model demonstrates internal coherence.

Figure 26- Alignment of two domain ontologies against the PDO of Customer

Alignment	Equivalent	Subset	Overlap	Disjoint
Equivalent	Interoperable	Limited Interoperable	Limited Interoperable	Non Interoperable
Subset	Limited Interoperable	Limited Interoperable	Limited Interoperable	Non Interoperable
Overlap	Limited Interoperable	Limited Interoperable	Limited Interoperable	Non Interoperable
Disjoint	Non Interoperable	Non Interoperable	Non Interoperable	Non Interoperable

Table 4- Interoperability analysis of two Domain Ontologies

Testing the PDOAM with Organizational Data

Now that the requirement for model coherence has been demonstrated via the use of theoretical data sets, the next step in the evaluation of the model is to use organizational data sets within the same validation constraints of apriori knowledge of outcomes to demonstrate model correspondence. This enables us to review how the model behaves under even more ambiguous conditions and to evaluate some of the PDOAM limitations and strengths. Here again the ontological engineering environment will permits us to model and exercise the analysis process. Using the set of alignment and interoperability criteria it is now possible to evaluate data interoperability based on a comparative analysis of their ontologies. This ontological analysis model was not available to data and information systems architect previously.

Sources of Organizational Policy-Driven Ontologies

The organization used had also stated an investigation into how to improve data quality issues within the context of application interoperability challenges. It is within the context of this activity that the data was collected.

The objective of their work was to create a definitive common information model that defined the concept of Customer at an enterprise level. The organization also developed business information standards as well as defined key data elements. It is these data sets that are going to be used to create the Policy-Driven Ontology of the concept of *customer*. In conjunction with the development of this common definition, a data standard definition process was also created. As a result a total of 23 key data elements were used too create a standard customer definition. Findings from participants obtained by exercising the business standards in that project identified major difficulties in developing this common concept:

- There were elements of the concept missing across all applications.
- Incomplete attributes such as “what is the correct length, valid values, and format?”
- There were incomplete business rules (edits, or implementation guidelines)
- There was no enterprise information model (required for a standard schema)
- Participants had trouble understanding the standards sufficiently to assess whether their applications were in compliance or not with the stated definitions.

In addition, the question was asked as to how to know if a given application is in compliance with an information standard of data or metadata? No standard processes or criteria were declared or to be found. In addition concern about the consequences that come from differences between enterprise-based metadata definition and the applications were enumerated but never resolved.

:

- Differences in technology lead to differences in application (actual field lengths, actual values, actual data types),

- Structural differences in the information structures, arrays vs. named fields, etc. E.g., if an application's length is longer than the standard, should it be considered 'non-compliant'.
- If the information model of the application is different than the standard, when then should it be declared 'noncompliant'? Will this have a regulatory impact or a change in compliance expectation for the organization?

The standardization initiative represented the first steps of a Policy-Driven Ontological Analysis Model process and an effort to create a PDO. All the initial steps of the PDOAM methodology were present in the initiative. But as they are not structured and articulated within an ontological engineering body of knowledge, tools, and techniques, that initiative had to limit itself to only one concept - *the customer*. Other observations were that:

- There was no overall methodological and representational logic underlying the representational activities that could have supported a possible structured analysis process.
- Although there was the creation of a reference model, there is no structured approach to a comparative analysis process derived from the representational schema that would meet criteria of ontology consistency as well as model coherence and correspondence across all outcomes.
- There was no complete comparative analysis framework that integrated all dimensions (i.e. structural and linguistic) all at the same time.

As stated in order to be efficient, the review was limited to just one core enterprise concept - the customer. To be truly effective, the work needs to be extended to other organizational concepts in order create an overall and complete set of corporate ontologies supporting an integrated enterprise model. So in addition to Customer, the following supporting concepts would also need to be formalized into unique Policy-Driven Ontologies.

- Offerings (products, services, accounts)
- Resources (people resources, info resources, facilities, infrastructures)
- Transactions (events, communications, messages)
- References (codes, Domains, Classifications)
- Locations (Geo-spatial, Fixed, areas)
- Policy and regulations (regulatory, HR, Legal)
- Directions (Plans, Strategies, Goals, Campaigns, Programs)
- Contracts (terms, conditions, arrangements)
- Finances (Metrics, capital, cost. infrastructure)
- Markets (Line of Business, Business areas, businesses)

The use of the PDOAM would have recognized the requirement for more PDO as well as addressed some of the methodological shortcoming mentioned earlier. Ultimately this illustrates the need for an ongoing research to address these management issues but at present this is beyond the scope of this research. However, it is important to mention that the use of the PDOAM will result ultimately in a more complete and enumerated organization because the main business models will be based on aligned concepts through the construction of an integrated set of Policy-Driven ontologies.

Constructing the Policy-Driven Ontology of the Concept of Customer

As stated, in order to apply the PDOAM to assess data interoperability between applications it is necessary to construct a Policy-Driven ontology as well as a set of domain ontologies. Using the work done during an initiative to standardize and operationalize the concept of customer it was possible to construct a Policy-Driven Ontology. Creating a Policy-Driven Ontology also allows for the start of a management process as well as helping identify and integrate all information that was currently held but managed across different organizational domains.. (Vergara et al. 2002a)

The organizational approach was essentially a paper and pencil driven activity, through which analysts reviewed past application architectures, laws, and related corporate

policies to create a conceptual representation of the elements and attributes that an ideal customer of the bank would have. The representation was constructed by also reviewing many of the existing business critical applications. In retrospect, this was done in accordance with the Pollock and Hodgson (Pollock et al. 2004) approach of artifacts required to construct enterprise ontologies. They suggest reviewing design artifacts such as:

- Requirement documents
- Modeled objects
- Logical representations
- Conceptual representations
- Business rules
- Data dictionaries, taxonomies, and thesauri
- Workflow models
- Interfaces and formal forms.

The representation in figure 27 shows a holistic understanding of the customer. The 23 data entities that constitute the Policy-Driven ontology of a customer are represented and structured as follows:

Customer Information Integrity Requirements — Holistic View of Customer

those highlighted in red have been standardized.

Figure 27- Holistic view of the customer

This enumeration of the different aspects of the customer was used to construct an ontology that contains main elements with associated attributes. These elements were used to construct within Protégé the Policy-Driven Ontology of the Customer.

The result is a Policy-Driven Ontology of the concept of customer that has a hierarchical structure as shown in figure 28. This is a very large ontology compared to all application domain ontologies. The XML/RDF version of this is in Appendix E.

This Customer ontology can now act as the Policy-Driven Ontology within the PDOAM and will be used throughout the rest of tests against all the various application domain

ontologies. The representation in the following figure shows only the high-level elements and the full detail of the ontology entities with its attributes can also be seen in Annex E.

Figure 28 – Main Entities of the Policy-Driven Ontology for the Concept of Customer

Constructing Domain Ontologies using Organizational Data Sets

The creation of the Policy-Driven Ontology of Customer was based on the formal hierarchical data association that was extracted out of the enterprise documents. In effect, this ontology can also be described as a formal taxonomy (Pollock et al. 2004). Another way to qualify this ontology would be to say that it is structured and formal but not to point where it is can be used for executing formal inferences using descriptive logic clauses via Ontology Web Language (OWL).

Still the formal nature of the ontology provides enough structure for analysis using the implicit parent-child relationship between the data elements. In that sense, all the elements have descriptions, properties and relationships.

Another approach for constructing the PDO could have been by using some kind of Relational Schema if available. This would also permits representation of the classes, attributes and some relationships. But as observed by Pollok and Hodgson, “the use of denormalized structures for an implementation-specific rationale limits the applicability of most existing relational schemas as a candidate for true information ontology in the classic sense” (Pollock et al. 2004).

However it is important to note that this level of representation of the PDO has also an influence on the way the DO are constructed. All the domain ontologies will also have to move to the same level of a formal taxonomy. The decision to stay at this level of detail ensures both a common abstraction process in creating ontologies and invoking a same level of comparison. This can be considered as a ontology normalization step.

In following the process outlined in the PDOAM the following steps are required to be executed using the organizational data sets:

1. Create a policy-driven ontology and domain ontologies
 - a. Map the vocabularies and class structures into structural relations of a PDO.
 - b. Map the vocabularies and class structures of the Domain Application into a DO.
2. Analyze the alignment using the alignment table of criteria as presented in between the various PDO and the DO.
3. Determine similarity and differences with regards to labels and structures.
4. Determine data interoperability between the domain ontologies by means of the Interoperability matrix.
5. Document evaluations and observations.

Validating the Model using the apriori Knowledge of Outcomes

The research established the way to evaluate the model: it is to observe if the model outcomes are similar to prior known outcomes. Three applications have been chosen to be reviewed by the model. They are all taken from within a suite of applications that deals primarily with financial customer services. It is important to keep in mind that these applications were created with different contexts, in generations of technologies and with differing design frameworks over different time frames.

It is important to note that the two first applications (A and B) were chosen because they have common elements that will permit data interoperability around some of their data elements. As well Applications C was chosen for the opposite reason: because it was not apparent as able to be made interoperable at any level against the two others.

Taking into consideration the apriori knowledge of outcomes the following results are to be reconfirmed by the PDOAM:

Application A and B will have some interoperability opportunities around the concept of customer and Application C will not have any common elements with the enterprise concept of customer. Results should indicate that there is no potential for data interoperability between application C and application A or B on that concept.

Application A - Domain Ontology - Involved Party

Application A is a business-critical, transaction processing legacy application. It is one of the oldest applications in the institution. It is very large and very complex and acts as the main engine for all transactions. In addition, it acts as a repository for many secondary applications. The technology and design principles are older and in some cases unique. The terms and labels used by this application are generally not common or standard, and were designed under the constraints of memory and storage. However one of the design

principles was to have “meaningful names”. This was part of the initial set of design and programming principles.

One of this application’s main concepts is labelled “Involved Party” and its review shows how that concept encompasses both the notion of corporate customer as well as individual customer. In its entirety, it composed 45 entities. A review of its structures shows relations to concepts that deal with contracts as well as with the notion of role, name, business imperatives, and relationship with a primary person in the company in addition to some financial attributes. Here employee really plays a role of relationship management.

Figure 29- Ontology of Involved Party from Application A – Involved Party

Alignment Analysis of the Involved Party Domain Ontology

The next step in the model is to do the Alignment Analysis of this domain ontology with the Policy-Driven Customer Ontology (PDO). The results are shown in Figure 30.

Prompt indicates that there are many alignments - some are based on terms and some are structural.

The results show that, this Domain Ontology is a Synonymous Subset of the Policy-Driven Ontology of a customer. As seen, some of the elements in the Domain Ontology map directly to similar ones in the PDO.

f1	f2	renamed	operation	map level	
Birth_date		No	Delete		<null>
Contact		No	Delete		<null>
DefaultOntology		No	Delete		<null>
Preferred_contact		No	Delete		<null>
Prefix		No	Delete		<null>
Branch		No	Delete		<null>
Departement		No	Delete		<null>
Annual_income_amount		No	Delete		<null>
Sharing_indicator		No	Delete		<null>
Individual		No	Delete		<null>
Locator		No	Delete		<null>
Household		No	Delete		<null>
Registration_date		No	Delete		<null>
Group_name		No	Delete		<null>
Middle_initial		No	Delete		<null>
Customer_since		No	Delete		<null>
Name		No	Delete		<null>
Hours		No	Delete		<null>
Business_identification_code	Business_identification_number	Yes	Map	Directly-changed	Names are similar
Customer_role	Customer_ID	Yes	Map	Directly-changed	Names are similar
Involved_party	PDO_customer	Yes	Map	Directly-changed	multiple unmatched siblings with same state
Relationship_managers_code	Relationship_Manager_Code	Yes	Map	Directly-changed	different delimiters
Role	Customer	Yes	Map	Directly-changed	Same superclass and subclasses
Address	Address	No	Map	Directly-changed	frame name and type are the same
Age	Age	No	Map	Directly-changed	frame name and type are the same
Banking_decision_maker	Banking_decision_maker	No	Map	Directly-changed	frame name and type are the same
Currency_code	Currency_code	No	Map	Directly-changed	frame name and type are the same
Designation	Designation	No	Map	Directly-changed	frame name and type are the same
Email	Email	No	Map	Directly-changed	frame name and type are the same
Functional_title	Functional_title	No	Map	Directly-changed	frame name and type are the same
Gender	Gender	No	Map	Directly-changed	frame name and type are the same
Language	Language	No	Map	Directly-changed	frame name and type are the same
Legal_name	Legal_name	No	Map	Directly-changed	frame name and type are the same
NAICS	NAICS	No	Map	Directly-changed	frame name and type are the same
SIN	SIN	No	Map	Directly-changed	frame name and type are the same
Telephone	Telephone	No	Map	Directly-changed	frame name and type are the same
Trade_name	Trade_name	No	Map	Directly-changed	frame name and type are the same

Figure 30 - Alignment of DO for application A with the PDO

Application B – Domain Ontology – Individual

Application B is a narrowly-focused private client application, unlike the transactional application A. This application is built with much more recent technology than application A. Within application B there is a concept called Individual with related

elements. As in application A, the ERD for application B involves a set of hierarchical elements that can be used to construct an ontology. Using this information is possible to construct the Domain Ontology of the concept of Individual as shown in Figure 31 below.

In many respects this ontology might meet most of the policy-based definition of the customer within the enterprise, as there are some of the main elements for demographic information, and it also presents a listing of financial assets in the form of financial products such as bonds, mutual funds, stocks as well as loans and mortgages.

Figure 31 - Ontology of Application B – Individual

The representation of a customer in the context of this application is mostly as a holder of paid financial goods. As such, this instance of customer is as an asset holder with a listing of associated products. The notion of a customer being a corporate entity is not present in this application as it was in A. Hence, there are no elements or attributes that have to do with a representation of a corporate entity.

Alignment Analysis of the Individual Domain Ontology

The alignment analysis generated the following listing of similarities and differences. There is some alignment in both names (address, email, ID and Telephone) and one structural alignment (Individual aligned to PDO_Customer). For the rest there is no other element aligned between the two ontologies.

Ⓢ Stocks	No	Delete	<null>		
Ⓢ Income	No	Delete	<null>		
Ⓢ Name	No	Delete	<null>		
Ⓢ Tax_info	No	Delete	<null>		
Ⓢ Others	No	Delete	<null>		
Ⓢ Middle	No	Delete	<null>		
Ⓢ Products	No	Delete	<null>		
Ⓢ Bonds	No	Delete	<null>		
Ⓢ Mutual_funds	No	Delete	<null>		
Ⓢ First	No	Delete	<null>		
Ⓢ DefaultOntology	No	Delete	<null>		
Ⓢ Loans	No	Delete	<null>		
Ⓢ Mortgage	No	Delete	<null>		
Ⓢ Investments	No	Delete	<null>		
Ⓢ Individual	Ⓢ PDO_customer	Yes	Map	Directly-changed	multiple unmatched siblings with same slots
Ⓢ Address	Ⓢ Address	No	Map	Directly-changed	frame name and type are the same
Ⓢ Email	Ⓢ Email	No	Map	Directly-changed	frame name and type are the same
Ⓢ ID	Ⓢ ID	No	Map	Directly-changed	frame name and type are the same
Ⓢ Telephone	Ⓢ Telephone	No	Map	Directly-changed	frame name and type are the same

Figure 32- Alignment of DO -B with PDO

One of the potential explanations for some of the misalignment between the two ontologies could be the way the entities have been modeled. The PDO defines financial products in terms of Customer Assets which include all types of products. In this domain ontology these entities are modeled as specific products such as mutual funds and bonds. Hence, even though there are some overlap between the DO and the PDO, there are also some aspects of the DO that are not mapped to the PDO. The concepts modeled in the DO are not sufficiently detailed or generic enough to produce either the structural or naming alignments (Pollock et al. 2004) for all elements.

In final analysis, there is still partial alignment mostly around demographic information in spite of the variation in context, scope, intentionality and modeling and design processes. In this case, also as predicted, the DO of Individual is a Synonymous Subset of the PDO.

Application C – Domain Ontology – Party

This is the most modern application and is, in fact, modeled as an inter-organizational B2B application. The concept of a customer here is taken into a broader context. In this application a Party is both a seller and a buyer. The consequence of this different world view interpretation of a fact is why this application was chosen as part of the validation testing process. There is sufficient ambiguity of structure and terms to ensure that there would be a disjoint outcome between the PDO and the DO.

The concept of Party is generic encompassing both Seller and Purchaser. This concept makes no difference which side of the transaction an agent situates itself. Party acknowledges that a transaction relationship is bi-directional. The concept of Party does not define one specific role. It defines two traditional roles under a new class of behaviour. Party makes the buyer and the seller equivalent in this ontology. A Party can be situated on either side of a transaction and that the concept of a transaction itself has now been reduced to a line item instead a broader listing of a class of products or services. This also means that there is some ambiguity with regards to the concept of Party as it relates to the concept of customer as consumer is a specific instance of buyer.

Figure 33 - Ontology of Party from Application C -Party

Alignment Analysis of the Party Domain Ontology

The DO for Party when analyzed against the PDO shows one major term alignment as seen in figure -34. It also shows a structural alignment in that Party and PDO_customer having some structural class similarity. One of the classes shows an alignment between Party_identification and Customer.

In accordance with the PDOAM, as small as the alignment is – this indicates that there is a potential for data interoperability. Therefore, the outcome of this evaluation must be recorded as a Synonymous Subset to the PDO as there is both a small structural and a naming alignment

Table view **Tree view**

Image table

f1	f2	f... op...	map level	rename explanation
⊙ Line_item	⊙ DefaultOntology	No Add		<null>
⊙ DefaultOntology		No Del...		<null>
⊙ Buyer		No Del...		<null>
⊙ Contact		No Del...		<null>
⊙ Seller		No Del...		<null>
⊙ Name		No Del...		<null>
⊙ Communication		No Del...		<null>
⊙ Tax_schema		No Del...		<null>
⊙ Type_of_party		No Del...		<null>
⊙ Party	⊙ PDO_customer	YesMap Directly-changed		multiple unmatched siblings with same slots
⊙ Party_identification	⊙ Customer	YesMap Directly-changed		Same superclass and subclasses
⊙ Language	⊙ Language	No Map Isomorphic		frame name and type are the same
⊙ xsd:int	⊙ xsd:int	No Map Unchanged		frame name and type are the same

Figure 34 - Alignment DO C with the PDO

Assessment of Potential Interoperability between Application A B and C

The Alignment analysis for all three applications is complete and can be reviewed. All applications have been found by the PDOAM as having some elements of their customer DO aligned with the PDO.

This means that in accordance with the Interoperability matrix it is possible to assert that when applications are aligned as Synonymous Subsets to the Policy Directed Ontology there are Limited Interoperability opportunities between the applications.

The PDOAM findings are in accordance with the apriori know knowledge of the outcomes for A and B. In the case of Application C were the organization felt that in practice there was no potential for data interoperability, the PDOAM found contrary results. The PDOAM shows that in fact, there is some alignment between the PDO and the DO. The results are analyzed in more detail in the next chapter.

CHAPTER 6

ANALYSIS AND DISCUSSION

Analysis of the Results

Overall the model has shown to be able to predict similar outcomes using theoretical and field observations but with some qualifiers. A general review and summary of the apriori interoperability outcomes is shown in the following table.

Table 5-Apriori Interoperability outcomes

Application n	Apriori Interoperability knowledge
A	This application has common data elements that were also part of application B. This was due to the fact that A is one of the initial bank applications which means that it had captured core elements that were carried over to application B. It was not seen as having any elements that were interoperate with application C.
B	This more recent client service application was found to be interoperability with A as the concept of customer is common with most client oriented application. It was not deemed to be interoperable with Application C
C	The B2B had overloaded the concept of party with two intentions. The notion of Party in application C is both a Customer and A Vendor. The concept of Party was deemed not to be interoperable with other two applications.

Table 6- PDOAM Outcomes

Application n	Outcome of the PDOAM
A	PDOAM found that there was some potential for data interoperability between it and application B. The ontology has some semantically aligned elements with the Policy-Driven Ontology of Customer.
B	This more recent client service application was found to be interoperable with A as the concept of customer is common with most client oriented application.
C	The B2B had overloaded the concept of party with two intentions. The concept of Party had both buyer and seller. However there were elements of the Ontology C that were aligned with the Policy-Driven Ontology. Elements of this concept were deemed to be interoperable with application A and B.

A review of these two tables shows that the PDOAM was able to replicate most of the same the observational organizational expectations but in addition identified some issues that were overlooked. The PDOAM indicated that application C had some elements that permitted semantic alignment with A and B.

The PDOAM delivered more truthful findings and conclusions than the organizational paper and pencil analysis. The PDOAM has permitted the reproduction of results that are congruent with the apriori observations. The results are coherent and consistent with one another which supports the assertion that the model is empirically adequate (Psillos 1999) and as well as sufficient to determine the potential for data interoperability.

Discussions and Observations

In practice, in all organizations and at any time, there are multiple ontologies at play that are used across multiple applications. The PDOAM framework can be used to identify which portions of an ontology can be interoperable with another through a review of its alignment to a common Policy-Driven Ontology. One of the main reason for the diversity of ontologies when dealing a same concept is that ontologies are initially created with a specific *perspective* aligned to the domain where the concept is used (Staab et al. 2004).

The challenge of working in a multi-ontology world is to first identify how ontologies are similar and how they are semantically different. The Policy-Driven Ontological Analysis Model structures ontologies in relation to a underlying world view. The PDOAM structures both a referent concept and the domain concept as well as making the review process systematic. Inevitably, this capability will also become automated.

The content and structure of the matching classes in the source ontologies (names of classes and slots, subclasses, super-classes and domains and ranges of slot values) permit us to determine a similarity and difference outcome. In our experimentation, even though the Protégé/Prompt environment is leveraged in a non-standard way by using the decision criteria regarding alignment and interoperability, the model is able to reproduce known results from both a theoretical and practical perspective. The use of the model ensures coherence and correspondence in the analysis.

Therefore it is not surprising that in the case of Application C, where there was a clear expectation of Disjunction and non-interoperability based on organizational observed expectations, the model showed that there was in fact there was a special case and clearly some potential for data interoperability. The most appropriate and significant explanation is that, in an organizational context, when it came time to assess interoperability challenges, not having core enterprise policy-driven ontology to act as model reference, the concept was deemed to be too different to the world view. Initially it was chosen as a good case for demonstrating a Disjunction. But in final analysis, the PDOAM determined it to be acceptable for a data interoperability initiative.

The PDOAM determined that within the calculus of interoperability, these three applications have some common semantics.

Significance and Contributions of the Model

The proposed model (Garigue 2004) is grounded in part in the research from Resnik on semantic similarity (Resnik 1998) and Visser research on ontological mismatch (Resnik 1998; Visser et al. 1997). In the proposed model, semantic equivalence is determined via an ontological analysis where domain ontologies are compared to standard Policy-Driven Ontologies.

This research will have wide spread impact since it will enable better understanding of semantic web issues. It will help in the creation of better medical guidelines for managing patient health, as well as improve collaborative engineering design activities. Finally, it will have an impact on the present practices of security analysis of B2B business processes such as the automated exchange of electronic information among commercial trading partners. For example, the ONTOLOG Forum, formed around a project, is to discuss practical issues and strategies associated with the development of both formal and informal ontologies used in business. Expected results are to identify ontological engineering approaches in the process of developing a Universal Business Language (UBL) ontology.

The justification for ontological modeling research is now very strong due to its extreme timeliness. There are tremendous benefits for organizations in managing policy-based coherent, correspondent and complete ontologies across multiple datasets, applications and using them to address the interoperability challenge.

- Using Policy-Driven Ontological Analysis, one can apply engineering and management principles to develop and maintain a coherent and integrated set of enterprise information component in the form of ontologies. This set of Policy-Driven Ontologies would act as standard business components for business

models. In turn this would increase the integrity of the data sets, permit interoperability as well as reduce costs.

- This approach would also introduce a new layer of abstraction, independent of the constraints of technology infrastructure or the implementation process. As a result, application development could then leverage information standards, common practices and an integrative environment more completely.
- Policy-Driven Ontological Analysis helps to identify the relevant concepts in the business and to describe their relationships in a way that can be later executed computationally through software implementations and across multiple business models.
- Having an understanding of the importance of Policy-Driven Ontologies, it behoves the organization to realize that ontologies are critical information assets in and of themselves. Identifying their presence, and analysing their use, will enable organizations to execute more sophisticated and complex business models.

The PDOAM permits these technical and managerial contributions because the methodology permits the evaluation of essential components in all business models. Ultimately this will ensure integrity of their data across functional applications. This will result in a semantically aligned organization, supporting integrated business models.

The PDOAM permits a critical review and analysis of all the main elements of an information architecture by ontologically modeling, systematically, all the Domain Ontologies as well as the enterprise-wide Policy-Driven ontologies. As such, this model brings into play a methodology that is as valuable to practitioners as it is to academics in analyzing organizations from an epistemological stance. A Policy-Driven Ontology methodology also permits the maintenance of the integrity concepts over time (Zhao 2003). A Policy-Driven Ontology ensures that there is no diffusion or drift of the meaning. It maintains the overall structure of the concept over time and across implementations.

The contributions of this research are at two levels: one is theoretical in the creation of the Policy-Driven ontological analysis model, and the second, is practical since it is demonstrated in a research using field data. The findings of this research also demonstrate how the use of ontologies can help identify and resolve semantic concerns. This is important because the creation and management of ontologies is becoming an enabling organizational strategy. In this context, when markets change, businesses can adapt to new environmental requirements by making changes to their underlying ontologies or adopting ones that align to their new underlying business models and concepts.

In essence, The PDOAM would force the creation of an ontological worldview of the enterprise. It would will also determine a set of ontology management activities that enable the enterprise, positioning it for new behaviours and specific activities (Gandon 2002). However, for this objective to be realized there is a requirement to use the PDOAM methodology to construct and legitimize Enterprise level Policy-driven Ontologies, and to use them in a standard way. This would also ensure interoperability across every application.

The findings of this research also impact the ontological engineering community by providing additional criteria for evaluating the quality of ontologies. Awareness of the foundation and practices that permit ontological interoperability ensure better correspondence over time between domain knowledge. This leads to smaller and shorter information system and application development cycles. By reducing the size and complexity of an application initiative, the risk of failure is reduced and the costs are reduced. From an economic point of view, organizations investment programs that deal with interoperability problems cost hundreds millions of dollars. So this contribution will also have an impact on investment strategies.

The PDOAM highlights the value of ontology management and the requirement for a new specific core organizational function within the IT community. Organizations will have to develop specific and commensurate management processes that ensure a measure

of quality of their ontologies. In fact, PDOAM augurs the beginning of a new discipline in organizations, specifically focusing on semantic management as the primary value-creation process.

In this respect, resolving ontological interoperability becomes the dominant activity in all information intensive organizations and a major part of the overall knowledge management focus and practice. Since business ontologies are called upon to link business strategies with business processes, the role and strategic importance of ontologies will increase accordingly. For that reason, the emerging practice of ontology engineering and ontology management requires better understandings of all the mechanisms that permit semantic interoperability. This means that more research needs to be proposed and relevant findings need to be formalized and diffused across the academic community as well as throughout the community of practitioners within organization.

At present, most organizations are at only the initial stages of maturity in their information management practices. It is not clear how organizations will life-cycle manage ontologies as distinct artifacts. At best, this will start with metadata repository issues. But as seen, in order to study ontological interoperability, there is a requirement to first extract and analyze the existing ontologies that are already hidden and embedded in applications across the organization. This research has given the research community an understanding why and how it is done.

Limitations of the Model

This approach to the study of data interoperability from an ontological perspective has limitations; some due to the fact that ontological engineering practices and the related know-how are still emerging. The tools for manipulating and analyzing ontologies are still fragile. More importantly, moving ontological analysis from the realm of academic context into the context of organizational processes such as this research has done, requires accepting practical limitations.

The semantic dimension of an organization will eventually be supported by a 'web of linked ontologies' since each information source will also have its ontologies articulated in a formal way, accessible to other applications (Osterwalder^y 2001). In addition, arguments concerning a semantically-capable organization are premised on the fact that there will be multiple ontologies in the organization that will have to be regulated in some way to ensure that the organization will achieve overall coherence across various applications, data repositories, and information systems (McComb 2004; Vergara et al. 2002b). Only within a Policy-Driven Ontological management context will ensure that this type of discourse remains valid.

Ontology extraction from existing applications is at best a difficult procedure. But it is an important process which ensures that ontologies have organizational legitimacy and are capable of representing the main concepts that are being used across data repositories. This extraction is a necessary part of the research but still open to more research to ensure a more structure approach. There are also some practical considerations with regards to concept extraction and ontology creation that are not yet well understood and not specifically addressed during the modeling phase of this research. There might be some additional steps required to ensure that the ontologies created, based on the differing datasets, are considered legitimate. The process of creating ontologies from the metadata requires a more standard process. The ontology creation process must be both a repeatable and structured process that will guaranty the creation of valid ontologies.

The present model process of comparing ontologies is a domain-level activity and does not leverage the existence of a Common Vocabulary and the existence of a Top/Upper level ontology. The PDOAM matching model does not take advantage of the top-down structure that is in effect and that is outside of any specific domain of knowledge (Nirenburg et al. 2001). It is possible that with the addition of those elements, an outside framework could also determine ontology alignment. In the case of the Common Vocabulary model, the locus will be with the governance that structures the vocabulary.

This research shows also that eventually all Policy-Driven Ontologies will be decided outside the governance any one organization. For example the United Nations controls the structure and content of the UN Standard Product and Service Codes (UNSPSC) that can be used to resolve e-Commerce interoperability functions across various applications. More and more interoperability issues are being resolved by outside vocabulary standardization mechanisms as they are introduced in organizations via common business processes protocols. These protocols are defined by business working groups and standardization bodies such as RosettaNet, UDDI, W3C, CPExchange, ebXML, and XBRL.

Business standard formal taxonomies/ontologies are being made available to facilitate business services (Obrst et al. 2001; Zhao 2003). Traditional data and application integration solutions have permitted some local successes but at great cost and effort. It is clear that in the long run organizations will be adopting common service-oriented architectures to connect systems. The availability of common, core ontologies does not make the problem on interoperability go away; it just makes the problem more visible and helps focus IT investments and business strategies (Schmelzer 2003).

In the case of standardization using a Top/Upper Level Ontology model, the locus of interoperability would be based on how a worldview is structured around principles of actuality and generic relationships between entities as they are affected by space and time. There are several Top/Upper level frameworks such as the Suggested Upper Merged Ontology (SUMO) and CYC. CYC is a knowledge base built upon a core of over 1,000,000 hand-entered assertions (or "rules") designed to capture a large portion of what we normally consider consensus knowledge about the world. The IEEE Standard Upper Model (SUO) Working Group is trying to standardize to a single model. Currently the evaluation of feasibility of system integration using general-purpose ontologies and upper-level models is expensive, mostly because of technical problems such as the different representation formalisms and terminologies used. Additionally, there are no formal mappings between the upper-level ontologies that could ease any kind of study

and comparisons. As a result, the Top/Upper-Level models are not widely used. These considerations did not present a problem in our research.

Another consideration in addressing interoperability challenges is how to leverage and integrate the results of a statistical analysis of instances. Instance-Based matching is the application of a bottom-up classification paradigm. In that model, the matching occurs using the clustering of a large set of instances, linking the resulting categories to common classes' across various domain ontologies. This is still a novel approach, simply because there are not many -different ontologies that subtend the same domain of instance. This particular method was shown to be possible in looking at the directories structures of Google and Yahoo, both of which are considered lightweight ontologies (Ichise et al. 2001). The co-occurrence theory could cluster (or group) similar classes, representing common concepts across different ontologies. Eventually, this approach can also generate an integrated (lightweight) common ontology and act a mapping function across various ontologies that deal with the same sets of instances. In another variant of the bottom-up approach, co-occurrence in the European part of CIA Fact-Book generates lightweight ontologies in specific domains (Fensel 2001b). However, this approach is still too recent and the opportunity to have two distinct classification structures over a same instance space is not easy to find. Successful work has been done integrating various catalogues of products (Bergamaschi et al. 2002). The instance-based model is not leveraged in our research to determine mapping between ontologies simply due to the fact that this would require access to the dataset instances themselves across each applications.

This means that the interoperability between the two domain ontologies is strictly based on the portions of concepts that are common elements with the Policy-Driven Ontology. It is possible that there are certain portions of the domain ontologies that have other classes in common with portions of other enterprise ontologies but since the mapping process is between the PDO and the DO, there is no way of ascertaining if these elements could be aligned to other PDO.

Finally the size of an PDO is a limitation. The effectiveness of the Policy-Driven Ontology Analysis Model is determined by the size and the structure of the Policy-Driven ontology itself. The benefit of a large number of small PDO is that they can be managed as legitimate element of the various enterprise departments. For example the employee ontology can be managed by HR, the Customer by the lines of businesses etc..

In demonstrating the model, one can see that there are major issues on how the organization life-cycles manages coherent Domain Ontologies and Policy-Driven Ontologies. If these ontologies are inadequate then the organization will have to deal with further semantic shortcomings that will be in turn biasing interoperability initiatives. The best approach would be to use the alignment evaluation phase of the model as a way to evaluate the completeness of the PDO itself. As part of the process it would be possible to continuously build this referent model by merging some of the elements of the domain ontologies with Policy-Driven one. This would be a way to update and to maintain the enterprise model. In the second approach one could merge some of the elements of the branch and products and services with the PDO so as to create complete enterprise model. Eventually and incrementally this would create a more complete and coherent PDO.

As the organization progresses into the life-cycle management of various types of ontologies, there will be a requirement for determining the minimum orthogonal dimensions that ensures that ontologies encompass the minimum set of classes for the entities they represent. This would ensure – as in our examples – Customer remains uniquely independent of the dimension in which it is contextualized.

Future Research: Alignment and Interoperability Metrics

Hughes and Benjamin (Hughes et al. 2002) have presented a series of metrics that are concerned with the correctness of the elements after an alignment has been performed. A number of these metrics are derivatives from within the information retrieval domain. These metrics can be used to improve on some of the limitations surrounding the

analysis, but as of yet are not part of the model. In many respects this is a way to further evaluate the efficacy of the model which at present is considered still too coarse.

These measures can serve the purpose of refining classification on “limited interoperability”. This offers some possibility to simulate alignments outcomes and to look for relationships between alignment issues and interoperability issues. For example, the following elements are a natural by-product of the Ontology Alignment analysis and can be used to further strengthen the model.

1. True Positives: the number of correct alignments
2. False Positive: the number of incorrect alignments
3. False Negative: the number of correct alignments missed
4. Precision: The proportion of correct alignments among those found ($\text{True Positives} / (\text{True Positives} + \text{False Positive})$)
5. Recall: The proportion of correct alignments among those found ($\text{True Positive} / (\text{True Positives} + \text{False Negatives})$)
6. Fmeasure: the harmonic mean of precision and recall ($2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$)
7. Alignment performance: indicates performance given the proportion of overlapping resources between ontology A and ontology B

As a brief illustration these metrics is applied to our first theoretical example to show how the metrics further support the PDOAM model. A Figure of Merit for Alignment can be calculated and is presented in the following table.

Table 7- *Ontology Alignment Metrics*

1. True Positives: the number of correct alignments :	5
2. False Negative: the number of correct alignments missed	0
3. False Positive: the number of incorrect alignments	1
4. Precision: The proportion of correct alignments among those found (True Positives / (True Positives + False Positive)	100%
5. Recall: The proportion of correct alignments among those found (True Positive / True Positives + False Negatives)	80%
6. Fmeasure: the harmonic mean of precision and recall ($2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$)	.8888
7. Alignment Challenge: the proportion of unique elements between ontology A and Ontology B to the total number of elements in ontology A and ontology B	.4782
8. Alignment performance: indicates performance given the proportion of overlapping resources between ontology A and ontology B (Alignment Challenge * Fmeasure)	.4250

This alignment Figure of Merit measure can also support decision criteria for which tools are better at evaluating alignment by also comparing Figure Of Merit against a PDO. In this manner, improvement in the ontology alignment and ontology comparison algorithms and tools can always be related back to accurate and reproducible measures.

CHAPTER 7

CONCLUSIONS

This research showed that the use of a policy-driven ontological analysis model is a way of comparing and identifying conceptual inconsistencies that prevent or enable data interoperability.

In support of this model, the concept of a Policy-Driven Ontology was introduced to create an enterprise-wide defined concept. This Policy-Driven Ontology acts as a reference model for comparing domain level ontologies. Furthermore this research proposed, described, and validated a novel approach in addressing this data interoperability challenge.

To support the PDOAM there was a requirement to create a comparative analysis framework as well as to leverage established criteria for evaluating the different types of semantic alignment possible between two ontologies.

By conducting a Policy-Driven Ontological Analysis between two ontologies, alignment between applications can be evaluated and assessed. The methodological research used to empirically validate the model was a set of known theoretical data sets in addition to field data with apriori knowledge of the data interoperability outcomes. This confirms the model to be empirically adequate (Psillos 1999).

In exercising the model, not only has it shown that it is able to recreate known outcomes, but while exercising it with organizational data it has shown to identify opportunities that were missed initially.

The model has shown that the outcomes from the model are the same as the ones that were based on apriori knowledge. The only comments with regards to its capability are

that the model appears to have more efficiency and sensitivity than expected, while identifying a potential opportunity for data interoperability that was not seen initially.

The major benefits that come from the use of the PDOAM are as follows.

- It will help organizations avoid creating applications that corrupt data sets because of semantically different underlying concepts.
- It will also permit more coherent business models since they are based on semantically aligned concepts across multiple applications. Eventually the use of the PDOAM will ensure compliance not just to business models, but to market expectancies of well-integrated value chains.
- Reduce the cost of application development and deployment because of standardization.
- As seen though the review of the theoretical underpinning for the Policy-Driven Ontological model, this approach also ensures data consistency, completeness and comprehensiveness over time and across the organization.
- Finally, it ensures that integration initiatives can be executed using a lower level of effort because it is now possible to predict which application can be made interoperable prior to starting the software design and development effort.

With the PDOAM enabling analytics of alignment and interoperability, the open ended ontological engineering issues such as, completeness and conciseness, expandability and consistency, will have to move from just being an ontological engineering concern to being considered more and more an ontology management problem. Therefore there is a requirement to understand better how and when to use ontologies in an organization in the context of data interoperability initiatives.

By using a Policy-Driven Ontological Analysis Model there is now a way of saving time, resources and effort during application interoperability initiatives. More research and investigative work is required to explore the additional resolution capabilities that would come from factoring Policy-Driven Ontology class properties and instances into the overall process. This, combined with alignment metrics, could improve the discriminatory powers of the model.

As organizations seek more and more to reuse existing data across multiple applications, data interoperability will become a strategic governance concern. Identifying and leveraging other organizations' interoperable datasets via ontologies can create wide and rich new marketplaces at low cost. The ability to exchange and reuse information in a meaningful way will makes possible the integration of various datasets and information into more complex, sophisticated and valuable enterprise wide applications.

Interoperability requires that the original concepts that are represented in each of the various databases are legitimately equivalent. Concepts that are used in the datasets can also be represented through ontologies. However, without being able to ensure conceptual equivalence across multiple datasets, the integrity of the datasets will be destroyed and the organizations will be making invalid assertions statements.

Contributions such as this research to the body of knowledge on ontology engineering and ontology management ensure that the critical semantic processes within an organization are not lost or masked by technology. Ultimately, the contribution to knowledge of the Policy-Driven Policy Analysis Model is the proof of a growing obligation to better understand the rise of semantic management in the organization.

Appendix A – Employee Ontology

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
  <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
  <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
  <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
  xmlns:rdf_="&rdf_;"
  xmlns:rdfs="&rdfs;">
  <rdf:Class rdf:about="&rdf_;Address"
    rdfs:label="Address">
    <rdfs:subClassOf rdf:resource="&rdf_;Customer_entity"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;Customer_entity"
    rdfs:label="Customer_entity">
    <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;Employee_number"
    rdfs:label="Employee_number">
    <rdfs:subClassOf rdf:resource="&rdf_;Customer_entity"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;First_Name"
    rdfs:label="First_Name">
    <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;Full_name"
    rdfs:label="Full_name">
    <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;Id_number"
    rdfs:label="Id_number">
    <rdfs:subClassOf rdf:resource="&rdf_;Customer_entity"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;Middle_name"
    rdfs:label="Middle_name">
    <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;Name"
    rdfs:label="Name">
    <rdfs:subClassOf rdf:resource="&rdf_;Customer_entity"/>
  </rdf:Class>
  <rdf:Class rdf:about="&rdf_;Start_date"
    rdfs:label="Start_date">
    <rdfs:subClassOf rdf:resource="&rdf_;Employee_number"/>
  </rdf:Class>
</rdf:RDF>
```

Appendix B – Customer Ontology

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
    <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
    xmlns:rdf_="&rdf_;"
    xmlns:rdfs="&rdfs;">
    <rdfs:Class rdf:about="&rdf_;Account_ID"
        rdfs:label="Account_ID">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Address"
        rdfs:label="Address">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Checking_account"
        rdfs:label="Checking_account">
        <rdfs:subClassOf rdf:resource="&rdf_;Account_ID"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Customer"
        rdfs:label="Customer">
        <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Customer_id"
        rdfs:label="Customer_id">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Deposit"
        rdfs:label="Deposit">
        <rdfs:subClassOf rdf:resource="&rdf_;Transaction"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;First_name"
        rdfs:label="First_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Middle_name"
        rdfs:label="Middle_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Name"
        rdfs:label="Name">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Saving_account"
        rdfs:label="Saving_account">
        <rdfs:subClassOf rdf:resource="&rdf_;Account_ID"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Surname"
        rdfs:label="Surname">
```

```
        <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Transaction"
  rdfs:label="Transaction">
  <rdfs:subClassOf rdf:resource="&rdf_;Account_ID"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Transfer"
  rdfs:label="Transfer">
  <rdfs:subClassOf rdf:resource="&rdf_;Transaction"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Withdrall"
  rdfs:label="Withdrall">
  <rdfs:subClassOf rdf:resource="&rdf_;Transaction"/>
</rdfs:Class>
</rdf:RDF>
```

Appendix C – Bank Branch Ontology

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
    <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
    xmlns:rdf_="&rdf_;"
    xmlns:rdfs="&rdfs;">
    <rdfs:Class rdf:about="&rdf_;Access_id"
        rdfs:label="Access_id">
        <rdfs:subClassOf rdf:resource="&rdf_;Role"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Accounts"
        rdfs:label="Accounts">
        <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Bank_branch"
        rdfs:label="Bank_branch">
        <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Branch_application"
        rdfs:label="Branch_application">
        <rdfs:subClassOf rdf:resource="&rdf_;Role"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Branch_id"
        rdfs:label="Branch_id">
        <rdfs:subClassOf rdf:resource="&rdf_;Bank_branch"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Business"
        rdfs:label="Business">
        <rdfs:subClassOf rdf:resource="&rdf_;Financial_Services"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Business_id"
        rdfs:label="Business_id">
        <rdfs:subClassOf rdf:resource="&rdf_;Business"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Chequing"
        rdfs:label="Chequing">
        <rdfs:subClassOf rdf:resource="&rdf_;Accounts"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Credit"
        rdfs:label="Credit">
        <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Customer"
        rdfs:label="Customer">
        <rdfs:subClassOf rdf:resource="&rdf_;Personal"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Customer_id"
        rdfs:label="Customer_id">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
    </rdfs:Class>
```



```

</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Employee"
  rdfs:label="Employee">
  <rdfs:subClassOf rdf:resource="&rdf_;Bank_branch"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Employee_id"
  rdfs:label="Employee_id">
  <rdfs:subClassOf rdf:resource="&rdf_;Employee"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Financial_Services"
  rdfs:label="Financial_Services">
  <rdfs:subClassOf rdf:resource="&rdf_;Bank_branch"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Investments"
  rdfs:label="Investments">
  <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Loans"
  rdfs:label="Loans">
  <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Mortgages"
  rdfs:label="Mortgages">
  <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Personal"
  rdfs:label="Personal">
  <rdfs:subClassOf rdf:resource="&rdf_;Financial_Services"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Products"
  rdfs:label="Products">
  <rdfs:subClassOf rdf:resource="&rdf_;Financial_Services"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Role"
  rdfs:label="Role">
  <rdfs:subClassOf rdf:resource="&rdf_;Employee_id"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Savings"
  rdfs:label="Savings">
  <rdfs:subClassOf rdf:resource="&rdf_;Accounts"/>
</rdfs:Class>
</rdf:RDF>

```

Appendix D : Financial Products and Services Ontology

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
    <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
    xmlns:rdf_="&rdf_;"
    xmlns:rdfs="&rdfs;">
    <rdfs:Class rdf:about="&rdf_;ABM"
        rdfs:label="ABM">
        <rdfs:subClassOf rdf:resource="&rdf_;Channels"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Accounts"
        rdfs:label="Accounts">
        <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Address"
        rdfs:label="Address">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Banch_banking"
        rdfs:label="Banch_banking">
        <rdfs:subClassOf rdf:resource="&rdf_;Channels"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Banking_Products_and_Services"
        rdfs:label="Banking_Products_and_Services">
        <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Businesses"
        rdfs:label="Businesses">
        <rdfs:subClassOf rdf:resource="&rdf_;Types"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Channels"
        rdfs:label="Channels">
        <rdfs:subClassOf rdf:resource="&rdf_;Banking_Products_and_Services"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Chequing"
        rdfs:label="Chequing">
        <rdfs:subClassOf rdf:resource="&rdf_;Accounts"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Credit_card"
        rdfs:label="Credit_card">
        <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_;Customer"
        rdfs:label="Customer">
        <rdfs:subClassOf rdf:resource="&rdf_;Banking_Products_and_Services"/>
    </rdfs:Class>
```

```

<rdfs:Class rdf:about="&rdf_;Debt_Consolidation"
  rdfs:label="Debt_Consolidation">
  <rdfs:subClassOf rdf:resource="&rdf_;Planning"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Direct_banking"
  rdfs:label="Direct_banking">
  <rdfs:subClassOf rdf:resource="&rdf_;Channels"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;ID_customer"
  rdfs:label="ID_customer">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Interest_Rates"
  rdfs:label="Interest_Rates">
  <rdfs:subClassOf rdf:resource="&rdf_;Services"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Joint_accounts"
  rdfs:label="Joint_accounts">
  <rdfs:subClassOf rdf:resource="&rdf_;Types"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Loans"
  rdfs:label="Loans">
  <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Mortgages"
  rdfs:label="Mortgages">
  <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Mutual_Funds"
  rdfs:label="Mutual_Funds">
  <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Name"
  rdfs:label="Name">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Planning"
  rdfs:label="Planning">
  <rdfs:subClassOf rdf:resource="&rdf_;Services"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Products"
  rdfs:label="Products">
  <rdfs:subClassOf rdf:resource="&rdf_;Banking_Products_and_Services"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;RRSP"
  rdfs:label="RRSP">
  <rdfs:subClassOf rdf:resource="&rdf_;Planning"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Savings"
  rdfs:label="Savings">
  <rdfs:subClassOf rdf:resource="&rdf_;Accounts"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Senior"
  rdfs:label="Senior">
  <rdfs:subClassOf rdf:resource="&rdf_;Types"/>
</rdfs:Class>

```

```

<rdfs:Class rdf:about="&rdf_;Services"
  rdfs:label="Services">
  <rdfs:subClassOf rdf:resource="&rdf_;Banking_Products_and_Services"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Students"
  rdfs:label="Students">
  <rdfs:subClassOf rdf:resource="&rdf_;Types"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Travel_Services"
  rdfs:label="Travel_Services">
  <rdfs:subClassOf rdf:resource="&rdf_;Services"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;Types"
  rdfs:label="Types">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;average_needs"
  rdfs:label="average_needs">
  <rdfs:subClassOf rdf:resource="&rdf_;Chequing"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;extensive_needs"
  rdfs:label="extensive_needs">
  <rdfs:subClassOf rdf:resource="&rdf_;Chequing"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;limited_needs"
  rdfs:label="limited_needs">
  <rdfs:subClassOf rdf:resource="&rdf_;Chequing"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;premium"
  rdfs:label="premium">
  <rdfs:subClassOf rdf:resource="&rdf_;Savings"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_;regular"
  rdfs:label="regular">
  <rdfs:subClassOf rdf:resource="&rdf_;Savings"/>
</rdfs:Class>
</rdf:RDF>

```

Appendix E: Enterprise PDO Ontology for Customer

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY a 'http://protege.stanford.edu/system#'>
    <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
    <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
    xmlns:rdf_="&rdf_;"
    xmlns:a="&a;"
    xmlns:rdfs="&rdfs;">
    <rdfs:Class rdf:about="&rdf_::OWL-ALL-DIFFERENT"
        rdfs:label=":OWL-ALL-DIFFERENT">
        <rdfs:subClassOf rdf:resource="&a;_system_class"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_::OWL-ALL-RESTRICTION"
        rdfs:label=":OWL-ALL-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-ANNOTATION-PROPERTY"
        rdfs:label=":OWL-ANNOTATION-PROPERTY">
        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-CLASS"
        rdfs:label=":OWL-ANONYMOUS-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-CLASS"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-ROOT"
        rdfs:label=":OWL-ANONYMOUS-ROOT">
        <rdfs:subClassOf rdf:resource="&a;_system_class"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_::OWL-CARDI-RESTRICTION"
        rdfs:label=":OWL-CARDI-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-MAXCARDI-RESTRICTION"/>
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-MINCARDI-RESTRICTION"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_::OWL-CLASS"
        rdfs:label=":OWL-CLASS">
        <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-CLASSIFICATION-STATUS"
        rdfs:label=":OWL-CLASSIFICATION-STATUS">
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-COMPLEMENT-CLASS"
        rdfs:label=":OWL-COMPLEMENT-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-DEPRECATED"
        rdfs:label=":OWL-DEPRECATED">
```

```

        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:domain rdf:resource="&rdfs;Class"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DIFFERENT-FROM"
        rdfs:label=":OWL-DIFFERENT-FROM">
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISJOINT-CLASSES"
        rdfs:label=":OWL-DISJOINT-CLASSES">
        <rdfs:domain rdf:resource="&rdf_::OWL-CLASS"/>
        <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISTINCT-MEMBERS"
        rdfs:label=":OWL-DISTINCT-MEMBERS">
        <rdfs:domain rdf:resource="&rdf_::OWL-ALL-DIFFERENT"/>
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-ENUMERATION-CLASS"
        rdfs:label=":OWL-ENUMERATION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-EQUIVALENT-SLOTS"
        rdfs:label=":OWL-EQUIVALENT-SLOTS">
        <rdfs:range rdf:resource="&rdf;Property"/>
        <rdfs:domain rdf:resource="&rdf;Property"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-FILLER-DEFINED"
        rdfs:label=":OWL-FILLER-DEFINED">
        <rdfs:domain rdf:resource="&rdf_::OWL-RESTRICTION"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-HAS-RESTRICTION"
        rdfs:label=":OWL-HAS-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUBCLASSES"
        rdfs:label=":OWL-INFERRED-SUBCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUPERCLASSES"
        rdfs:label=":OWL-INFERRED-SUPERCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-INTERSECTION-CLASS"
        rdfs:label=":OWL-INTERSECTION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INVERSE-FUNCTIONAL"
        rdfs:label=":OWL-INVERSE-FUNCTIONAL">
        <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-LOGICAL-CLASS"
        rdfs:label=":OWL-LOGICAL-CLASS">

```

```

        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-LOGICAL-OPERANDS"
        rdfs:label=":OWL-LOGICAL-OPERANDS">
        <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
        <rdfs:domain rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-MAXCARDI-RESTRICTION"
        rdfs:label=":OWL-MAXCARDI-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="&rdf_::OWL-MINCARDI-RESTRICTION"
        rdfs:label=":OWL-MINCARDI-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-IMPORTS"
        rdfs:label=":OWL-ONTOLOGY-IMPORTS">
        <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-PREFIXES"
        rdfs:label=":OWL-ONTOLOGY-PREFIXES">
        <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-URI"
        rdfs:label=":OWL-ONTOLOGY-URI">
        <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-RESOURCE-URI"
        rdfs:label=":OWL-RESOURCE-URI">
        <rdfs:domain rdf:resource="&rdf_::ExternalResource"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-RESTRICTION"
        rdfs:label=":OWL-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-SAME-AS"
        rdfs:label=":OWL-SAME-AS">
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME"
        rdfs:label=":OWL-SLOT-SOME">
        <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-CLASSES"
        rdfs:label=":OWL-SLOT-SOME-CLASSES">
        <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-VALUES"
        rdfs:label=":OWL-SLOT-SOME-VALUES">

```

```

        <rdfs:domain rdf:resource="&rdf_::OWL-SOME-RESTRICTION"/>
        <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-SOME-RESTRICTION"
        rdfs:label=":OWL-SOME-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-SYMMETRIC"
        rdfs:label=":OWL-SYMMETRIC">
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-TRANSITIVE"
        rdfs:label=":OWL-TRANSITIVE">
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-UNION-CLASS"
        rdfs:label=":OWL-UNION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-XML-SCHEMA-DATATYPE"
        rdfs:label=":OWL-XML-SCHEMA-DATATYPE">
        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdf_::Account_balance"
        rdfs:label="Account_balance">
        <rdfs:subClassOf rdf:resource="&rdf_::Customer_holdings_summary"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Age"
        rdfs:label="Age">
        <rdfs:subClassOf rdf:resource="&rdf_::Date_of_birth"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Annual_income"
        rdfs:label="Annual_income">
        <rdfs:subClassOf rdf:resource="&rdf_::Customer"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Area_code"
        rdfs:label="Area_code">
        <rdfs:subClassOf rdf:resource="&rdf_::Telephone_number"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::BIN"
        rdfs:label="BIN">
        <rdfs:subClassOf rdf:resource="&rdf_::Business_identification_number"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::BN"
        rdfs:label="BN">
        <rdfs:subClassOf rdf:resource="&rdf_::Business_number"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>

```



```

</rdf:Description>
<rdf:Description rdf:about="&rdf_;Banking_decision_maker"
  rdfs:label="Banking_decision_maker">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Business"
  rdfs:label="Business">
  <rdfs:subClassOf rdf:resource="&rdf_;Classification"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Business_identification_number"
  rdfs:label="Business_identification_number">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Business_number"
  rdfs:label="Business_number">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Business_to_business"
  rdfs:label="Business_to_business">
  <rdfs:subClassOf rdf:resource="&rdf_;Relationship"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Canadian_province_Territory_US_state"
  rdfs:label="Canadian_province_Territory_US_state">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Civic_number"
  rdfs:label="Civic_number">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Civic_number_suffix"
  rdfs:label="Civic_number_suffix">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Classification"
  rdfs:label="Classification">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdfs:subClassOf rdf:resource="&rdf_;Effective_Date"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Common_law"
  rdfs:label="Common_law">
  <rdfs:subClassOf rdf:resource="&rdf_;Marital_status"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Country"
  rdfs:label="Country">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>

```

```

</rdf:Description>
<rdf:Description rdf:about="&rdf_;Currency"
  rdfs:label="Currency">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer_holdings_summary"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Currency_code"
  rdfs:label="Currency_code">
  <rdfs:subClassOf rdf:resource="&rdf_;Annual_income"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Customer"
  rdfs:label="Customer">
  <rdfs:subClassOf rdf:resource="&rdf_;PDO_customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Customer_ID"
  rdfs:label="Customer_ID">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Customer_holdings"
  rdfs:label="Customer_holdings">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer_holdings_summary"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Customer_holdings_summary"
  rdfs:label="Customer_holdings_summary">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdfs:subClassOf rdf:resource="&rdf_;Effective_Date"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Date"
  rdfs:label="Date">
  <rdfs:subClassOf rdf:resource="&rdf_;Date_of_birth"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Date_customer_commenced"
  rdfs:label="Date_customer_commenced">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Date_of_birth"
  rdfs:label="Date_of_birth">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Date_of_death"
  rdfs:label="Date_of_death">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Date_of_registration"
  rdfs:label="Date_of_registration">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>

```

```

</rdf:Description>
<rdf:Description rdf:about="&rdf_;Designation"
  rdfs:label="Designation">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Divorced"
  rdfs:label="Divorced">
  <rdfs:subClassOf rdf:resource="&rdf_;Marital_status"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;E_mail"
  rdfs:label="E_mail">
  <rdfs:subClassOf rdf:resource="&rdf_;Email"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Effective_Date"
  rdfs:label="Effective_Date">
  <rdfs:subClassOf rdf:resource="&rdf_;PDO_customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Email"
  rdfs:label="Email">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdfs:subClassOf rdf:resource="&rdf_;Effective_Date"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;End_date"
  rdfs:label="End_date">
  <rdfs:subClassOf rdf:resource="&rdf_;Relationship"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;English"
  rdfs:label="English">
  <rdfs:subClassOf rdf:resource="&rdf_;Official_language"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Extension"
  rdfs:label="Extension">
  <rdfs:subClassOf rdf:resource="&rdf_;Telephone_number"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Class rdf:about="&rdf_;ExternalResource"
  rdfs:label="ExternalResource">
  <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
</rdf:Class>
<rdf:Description rdf:about="&rdf_;Familial"
  rdfs:label="Familial">
  <rdfs:subClassOf rdf:resource="&rdf_;Relationship"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Female"
  rdfs:label="Female">
  <rdfs:subClassOf rdf:resource="&rdf_;Gender"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>

```

```

<rdf:Description rdf:about="&rdf_;First_name"
  rdfs:label="First_name">
  <rdfs:subClassOf rdf:resource="&rdf_;Personal_customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;French"
  rdfs:label="French">
  <rdfs:subClassOf rdf:resource="&rdf_;Official_language"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Full_address"
  rdfs:label="Full_address">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdfs:subClassOf rdf:resource="&rdf_;Effective_Date"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Functional_title"
  rdfs:label="Functional_title">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Gender"
  rdfs:label="Gender">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Group_Departement_Division_name"
  rdfs:label="Group_Departement_Division_name">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;ID"
  rdfs:label="ID">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer_ID"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;International_number"
  rdfs:label="International_number">
  <rdfs:subClassOf rdf:resource="&rdf_;Telephone_number"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Language"
  rdfs:label="Language">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Language_of_choice"
  rdfs:label="Language_of_choice">
  <rdfs:comment>English = en
French = r
Chinese = zh
Not provided = np</rdfs:comment>
  <rdfs:subClassOf rdf:resource="&rdf_;Preferred_language"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Last_name"

```

```

        rdfs:label="Last_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Personal_customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Legal_name"
        rdfs:comment="Business customer"
        rdfs:label="Legal_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
        <rdfs:subClassOf rdf:resource="&rdf_;Personal_customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Male"
        rdfs:label="Male">
        <rdfs:subClassOf rdf:resource="&rdf_;Gender"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Marital_status"
        rdfs:label="Marital_status">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Married"
        rdfs:label="Married">
        <rdfs:subClassOf rdf:resource="&rdf_;Marital_status"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Middle_name"
        rdfs:label="Middle_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Personal_customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Mixed"
        rdfs:label="Mixed">
        <rdfs:subClassOf rdf:resource="&rdf_;Classification"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Municipality_Town_City"
        rdfs:label="Municipality_Town_City">
        <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;NAICS"
        rdfs:comment="North American Industry Classification System"
        rdfs:label="NAICS">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Name_prefix"
        rdfs:label="Name_prefix">
        <rdfs:subClassOf rdf:resource="&rdf_;Personal_customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Name_qualifier"
        rdfs:label="Name_qualifier">
        <rdfs:subClassOf rdf:resource="&rdf_;Personal_customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>

```

```

</rdf:Description>
<rdf:Description rdf:about="&rdf_;Not_provided"
  rdfs:label="Not_provided">
  <rdfs:subClassOf rdf:resource="&rdf_;Gender"/>
  <rdfs:subClassOf rdf:resource="&rdf_;Marital_status"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Number"
  rdfs:label="Number">
  <rdfs:subClassOf rdf:resource="&rdf_;Telephone_number"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Official_language"
  rdfs:label="Official_language">
  <rdfs:subClassOf rdf:resource="&rdf_;Language"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Other_income"
  rdfs:label="Other_income">
  <rdfs:subClassOf rdf:resource="&rdf_;Annual_income"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;PDO_customer"
  rdfs:label="PDO_customer">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Permission_to_market"
  rdfs:label="Permission_to_market">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Permission_to_share_information"
  rdfs:label="Permission_to_share_information">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Persmission_for_the_use_of_SIN"
  rdfs:label="Persmission_for_the_use_of_SIN">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Person_to_business"
  rdfs:label="Person_to_business">
  <rdfs:subClassOf rdf:resource="&rdf_;Relationship"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Person_to_person"
  rdfs:label="Person_to_person">
  <rdfs:subClassOf rdf:resource="&rdf_;Relationship"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Personal"
  rdfs:label="Personal">
  <rdfs:subClassOf rdf:resource="&rdf_;Classification"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>

```

```

</rdf:Description>
<rdf:Description rdf:about="&rdf_;Personal_customer"
  rdfs:label="Personal_customer">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Postal_code"
  rdfs:label="Postal_code">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Prefered_contact_name"
  rdfs:label="Prefered_contact_name">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Prefered_frequency_of_contact"
  rdfs:label="Prefered_frequency_of_contact">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Prefered_language"
  rdfs:label="Prefered_language">
  <rdfs:subClassOf rdf:resource="&rdf_;Language"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Prefered_method_of_contact"
  rdfs:label="Prefered_method_of_contact">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Prefered_time_of_contact"
  rdfs:label="Prefered_time_of_contact">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Preferences"
  rdfs:label="Preferences">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Preferred_channel"
  rdfs:label="Preferred_channel">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Primary_contact_role_preference"
  rdfs:label="Primary_contact_role_preference">
  <rdfs:subClassOf rdf:resource="&rdf_;Preferences"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Primary_income"
  rdfs:label="Primary_income">
  <rdfs:subClassOf rdf:resource="&rdf_;Annual_income"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>

```

```

<rdf:Description rdf:about="&rdf_;RMC"
  rdfs:label="RMC">
  <rdfs:subClassOf rdf:resource="&rdf_;Relationship_Manager_Code"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Relationship"
  rdfs:comment="to other bank customers for Joint Accounts"
  rdfs:label="Relationship">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdfs:subClassOf rdf:resource="&rdf_;Effective_Date"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Relationship_Manager_Code"
  rdfs:label="Relationship_Manager_Code">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdfs:subClassOf rdf:resource="&rdf_;Effective_Date"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Rural_route_Post_office_box"
  rdfs:label="Rural_route_Post_office_box">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;SIN"
  rdfs:label="SIN">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;SSN"
  rdfs:label="SSN">
  <rdfs:subClassOf rdf:resource="&rdf_;Social_Security_Number"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Separated"
  rdfs:label="Separated">
  <rdfs:subClassOf rdf:resource="&rdf_;Marital_status"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Single"
  rdfs:label="Single">
  <rdfs:subClassOf rdf:resource="&rdf_;Marital_status"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Social_Security_Number"
  rdfs:label="Social_Security_Number">
  <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Street_direction"
  rdfs:label="Street_direction">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Street_name"
  rdfs:label="Street_name">
  <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>

```



```

        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Telephone_number"
        rdfs:label="Telephone_number">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer"/>
        <rdfs:subClassOf rdf:resource="&rdf_;Effective_Date"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Temporary_address_start_date"
        rdfs:label="Temporary_address_start_date">
        <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Temporary_address_stop_date"
        rdfs:label="Temporary_address_stop_date">
        <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Trade_name"
        rdfs:label="Trade_name">
        <rdfs:comment>Business customer
    </rdfs:comment>
        <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
        <rdfs:subClassOf rdf:resource="&rdf_;Personal_customer"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Transit_Account_number"
        rdfs:label="Transit_Account_number">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer_holdings_summary"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;US_zip_code"
        rdfs:label="US_zip_code">
        <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Unit_desingation"
        rdfs:label="Unit_desingation">
        <rdfs:subClassOf rdf:resource="&rdf_;Full_address"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Widowed"
        rdfs:label="Widowed">
        <rdfs:subClassOf rdf:resource="&rdf_;Marital_status"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:Class"
        rdfs:label="owl:Class">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
    </rdf:Description>
    <rdfs:Class rdf:about="&rdf_;owl:DataRange"
        rdfs:label="owl:DataRange">
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdfs:Class>
    <rdf:Description rdf:about="&rdf_;owl:DatatypeProperty"

```

```

        rdfs:label="owl:DatatypeProperty">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&rdf;Property"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:Nothing"
        rdfs:label="owl:Nothing">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:ObjectProperty"
        rdfs:label="owl:ObjectProperty">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&rdf;Property"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:Ontology"
        rdfs:label="owl:Ontology">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&a;_system_class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:backwardCompatibleWith"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:backwardCompatibleWith">
        <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:incompatibleWith"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:incompatibleWith">
        <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:priorVersion"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:priorVersion">
        <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:versionInfo"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:versionInfo">
        <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Property rdf:about="&a;_direct_template_slots"
        rdfs:label=":DIRECT-TEMPLATE-SLOTS"/>
    <rdf:Description rdf:about="&rdf;first"
        rdfs:label="rdf:first">
        <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
        <rdfs:domain rdf:resource="&rdf;List"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Description>
    <rdf:Property rdf:about="&rdf;object"
        rdfs:label="rdf:object">

```

```

        <rdfs:domain rdf:resource="&rdf;Statement"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf;predicate"
        rdfs:label="rdf:predicate">
        <rdfs:domain rdf:resource="&rdf;Statement"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdf;rest"
        rdfs:label="rdf:rest">
        <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
        <rdfs:domain rdf:resource="&rdf;List"/>
        <rdfs:range rdf:resource="&rdf;List"/>
    </rdf:Description>
    <rdf:Property rdf:about="&rdf;subject"
        rdfs:label="rdf:subject">
        <rdfs:domain rdf:resource="&rdf;Statement"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf;value"
        rdfs:label="rdf:value">
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdfs;comment"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:comment">
        <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdfs;isDefinedBy"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:isDefinedBy">
        <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdfs;label"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:label">
        <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Property rdf:about="&rdfs;member"
        rdfs:label="rdfs:member">
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdfs;seeAlso"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:seeAlso">
        <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Description>
</rdf:RDF>

```

Appendix F: Application A Involved_Party

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY a 'http://protege.stanford.edu/system#'>
    <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
    <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
    xmlns:rdf_="&rdf_;"
    xmlns:a="&a;"
    xmlns:rdfs="&rdfs;">
  <rdfs:Class rdf:about="&rdf_::OWL-ALL-DIFFERENT"
    rdfs:label=":OWL-ALL-DIFFERENT">
    <rdfs:subClassOf rdf:resource="&a;_system_class"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-ALL-RESTRICTION"
    rdfs:label=":OWL-ALL-RESTRICTION">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-ANNOTATION-PROPERTY"
    rdfs:label=":OWL-ANNOTATION-PROPERTY">
    <rdfs:domain rdf:resource="&rdf;Property"/>
    <rdfs:range rdf:resource="&rdfs;Literal"/>
  </rdf:Property>
  <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-CLASS"
    rdfs:label=":OWL-ANONYMOUS-CLASS">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-CLASS"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-ROOT"
    rdfs:label=":OWL-ANONYMOUS-ROOT">
    <rdfs:subClassOf rdf:resource="&a;_system_class"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-CARDI-RESTRICTION"
    rdfs:label=":OWL-CARDI-RESTRICTION">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-MAXCARDI-RESTRICTION"/>
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-MINCARDI-RESTRICTION"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-CLASS"
    rdfs:label=":OWL-CLASS">
    <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-CLASSIFICATION-STATUS"
    rdfs:label=":OWL-CLASSIFICATION-STATUS">
    <rdfs:range rdf:resource="&rdfs;Literal"/>
  </rdf:Property>
  <rdfs:Class rdf:about="&rdf_::OWL-COMPLEMENT-CLASS"
    rdfs:label=":OWL-COMPLEMENT-CLASS">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-DEPRECATED"
    rdfs:label=":OWL-DEPRECATED">
    <rdfs:domain rdf:resource="&rdf;Property"/>
    <rdfs:domain rdf:resource="&rdfs;Class"/>
  </rdf:Property>
</rdf:RDF>
```

```

        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DIFFERENT-FROM"
        rdfs:label=":OWL-DIFFERENT-FROM">
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISJOINT-CLASSES"
        rdfs:label=":OWL-DISJOINT-CLASSES">
        <rdfs:domain rdf:resource="&rdf_::OWL-CLASS"/>
        <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISTINCT-MEMBERS"
        rdfs:label=":OWL-DISTINCT-MEMBERS">
        <rdfs:domain rdf:resource="&rdf_::OWL-ALL-DIFFERENT"/>
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-ENUMERATION-CLASS"
        rdfs:label=":OWL-ENUMERATION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-EQUIVALENT-SLOTS"
        rdfs:label=":OWL-EQUIVALENT-SLOTS">
        <rdfs:range rdf:resource="&rdf;Property"/>
        <rdfs:domain rdf:resource="&rdf;Property"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-FILLER-DEFINED"
        rdfs:label=":OWL-FILLER-DEFINED">
        <rdfs:domain rdf:resource="&rdf_::OWL-RESTRICTION"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-HAS-RESTRICTION"
        rdfs:label=":OWL-HAS-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUBCLASSES"
        rdfs:label=":OWL-INFERRED-SUBCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUPERCLASSES"
        rdfs:label=":OWL-INFERRED-SUPERCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-INTERSECTION-CLASS"
        rdfs:label=":OWL-INTERSECTION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INVERSE-FUNCTIONAL"
        rdfs:label=":OWL-INVERSE-FUNCTIONAL">
        <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-LOGICAL-CLASS"
        rdfs:label=":OWL-LOGICAL-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>

```

```

<rdf:Property rdf:about="&rdf_::OWL-LOGICAL-OPERANDS"
  rdfs:label=":OWL-LOGICAL-OPERANDS">
  <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
  <rdfs:domain rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
</rdf:Property>
<rdfs:Class rdf:about="&rdf_::OWL-MAXCARDI-RESTRICTION"
  rdfs:label=":OWL-MAXCARDI-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_::OWL-MINCARDI-RESTRICTION"
  rdfs:label=":OWL-MINCARDI-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
</rdfs:Class>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-IMPORTS"
  rdfs:label=":OWL-ONTOLOGY-IMPORTS">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-PREFIXES"
  rdfs:label=":OWL-ONTOLOGY-PREFIXES">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-URI"
  rdfs:label=":OWL-ONTOLOGY-URI">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-RESOURCE-URI"
  rdfs:label=":OWL-RESOURCE-URI">
  <rdfs:domain rdf:resource="&rdf_::ExternalResource"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdfs:Class rdf:about="&rdf_::OWL-RESTRICTION"
  rdfs:label=":OWL-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
</rdfs:Class>
<rdf:Property rdf:about="&rdf_::OWL-SAME-AS"
  rdfs:label=":OWL-SAME-AS">
  <rdfs:range rdf:resource="&a;owl_thing"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME"
  rdfs:label=":OWL-SLOT-SOME">
  <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
  <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-CLASSES"
  rdfs:label=":OWL-SLOT-SOME-CLASSES">
  <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
  <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-VALUES"
  rdfs:label=":OWL-SLOT-SOME-VALUES">
  <rdfs:domain rdf:resource="&rdf_::OWL-SOME-RESTRICTION"/>
  <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>

```

```

        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-SOME-RESTRICTION"
        rdfs:label=":OWL-SOME-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-SYMMETRIC"
        rdfs:label=":OWL-SYMMETRIC">
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-TRANSITIVE"
        rdfs:label=":OWL-TRANSITIVE">
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-UNION-CLASS"
        rdfs:label=":OWL-UNION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-XML-SCHEMA-DATATYPE"
        rdfs:label=":OWL-XML-SCHEMA-DATATYPE">
        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdf_::Address"
        rdfs:label="Address">
        <rdfs:subClassOf rdf:resource="&rdf_::Contact"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Age"
        rdfs:label="Age">
        <rdfs:subClassOf rdf:resource="&rdf_::Individual"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Annual_income_amount"
        rdfs:label="Annual_income_amount">
        <rdfs:subClassOf rdf:resource="&rdf_::Involved_party"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Banking_decision_maker"
        rdfs:label="Banking_decision_maker">
        <rdfs:subClassOf rdf:resource="&rdf_::Customer_role"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Birth_date"
        rdfs:label="Birth_date">
        <rdfs:subClassOf rdf:resource="&rdf_::Individual"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Branch"
        rdfs:label="Branch">
        <rdfs:subClassOf rdf:resource="&rdf_::Role"/>
        <rdf:type rdf:resource="&rdf_::owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_::Business_identification_code"

```

```

        rdfs:label="Business_identification_code">
        <rdfs:subClassOf rdf:resource="&rdf_;Organization"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Business_name"
        rdfs:label="Business_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Organization"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Contact"
        rdfs:label="Contact">
        <rdfs:subClassOf rdf:resource="&rdf_;Locator"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Currency_code"
        rdfs:label="Currency_code">
        <rdfs:subClassOf rdf:resource="&rdf_;Involved_party"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Customer_role"
        rdfs:label="Customer_role">
        <rdfs:subClassOf rdf:resource="&rdf_;Role"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Customer_since"
        rdfs:label="Customer_since">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer_role"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Customer_status"
        rdfs:label="Customer_status">
        <rdfs:subClassOf rdf:resource="&rdf_;Customer_role"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Deceased_date"
        rdfs:label="Deceased_date">
        <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Departement"
        rdfs:label="Departement">
        <rdfs:subClassOf rdf:resource="&rdf_;Employee"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Designation"
        rdfs:label="Designation">
        <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Email"
        rdfs:label="Email">
        <rdfs:subClassOf rdf:resource="&rdf_;Contact"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Employee"
        rdfs:label="Employee">

```



```

        <rdfs:subClassOf rdf:resource="&rdf_;Role"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdfs:Class rdf:about="&rdf_;ExternalResource"
        rdfs:label="ExternalResource">
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdfs:Class>
    <rdf:Description rdf:about="&rdf_;Functional_title"
        rdfs:label="Functional_title">
        <rdfs:subClassOf rdf:resource="&rdf_;Employee"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Gender"
        rdfs:label="Gender">
        <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Given_name"
        rdfs:label="Given_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Group_name"
        rdfs:label="Group_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Employee"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Hours"
        rdfs:label="Hours">
        <rdfs:subClassOf rdf:resource="&rdf_;Locator"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Household"
        rdfs:label="Household">
        <rdfs:subClassOf rdf:resource="&rdf_;Role"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Individual"
        rdfs:label="Individual">
        <rdfs:subClassOf rdf:resource="&rdf_;Type_code"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Industry_sector_code"
        rdfs:label="Industry_sector_code">
        <rdfs:subClassOf rdf:resource="&rdf_;Organization"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Involved_party"
        rdfs:label="Involved_party">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Language"
        rdfs:label="Language">
        <rdfs:subClassOf rdf:resource="&rdf_;Involved_party"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>

```

```

</rdf:Description>
<rdf:Description rdf:about="&rdf_;Legal_name"
  rdfs:label="Legal_name">
  <rdfs:subClassOf rdf:resource="&rdf_;Business_name"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Locator"
  rdfs:label="Locator">
  <rdfs:subClassOf rdf:resource="&rdf_;Involved_party"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Middle_initial"
  rdfs:label="Middle_initial">
  <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;NAICS"
  rdfs:label="NAICS">
  <rdfs:subClassOf rdf:resource="&rdf_;Organization"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Name"
  rdfs:label="Name">
  <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Organization"
  rdfs:label="Organization">
  <rdfs:subClassOf rdf:resource="&rdf_;Type_code"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Preferred_contact"
  rdfs:label="Preferred_contact">
  <rdfs:subClassOf rdf:resource="&rdf_;Role"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Prefix"
  rdfs:label="Prefix">
  <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Prospect"
  rdfs:label="Prospect">
  <rdfs:subClassOf rdf:resource="&rdf_;Role"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Registration_date"
  rdfs:label="Registration_date">
  <rdfs:subClassOf rdf:resource="&rdf_;Organization"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Relationship_manager_code"
  rdfs:label="Relationship_manager_code">
  <rdfs:subClassOf rdf:resource="&rdf_;Employee"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>

```

```

<rdf:Description rdf:about="&rdf_;Role"
  rdfs:label="Role">
  <rdfs:subClassOf rdf:resource="&rdf_;Involved_party"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;SIN"
  rdfs:label="SIN">
  <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Sharing_indicator"
  rdfs:label="Sharing_indicator">
  <rdfs:subClassOf rdf:resource="&rdf_;Involved_party"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Telephone"
  rdfs:label="Telephone">
  <rdfs:subClassOf rdf:resource="&rdf_;Contact"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Trade_name"
  rdfs:label="Trade_name">
  <rdfs:subClassOf rdf:resource="&rdf_;Business_name"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Type_code"
  rdfs:label="Type_code">
  <rdfs:subClassOf rdf:resource="&rdf_;Involved_party"/>
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Class"
  rdfs:label="owl:Class">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
</rdf:Description>
<rdfs:Class rdf:about="&rdf_;owl:DataRange"
  rdfs:label="owl:DataRange">
  <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
</rdfs:Class>
<rdf:Description rdf:about="&rdf_;owl:DatatypeProperty"
  rdfs:label="owl:DatatypeProperty">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&rdf;Property"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Nothing"
  rdfs:label="owl:Nothing">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:ObjectProperty"
  rdfs:label="owl:ObjectProperty">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&rdf;Property"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Ontology"
  rdfs:label="owl:Ontology">

```

```

        <rdf:type rdf:resource="&rdf_owl:Class"/>
        <rdfs:subClassOf rdf:resource="&a;_system_class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_owl:backwardCompatibleWith"
        rdf::OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:backwardCompatibleWith">
        <rdf:type rdf:resource="&rdf_owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfsLiteral"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_owl:incompatibleWith"
        rdf::OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:incompatibleWith">
        <rdf:type rdf:resource="&rdf_owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfsLiteral"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_owl:priorVersion"
        rdf::OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:priorVersion">
        <rdf:type rdf:resource="&rdf_owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_owl:Ontology"/>
        <rdfs:range rdf:resource="&rdfsLiteral"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_owl:versionInfo"
        rdf::OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:versionInfo">
        <rdf:type rdf:resource="&rdf_owl:DatatypeProperty"/>
        <rdfs:range rdf:resource="&rdfsLiteral"/>
    </rdf:Description>
    <rdf:Property rdf:about="&a;_direct_template_slots"
        rdfs:label="DIRECT-TEMPLATE-SLOTS"/>
    <rdf:Description rdf:about="&rdf_first"
        rdfs:label="rdf:first">
        <rdf:type rdf:resource="&rdf_owl:ObjectProperty"/>
        <rdfs:domain rdf:resource="&rdf_List"/>
        <rdfs:range rdf:resource="&rdfsResource"/>
    </rdf:Description>
    <rdf:Property rdf:about="&rdf_object"
        rdfs:label="rdf:object">
        <rdfs:domain rdf:resource="&rdf_Statement"/>
        <rdfs:range rdf:resource="&rdfsResource"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_predicate"
        rdfs:label="rdf:predicate">
        <rdfs:domain rdf:resource="&rdf_Statement"/>
        <rdfs:range rdf:resource="&rdfsResource"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdf_rest"
        rdfs:label="rdf:rest">
        <rdf:type rdf:resource="&rdf_owl:ObjectProperty"/>
        <rdfs:domain rdf:resource="&rdf_List"/>
        <rdfs:range rdf:resource="&rdf_List"/>
    </rdf:Description>
    <rdf:Property rdf:about="&rdf_subject"
        rdfs:label="rdf:subject">

```

```

        <rdfs:domain rdf:resource="&rdfs;Statement"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdfs;value"
        rdfs:label="rdfs:value">
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdfs;comment"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:comment">
        <rdf:type rdf:resource="&rdfs;owl:DatatypeProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdfs;isDefinedBy"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:isDefinedBy">
        <rdf:type rdf:resource="&rdfs;owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdfs;label"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:label">
        <rdf:type rdf:resource="&rdfs;owl:DatatypeProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Description>
    <rdf:Property rdf:about="&rdfs;member"
        rdfs:label="rdfs:member">
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdfs;seeAlso"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:seeAlso">
        <rdf:type rdf:resource="&rdfs;owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Resource"/>
    </rdf:Description>
</rdf:RDF>

```

Appendix G: Application B - Individual

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
  <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
  <!ENTITY a 'http://protege.stanford.edu/system#'>
  <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
  <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
  xmlns:rdf_="&rdf_;"
  xmlns:a="&a;"
  xmlns:rdfs="&rdfs;">
  <rdfs:Class rdf:about="&rdf_::OWL-ALL-DIFFERENT"
    rdfs:label=":OWL-ALL-DIFFERENT">
    <rdfs:subClassOf rdf:resource="&a;_system_class"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-ALL-RESTRICTION"
    rdfs:label=":OWL-ALL-RESTRICTION">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-ANNOTATION-PROPERTY"
    rdfs:label=":OWL-ANNOTATION-PROPERTY">
    <rdfs:domain rdf:resource="&rdf;Property"/>
    <rdfs:range rdf:resource="&rdfs;Literal"/>
  </rdf:Property>
  <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-CLASS"
    rdfs:label=":OWL-ANONYMOUS-CLASS">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-CLASS"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-ROOT"
    rdfs:label=":OWL-ANONYMOUS-ROOT">
    <rdfs:subClassOf rdf:resource="&a;_system_class"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-CARDI-RESTRICTION"
    rdfs:label=":OWL-CARDI-RESTRICTION">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-MAXCARDI-RESTRICTION"/>
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-MINCARDI-RESTRICTION"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-CLASS"
    rdfs:label=":OWL-CLASS">
    <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-CLASSIFICATION-STATUS"
    rdfs:label=":OWL-CLASSIFICATION-STATUS">
    <rdfs:range rdf:resource="&rdfs;Literal"/>
  </rdf:Property>
  <rdfs:Class rdf:about="&rdf_::OWL-COMPLEMENT-CLASS"
    rdfs:label=":OWL-COMPLEMENT-CLASS">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-DEPRECATED"
    rdfs:label=":OWL-DEPRECATED">
    <rdfs:domain rdf:resource="&rdf;Property"/>
    <rdfs:domain rdf:resource="&rdfs;Class"/>
```

```

        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DIFFERENT-FROM"
        rdfs:label=":OWL-DIFFERENT-FROM">
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISJOINT-CLASSES"
        rdfs:label=":OWL-DISJOINT-CLASSES">
        <rdfs:domain rdf:resource="&rdf_::OWL-CLASS"/>
        <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISTINCT-MEMBERS"
        rdfs:label=":OWL-DISTINCT-MEMBERS">
        <rdfs:domain rdf:resource="&rdf_::OWL-ALL-DIFFERENT"/>
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-ENUMERATION-CLASS"
        rdfs:label=":OWL-ENUMERATION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-EQUIVALENT-SLOTS"
        rdfs:label=":OWL-EQUIVALENT-SLOTS">
        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:range rdf:resource="&rdf;Property"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-FILLER-DEFINED"
        rdfs:label=":OWL-FILLER-DEFINED">
        <rdfs:domain rdf:resource="&rdf_::OWL-RESTRICTION"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-HAS-RESTRICTION"
        rdfs:label=":OWL-HAS-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUBCLASSES"
        rdfs:label=":OWL-INFERRED-SUBCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUPERCLASSES"
        rdfs:label=":OWL-INFERRED-SUPERCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-INTERSECTION-CLASS"
        rdfs:label=":OWL-INTERSECTION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INVERSE-FUNCTIONAL"
        rdfs:label=":OWL-INVERSE-FUNCTIONAL">
        <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-LOGICAL-CLASS"
        rdfs:label=":OWL-LOGICAL-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>

```

```

<rdf:Property rdf:about="&rdf_::OWL-LOGICAL-OPERANDS"
  rdfs:label=":OWL-LOGICAL-OPERANDS">
  <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
  <rdfs:domain rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
</rdf:Property>
<rdfs:Class rdf:about="&rdf_::OWL-MAXCARDI-RESTRICTION"
  rdfs:label=":OWL-MAXCARDI-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_::OWL-MINCARDI-RESTRICTION"
  rdfs:label=":OWL-MINCARDI-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
</rdfs:Class>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-IMPORTS"
  rdfs:label=":OWL-ONTOLOGY-IMPORTS">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-PREFIXES"
  rdfs:label=":OWL-ONTOLOGY-PREFIXES">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-URI"
  rdfs:label=":OWL-ONTOLOGY-URI">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-RESOURCE-URI"
  rdfs:label=":OWL-RESOURCE-URI">
  <rdfs:domain rdf:resource="&rdf_::ExternalResource"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdfs:Class rdf:about="&rdf_::OWL-RESTRICTION"
  rdfs:label=":OWL-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
</rdfs:Class>
<rdf:Property rdf:about="&rdf_::OWL-SAME-AS"
  rdfs:label=":OWL-SAME-AS">
  <rdfs:range rdf:resource="&a;owl_thing"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME"
  rdfs:label=":OWL-SLOT-SOME">
  <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
  <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-CLASSES"
  rdfs:label=":OWL-SLOT-SOME-CLASSES">
  <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
  <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-VALUES"
  rdfs:label=":OWL-SLOT-SOME-VALUES">
  <rdfs:domain rdf:resource="&rdf_::OWL-SOME-RESTRICTION"/>
  <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>

```



```

        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-SOME-RESTRICTION"
        rdfs:label=":OWL-SOME-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-SYMMETRIC"
        rdfs:label=":OWL-SYMMETRIC">
        <rdfs:domain rdf:resource="&rdf_owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-TRANSITIVE"
        rdfs:label=":OWL-TRANSITIVE">
        <rdfs:domain rdf:resource="&rdf_owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-UNION-CLASS"
        rdfs:label=":OWL-UNION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-XML-SCHEMA-DATATYPE"
        rdfs:label=":OWL-XML-SCHEMA-DATATYPE">
        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdf_;Address"
        rdfs:label="Address">
        <rdfs:subClassOf rdf:resource="&rdf_;Contact"/>
        <rdf:type rdf:resource="&rdf_owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Bonds"
        rdfs:label="Bonds">
        <rdfs:subClassOf rdf:resource="&rdf_;Investements"/>
        <rdf:type rdf:resource="&rdf_owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Contact"
        rdfs:label="Contact">
        <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
        <rdf:type rdf:resource="&rdf_owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Email"
        rdfs:label="Email">
        <rdfs:subClassOf rdf:resource="&rdf_;Contact"/>
        <rdf:type rdf:resource="&rdf_owl:Class"/>
    </rdf:Description>
    <rdfs:Class rdf:about="&rdf_;ExternalResource"
        rdfs:label="ExternalResource">
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdfs:Class>
    <rdf:Description rdf:about="&rdf_;First"
        rdfs:label="First">
        <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
        <rdf:type rdf:resource="&rdf_owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;ID"
        rdfs:label="ID">

```

```

        <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Income"
        rdfs:label="Income">
        <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Individual"
        rdfs:label="Individual">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Investments"
        rdfs:label="Investments">
        <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Loans"
        rdfs:label="Loans">
        <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Middle"
        rdfs:label="Middle">
        <rdfs:subClassOf rdf:resource="&rdf_;Name"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Mortgage"
        rdfs:label="Mortgage">
        <rdfs:subClassOf rdf:resource="&rdf_;Products"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Mutual_funds"
        rdfs:label="Mutual_funds">
        <rdfs:subClassOf rdf:resource="&rdf_;Investments"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Name"
        rdfs:label="Name">
        <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Others"
        rdfs:label="Others">
        <rdfs:subClassOf rdf:resource="&rdf_;Investments"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Products"
        rdfs:label="Products">
        <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Stocks"
        rdfs:label="Stocks">
        <rdfs:subClassOf rdf:resource="&rdf_;Investments"/>

```

```

        <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Tax_info"
    rdfs:label="Tax_info">
    <rdfs:subClassOf rdf:resource="&rdf_;Individual"/>
    <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;Telephone"
    rdfs:label="Telephone">
    <rdfs:subClassOf rdf:resource="&rdf_;Contact"/>
    <rdf:type rdf:resource="&rdf_;owl:Class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Class"
    rdfs:label="owl:Class">
    <rdf:type rdf:resource="&rdf_;owl:Class"/>
    <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
</rdf:Description>
<rdfs:Class rdf:about="&rdf_;owl:DataRange"
    rdfs:label="owl:DataRange">
    <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
</rdfs:Class>
<rdf:Description rdf:about="&rdf_;owl:DatatypeProperty"
    rdfs:label="owl:DatatypeProperty">
    <rdf:type rdf:resource="&rdf_;owl:Class"/>
    <rdfs:subClassOf rdf:resource="&rdf;Property"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Nothing"
    rdfs:label="owl:Nothing">
    <rdf:type rdf:resource="&rdf_;owl:Class"/>
    <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:ObjectProperty"
    rdfs:label="owl:ObjectProperty">
    <rdf:type rdf:resource="&rdf_;owl:Class"/>
    <rdfs:subClassOf rdf:resource="&rdf;Property"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Ontology"
    rdfs:label="owl:Ontology">
    <rdf:type rdf:resource="&rdf_;owl:Class"/>
    <rdfs:subClassOf rdf:resource="&a;_system_class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:backwardCompatibleWith"
    rdf_:OWL-ANNOTATION-PROPERTY="true"
    rdfs:label="owl:backwardCompatibleWith">
    <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
    <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
    <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:incompatibleWith"
    rdf_:OWL-ANNOTATION-PROPERTY="true"
    rdfs:label="owl:incompatibleWith">
    <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
    <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
    <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:priorVersion"

```

```

        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:priorVersion">
<rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
<rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
<rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:versionInfo"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="owl:versionInfo">
<rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
<rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Property rdf:about="&a_direct_template_slots"
        rdfs:label="DIRECT-TEMPLATE-SLOTS"/>
<rdf:Description rdf:about="&rdf;first"
        rdfs:label="rdf:first">
<rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
<rdfs:domain rdf:resource="&rdfs;List"/>
<rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Description>
<rdf:Property rdf:about="&rdfs;object"
        rdfs:label="rdf:object">
<rdfs:domain rdf:resource="&rdfs;Statement"/>
<rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Property rdf:about="&rdfs;predicate"
        rdfs:label="rdf:predicate">
<rdfs:domain rdf:resource="&rdfs;Statement"/>
<rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Description rdf:about="&rdfs;rest"
        rdfs:label="rdf:rest">
<rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
<rdfs:domain rdf:resource="&rdfs;List"/>
<rdfs:range rdf:resource="&rdfs;List"/>
</rdf:Description>
<rdf:Property rdf:about="&rdfs;subject"
        rdfs:label="rdf:subject">
<rdfs:domain rdf:resource="&rdfs;Statement"/>
<rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Property rdf:about="&rdfs;value"
        rdfs:label="rdf:value">
<rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Description rdf:about="&rdfs;comment"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:comment">
<rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
<rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdfs;isDefinedBy"
        rdf_:OWL-ANNOTATION-PROPERTY="true"
        rdfs:label="rdfs:isDefinedBy">
<rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
<rdfs:range rdf:resource="&rdfs;Resource"/>

```

```

</rdf:Description>
<rdf:Description rdf:about="&rdfs;label"
    rdf_:OWL-ANNOTATION-PROPERTY="true"
    rdfs:label="rdfs:label">
    <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
    <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Property rdf:about="&rdfs;member"
    rdfs:label="rdfs:member">
    <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Description rdf:about="&rdfs;seeAlso"
    rdf_:OWL-ANNOTATION-PROPERTY="true"
    rdfs:label="rdfs:seeAlso">
    <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
    <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Description>
</rdf:RDF>

```

Appendix H: Application C - Party

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
    <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
    <!ENTITY a 'http://protege.stanford.edu/system#'>
    <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
    <!ENTITY rdfs 'http://www.w3.org/TR/1999/PR-rdf-schema-19990303#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
    xmlns:rdf_="&rdf_;"
    xmlns:a="&a;"
    xmlns:rdfs="&rdfs;">
  <rdfs:Class rdf:about="&rdf_::OWL-ALL-DIFFERENT"
    rdfs:label=":OWL-ALL-DIFFERENT">
    <rdfs:subClassOf rdf:resource="&a;_system_class"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-ALL-RESTRICTION"
    rdfs:label=":OWL-ALL-RESTRICTION">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-ANNOTATION-PROPERTY"
    rdfs:label=":OWL-ANNOTATION-PROPERTY">
    <rdfs:domain rdf:resource="&rdf;Property"/>
    <rdfs:range rdf:resource="&rdfs;Literal"/>
  </rdf:Property>
  <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-CLASS"
    rdfs:label=":OWL-ANONYMOUS-CLASS">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-CLASS"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-ANONYMOUS-ROOT"
    rdfs:label=":OWL-ANONYMOUS-ROOT">
    <rdfs:subClassOf rdf:resource="&a;_system_class"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-CARDI-RESTRICTION"
    rdfs:label=":OWL-CARDI-RESTRICTION">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-MAXCARDI-RESTRICTION"/>
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-MINCARDI-RESTRICTION"/>
  </rdfs:Class>
  <rdfs:Class rdf:about="&rdf_::OWL-CLASS"
    rdfs:label=":OWL-CLASS">
    <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-CLASSIFICATION-STATUS"
    rdfs:label=":OWL-CLASSIFICATION-STATUS">
    <rdfs:range rdf:resource="&rdfs;Literal"/>
  </rdf:Property>
  <rdfs:Class rdf:about="&rdf_::OWL-COMPLEMENT-CLASS"
    rdfs:label=":OWL-COMPLEMENT-CLASS">
    <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
  </rdfs:Class>
  <rdf:Property rdf:about="&rdf_::OWL-DEPRECATED"
    rdfs:label=":OWL-DEPRECATED">
    <rdfs:domain rdf:resource="&rdf;Property"/>
    <rdfs:domain rdf:resource="&rdfs;Class"/>
```

```

        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DIFFERENT-FROM"
        rdfs:label=":OWL-DIFFERENT-FROM">
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISJOINT-CLASSES"
        rdfs:label=":OWL-DISJOINT-CLASSES">
        <rdfs:domain rdf:resource="&rdf_::OWL-CLASS"/>
        <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-DISTINCT-MEMBERS"
        rdfs:label=":OWL-DISTINCT-MEMBERS">
        <rdfs:domain rdf:resource="&rdf_::OWL-ALL-DIFFERENT"/>
        <rdfs:range rdf:resource="&a;owl_thing"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-ENUMERATION-CLASS"
        rdfs:label=":OWL-ENUMERATION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-EQUIVALENT-SLOTS"
        rdfs:label=":OWL-EQUIVALENT-SLOTS">
        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:range rdf:resource="&rdf;Property"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-FILLER-DEFINED"
        rdfs:label=":OWL-FILLER-DEFINED">
        <rdfs:domain rdf:resource="&rdf_::OWL-RESTRICTION"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-HAS-RESTRICTION"
        rdfs:label=":OWL-HAS-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUBCLASSES"
        rdfs:label=":OWL-INFERRED-SUBCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_::OWL-INFERRED-SUPERCLASSES"
        rdfs:label=":OWL-INFERRED-SUPERCLASSES">
        <rdfs:range rdf:resource="&rdfs;Class"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-INTERSECTION-CLASS"
        rdfs:label=":OWL-INTERSECTION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_::OWL-INVERSE-FUNCTIONAL"
        rdfs:label=":OWL-INVERSE-FUNCTIONAL">
        <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
        <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_::OWL-LOGICAL-CLASS"
        rdfs:label=":OWL-LOGICAL-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
    </rdfs:Class>

```

```

<rdf:Property rdf:about="&rdf_::OWL-LOGICAL-OPERANDS"
  rdfs:label=":OWL-LOGICAL-OPERANDS">
  <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
  <rdfs:domain rdf:resource="&rdf_::OWL-LOGICAL-CLASS"/>
</rdf:Property>
<rdfs:Class rdf:about="&rdf_::OWL-MAXCARDI-RESTRICTION"
  rdfs:label=":OWL-MAXCARDI-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
</rdfs:Class>
<rdfs:Class rdf:about="&rdf_::OWL-MINCARDI-RESTRICTION"
  rdfs:label=":OWL-MINCARDI-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-RESTRICTION"/>
</rdfs:Class>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-IMPORTS"
  rdfs:label=":OWL-ONTOLOGY-IMPORTS">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-PREFIXES"
  rdfs:label=":OWL-ONTOLOGY-PREFIXES">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-ONTOLOGY-URI"
  rdfs:label=":OWL-ONTOLOGY-URI">
  <rdfs:domain rdf:resource="&rdf_::owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-RESOURCE-URI"
  rdfs:label=":OWL-RESOURCE-URI">
  <rdfs:domain rdf:resource="&rdf_::ExternalResource"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdfs:Class rdf:about="&rdf_::OWL-RESTRICTION"
  rdfs:label=":OWL-RESTRICTION">
  <rdfs:subClassOf rdf:resource="&rdf_::OWL-ANONYMOUS-CLASS"/>
</rdfs:Class>
<rdf:Property rdf:about="&rdf_::OWL-SAME-AS"
  rdfs:label=":OWL-SAME-AS">
  <rdfs:range rdf:resource="&a;owl_thing"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME"
  rdfs:label=":OWL-SLOT-SOME">
  <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>
  <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-CLASSES"
  rdfs:label=":OWL-SLOT-SOME-CLASSES">
  <rdfs:range rdf:resource="&rdf_::OWL-CLASS"/>
  <rdfs:domain rdf:resource="&rdf_::owl:ObjectProperty"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf_::OWL-SLOT-SOME-VALUES"
  rdfs:label=":OWL-SLOT-SOME-VALUES">
  <rdfs:domain rdf:resource="&rdf_::OWL-SOME-RESTRICTION"/>
  <rdfs:domain rdf:resource="&rdf_::owl:DatatypeProperty"/>

```



```

        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_;;OWL-SOME-RESTRICTION"
        rdfs:label=":OWL-SOME-RESTRICTION">
        <rdfs:subClassOf rdf:resource="&rdf_;;OWL-RESTRICTION"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_;;OWL-SYMMETRIC"
        rdfs:label=":OWL-SYMMETRIC">
        <rdfs:domain rdf:resource="&rdf_;;owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Property rdf:about="&rdf_;;OWL-TRANSITIVE"
        rdfs:label=":OWL-TRANSITIVE">
        <rdfs:domain rdf:resource="&rdf_;;owl:ObjectProperty"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdfs:Class rdf:about="&rdf_;;OWL-UNION-CLASS"
        rdfs:label=":OWL-UNION-CLASS">
        <rdfs:subClassOf rdf:resource="&rdf_;;OWL-LOGICAL-CLASS"/>
    </rdfs:Class>
    <rdf:Property rdf:about="&rdf_;;OWL-XML-SCHEMA-DATATYPE"
        rdfs:label=":OWL-XML-SCHEMA-DATATYPE">
        <rdfs:domain rdf:resource="&rdf;Property"/>
        <rdfs:range rdf:resource="&rdfs;Literal"/>
    </rdf:Property>
    <rdf:Description rdf:about="&rdf_;;Address"
        rdfs:label="Address">
        <rdfs:subClassOf rdf:resource="&rdf_;;Contact"/>
        <rdf:type rdf:resource="&rdf_;;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;;Buyer"
        rdfs:label="Buyer">
        <rdfs:subClassOf rdf:resource="&rdf_;;Type"/>
        <rdf:type rdf:resource="&rdf_;;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;;Communication"
        rdfs:label="Communication">
        <rdfs:subClassOf rdf:resource="&rdf_;;Party"/>
        <rdf:type rdf:resource="&rdf_;;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;;Contact"
        rdfs:label="Contact">
        <rdfs:subClassOf rdf:resource="&rdf_;;Party_identification"/>
        <rdf:type rdf:resource="&rdf_;;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;;Email"
        rdfs:label="Email">
        <rdfs:subClassOf rdf:resource="&rdf_;;Contact"/>
        <rdf:type rdf:resource="&rdf_;;owl:Class"/>
    </rdf:Description>
    <rdfs:Class rdf:about="&rdf_;;ExternalResource"
        rdfs:label="ExternalResource">
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdfs:Class>
    <rdf:Description rdf:about="&rdf_;;Language"
        rdfs:label="Language">

```

```

        <rdfs:subClassOf rdf:resource="&rdf_;Party_identification"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Line_item"
        rdfs:label="Line_item">
        <rdfs:subClassOf rdf:resource="&rdf_;Party"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Party"
        rdfs:label="Party">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Party_identification"
        rdfs:label="Party_identification">
        <rdfs:subClassOf rdf:resource="&rdf_;Party"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Party_name"
        rdfs:label="Party_name">
        <rdfs:subClassOf rdf:resource="&rdf_;Party_identification"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Seller"
        rdfs:label="Seller">
        <rdfs:subClassOf rdf:resource="&rdf_;Type"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Tax_schema"
        rdfs:label="Tax_schema">
        <rdfs:subClassOf rdf:resource="&rdf_;Party_identification"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Telephone"
        rdfs:label="Telephone">
        <rdfs:subClassOf rdf:resource="&rdf_;Contact"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;Type"
        rdfs:label="Type">
        <rdfs:subClassOf rdf:resource="&rdf_;Party"/>
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
    </rdf:Description>
    <rdf:Description rdf:about="&rdf_;owl:Class"
        rdfs:label="owl:Class">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&rdfs;Class"/>
    </rdf:Description>
    <rdfs:Class rdf:about="&rdf_;owl:DataRange"
        rdfs:label="owl:DataRange">
        <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
    </rdfs:Class>
    <rdf:Description rdf:about="&rdf_;owl:DatatypeProperty"
        rdfs:label="owl:DatatypeProperty">
        <rdf:type rdf:resource="&rdf_;owl:Class"/>
        <rdfs:subClassOf rdf:resource="&rdf;Property"/>

```

```

</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Nothing"
  rdfs:label="owl:Nothing">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&a;owl_thing"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:ObjectProperty"
  rdfs:label="owl:ObjectProperty">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&rdf;Property"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:Ontology"
  rdfs:label="owl:Ontology">
  <rdf:type rdf:resource="&rdf_;owl:Class"/>
  <rdfs:subClassOf rdf:resource="&a;_system_class"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:backwardCompatibleWith"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="owl:backwardCompatibleWith">
  <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
  <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:incompatibleWith"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="owl:incompatibleWith">
  <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
  <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:priorVersion"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="owl:priorVersion">
  <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
  <rdfs:domain rdf:resource="&rdf_;owl:Ontology"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdf_;owl:versionInfo"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="owl:versionInfo">
  <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Property rdf:about="&a;_direct_template_slots"
  rdfs:label="DIRECT-TEMPLATE-SLOTS"/>
<rdf:Description rdf:about="&rdf;first"
  rdfs:label="rdf:first">
  <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
  <rdfs:domain rdf:resource="&rdfs;List"/>
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Description>
<rdf:Property rdf:about="&rdf;object"
  rdfs:label="rdf:object">
  <rdfs:domain rdf:resource="&rdfs;Statement"/>
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>

```

```

<rdf:Property rdf:about="&rdf;predicate"
  rdfs:label="rdf:predicate">
  <rdfs:domain rdf:resource="&rdf;Statement"/>
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Description rdf:about="&rdf;rest"
  rdfs:label="rdf:rest">
  <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
  <rdfs:domain rdf:resource="&rdf;List"/>
  <rdfs:range rdf:resource="&rdf;List"/>
</rdf:Description>
<rdf:Property rdf:about="&rdf;subject"
  rdfs:label="rdf:subject">
  <rdfs:domain rdf:resource="&rdf;Statement"/>
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Property rdf:about="&rdf;value"
  rdfs:label="rdf:value">
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Description rdf:about="&rdfs;comment"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="rdfs:comment">
  <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Description rdf:about="&rdfs;isDefinedBy"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="rdfs:isDefinedBy">
  <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Description>
<rdf:Description rdf:about="&rdfs;label"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="rdfs:label">
  <rdf:type rdf:resource="&rdf_;owl:DatatypeProperty"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Description>
<rdf:Property rdf:about="&rdfs;member"
  rdfs:label="rdfs:member">
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Property>
<rdf:Description rdf:about="&rdfs;seeAlso"
  rdf_:OWL-ANNOTATION-PROPERTY="true"
  rdfs:label="rdfs:seeAlso">
  <rdf:type rdf:resource="&rdf_;owl:ObjectProperty"/>
  <rdfs:range rdf:resource="&rdfs;Resource"/>
</rdf:Description>
</rdf:RDF>

```

Bibliography

- Artale, A., Franconi, E., Guarino, N., and Pazzi, L. "Part-Whole Relations in Object-Centered Systems: An Overview," in: *Data & Knowledge Engineering*, North-Holland, Elsevier, 1996, pp. 347-383.
- Aussenac-Gilles, N., Biébow, B., and Szulman, S. "Revisiting Ontology Design: A Methodology Based on Corpus Analysis," *EKA W 2000*, 2000, pp. 172 -188.
- Bergamaschi, S., Guerra, F., and Vincini, M. "A Data Integration Framework for E-commerce Product Classification," Dipartimento di Ingegneria dell'Informazione Universit di Modena e Reggio, Modena.
- Boisot, M. *Knowledge assets : securing competitive advantage in the information economy* Oxford University Press, Oxford ; New York, 1998, pp. xx, 284 p.
- Borghoff, U., and Pareschi, R. *Information technology for knowledge management* Springer, Berlin ; New York, 1998a, pp. xi, 232 p.
- Borghoff, U.M., and Pareschi, R. *Information technology for knowledge management* Springer, Berlin ; New York, 1998b, pp. xi, 232 p.
- Botkin, J.W. *Smart business : how knowledge communities can revolutionize your company* Free Press, New York, 1999, p. 291 p.
- Bruijn, J.d. "Using Ontologies: Enabling Knowledge Sharing and Reuse on the Semantic Web," DERI Technical Report DERI-2003-10-29 , DERI – Digital Enterprise Research Institute.
- Bunge, M.A. *A world of systems* Reidel, Dordrecht ; Boston, 1979, pp. xvi, 314.
- Butler, J., Scott, F., and Edwards, J. "Evaluating Organizational Change: The Role of Ontology And Epistemology," *Journal Of Critical Postmodern Organization Science* (2:2) 2003, pp 55 - 67.
- Choo, C.W. *The knowing organization : how organizations use information to construct meaning, create knowledge, and make decisions* Oxford University Press, New York, 1998, pp. xviii, 298.
- Cohen, D., and Conference Board. *Managing knowledge in the new economy* Conference Board, New York, 1998, p. 26 p. ;.

- Creswell, J.W. *Research design : qualitative & quantitative approaches* Sage Publications, Thousand Oaks, Calif., 1994, pp. xix, 228.
- Daconta, M.C., Obrst, L.J., and Smith, K.T. *The semantic Web : a guide to the future of XML, Web services, and knowledge management* Wiley Pub., Indianapolis, Ind., 2003, pp. xxii, 281 p.
- Davies, I., Green, P., Milton, S., and Rosemann, M. "Using Meta Models for the Comparison of Ontologies," Proceedings of Eighth CAiSE/IFIP8.1 International Workshop on Evaluation of Modeling Methods in Systems Analysis and Design, Austria, 2003, p. 12 pages.
- Davies, J., Fensel, D., and Van Harmelen, F. *Towards the semantic web : ontology-driven knowledge management* J. Wiley, New York, 2002.
- Denny, m. "Ontology Tool Survey, Revisited," XML.Org (ed.), 2004.
- Doan, A., Madhavan, J., Domingos, P., and Halevy, A. "Learning to Map between Ontologies on the Semantic Web," WWW2002,, ACM, Honolulu, Hawaii, USA., 2002.
- Eriksson, H., Fergerson, R.W., Shahar, Y., and Musen, M.A. "Automatic Generation of Ontology Editors," Department of Computer and Information Science Linkoping University SE-581 83 Linkoping, Sweden.
- Fensel, D. *Ontologies : a silver bullet for knowledge management and electronic commerce* Springer, Berlin ; New York, 2001a, pp. ix, 138.
- Fensel, D. *Ontologies : a silver bullet for knowledge management and electronic commerce* Springer, Berlin ; New York, 2001b, pp. ix, 138 p.
- Fensel, D. *Intelligent information integration in B2B electronic commerce* Kluwer Academic Publishers, Boston, 2002, pp. xii, 144.
- Fettke, P., and Loos, P. "Ontological Evaluation of Reference Models Using the Bunge-Wand-Weber Model," Ninth Americas Conference on Information Systems, 2003, pp. 2944-2955.
- Firat, A., Madnick, S., and Grosz, B.N. "Knowledge Integration to Overcome Ontological Heterogeneity: Challenges from Financial Information Systems," International Conference on Information Systems (ICIS-2002), Barcelona, Spain, 2002.

- Fisher, M., and Sheth, A. "Semantic Enterprise Content Management," in: *Practical Handbook of Internet Computing*, CRC Press, 2003.
- Gandon, F. "Ontology Engineering: a survey and a return on experience," Institut de Recherche En Informatique et Automatique, France, p. 187.
- Gangemi, A., Pisanelli, D.M., and Steve, G. "Understanding Systematic Conceptual Structures in Polysemous Medical Terms."
- Garigue, R. "Managing Ontological Ambiguity: Extending the Knowledge Management Framework," Carleton University, Kingston, p. 15.
- Garigue, R. "Evaluating Data Interoperability: A Policy Driven Ontological Analysis Model," in: *Faculty of Public Affairs and Management*, Carleton University, Ottawa, 2004, p. 80.
- Geerts, G.L., and McCarthy, W.E. "An ontological analysis of the economic primitives of the extended-REA enterprise information architecture," *International Journal of Accounting Information Systems* (3 (2002):3), 23 May 2001 2002, pp 1-16.
- Gomez-Perez, A. "Ontology Evaluation," in: *Handbook on Ontologies*, S. Staab and R. Studer (eds.), Springer, Berlin, 2004, pp. 251 - 273.
- Gomez-Perez, A., Fernandez--Lopez, M., and Corcho, O. *Ontological Engineering: With Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web* Springer- Verlag, London Berlin Heidelberg, 2004, p. 403.
- Gruber, T.R. *The acquisition of strategic knowledge* Academic Press, Boston ; Toronto, 1989, pp. xxiv, 311.
- Gruber, T.R. "Towards Principles for the Design of Ontologies Used for Knowledge Sharing," in: *Formal Ontology in Conceptual Analysis and Knowledge Representation*, N.G.a.R. Poli (ed.), Kluwer Academic Publishers, 1993.
- Gruninger, M., and Lee, J. "Ontology: different ways of representing the same concept - Special Issue: Ontology applications and design," *Communications of the ACM* (45:2), Feb 2002 2002, pp 39-41.
- Guarino, N. *Formal ontology in information systems : proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy* IOS Press ; Ohmsha, Amsterdam ; Washington, DC, Tokyo, 1998, pp. x, 337.

- Guizzardi, G., Herre, H., and Wagner, G. "Towards Ontological Foundations for UML Conceptual Models," In Proceedings of 1st Int. Conf. on Ontologies, Databases and Applications of Semantics (ODBASE 2002), Springer LNCS 2519, 2002, Springer, 2002, p. 18.
- Haarslev, V., and Möller, R. "Description of the RACER System and its Applications," Proceedings of the International Workshop on Description Logics (DL-2001), Stanford, USA, . 2001.
- Hameed, A., Sleeman, D., and Preece, A. "Detecting Mismatches Among Experts' Ontologies Acquired through Knowledge Elicitation," *Knowledge Based Systems* (15) 2002, pp 265-273.
- Haugeland, J. *Artificial intelligence : the very idea* MIT 1986, Cambridge, Mass., 1985, p. 287.
- Haugeland, J. *Having thought : essays in the metaphysics of mind* Harvard University Press, Cambridge, Mass. ; London, 1998, p. 390.
- Huges, T., and Ashpole, B. "The Semantics of Ontology Alignment," Lockheed Martin Advanced Technology Laboratories, Cherry Hill, NJ, p. 3.
- Ichise, R., Takeda, H., and Honiden, S. "Integrating Multiple Internet Directories by Instance-based Learning," National Institute of Informatics, p. 7.
- Kalfoglou, Y. "Maintaining Ontologies with Organisational Memories,," in: *Knowledge Management and Organizational Memories*, R. Dieng-Kuntz and N. Matta (eds.), Kluwer Academic Publishers, 2002.
- Kalfoglou, Y., and Schorlemmer, M. "Ontology mapping: the state of the art," *The Knowledge Engineering Review* (18:1), January 2003, pp 1-31.
- Kashyap, V., and Sheth, A. "Semantic and schematic similarities between database objects: a context-based approach," *The VLDB Journal*:5), 1996 1996, p 276–304.
- Kashyap, V., and Sheth, A. "Schema Correspondences Between Objects with Semantic Proximity."
- Kavouras "A Unified Ontological Framework for Semantic Integration," International Workshop on Next Generation Geospatial Information,, Cambridge (Boston), Massachusetts, USA,, 2003.

- Kavouras, M., and Kokla, M. "Ontology-Based Fusion of Geographic Databases," International Federation of Surveyors, Commission 3-WG 3.1, Athens, Greece.
- Kementsietsidis, A., Arenas, M., and Miller, R.J. "Mapping data in peer-to-peer systems: semantics and algorithmic issues," Proceedings of the 2003 ACM SIGMOD international conference on on Management of data, ACM Press New York, NY, USA, San Diego, California, 2003, pp. 325 - 336.
- Kim, H. "Developing Ontologies to Enable Knowledge Management: Integrating Business Process and Data Driven Approaches," Papers from 2000 AAAI Spring Symposium, AAAI, 2000.
- Kim, H., and Fox, M. "Towards A Data Model for Quality managment Web Services: An Ontology of Measurement for Enterprise Modeling," A.B.P.e. al. (ed.), Springer-Verlag, Berlin Heirdelberg, 2002.
- Knublauch, H., Musen, M.A., and Noy, N.F. "Tutorial:Creating Semantic Web (OWL) Ontologies with Protégé," 2nd International Semantic Web Conference - ISWC2003, Stanford Medical Informatics, Sanibel Island, Florida,USA, 2003, p. 52.
- Lakatos, I., and Musgrave, A. *Criticism and the growth of knowledge* University Press, Cambridge [Eng.], 1970, pp. viii, 282 p.
- Leonard, H.S., and Goodman, N. "The calculus of individuals and its uses," *Journal of Symbolic Logic* (5) 1940, pp 45--55.
- Lesser, E.L. *Knowledge and social capital : foundations and applications* Butterworth-Heinemann, Boston, 2000, p. 323 p.
- Lloyd, P., and Boyle, P. *Web-weaving : intranets, extranets and strategic alliances* Butterworth-Heinemann, Oxford ; Boston, 1998, pp. xxxvii, 374 p.
- Maedche, A., and Staab, S. "Measuring Similarity between Ontologies," Proc. Of the European Conference on Knowledge Acquisition and Management - EKAW-2002, Springer, Madrid, Spain, 2002.
- March, J.G. *The pursuit of organizational intelligence* Blackwell Business, Malden, Mass., 1999, pp. ix, 397.
- Margulius, D.L. "Finding the right tools," *InfoWorld.*), August 30 2002.

- Mark S. Fox, M.B., Michael Gruninger, Jinxin Lin "An Organization Ontology for Enterprise Modelling," in: *Simulating Organizations: Computational Models of Institutions and Groups*, K.C. M. Prietula, L. Gasser (ed.), AAAI/MIT Press, Melo Park, 1998, pp. 131-152.
- McComb, D. *Semantics in business systems : the savvy manager's guide* Morgan Kaufmann Publishers, Amsterdam ; Boston, 2004, p. 399 p.
- Mentzas, G. *Knowledge asset management : beyond the process-centered and product-centered approaches* Springer, London ; New York, 2002, pp. xv, 195.
- Moen, W.E. "Mapping the Interoperability Landscape for Networked Information Retrieval," Proceedings of the first ACM/IEEE-CS joint conference on Digital libraries, ACM Press New York, NY, USA, Roanoke, Virginia, United States, 2001, pp. 50 - 51.
- Moody, D.L., Sindre, G., Brasethvik, T., and Solvberg, A. "Evaluating the Quality of Information Models: Empirical Testing Conceptual Model Quality Framework," 25th International Conference on Software Engineering, IEEE Computer Society, Portland, Oregon, 2003.
- Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., and Swartout, W. "Enabling technology for knowledge sharing," *AI Magazine* (12:3), Fall 1991, pp 36- 56,.
- Nirenburg, S., and Raskin, V. "Ontological semantics, formal ontology, and ambiguity," Formal Ontology in Information Systems: Proceedings of the international conference on Formal Ontology in Information Systems, ACM Press New York, NY, USA, Ogunquit, Maine, USA, 2001, pp. 151-161.
- Noy, N.F., and McGuinness, D.L. "Ontology Development 101: A Guide To Creating Your First Ontology," Stanford, p. 25.
- Noy, N.F., and Musen, M.A. "PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment," Proc. 17th Natl. Conf. on Artificial Intelligence (AAAI'2000),, Austin, TX,, 2000.
- Noy, N.F., and Musen, M.A. "Evaluating Ontology-Mapping Tools: Requirements and Experience," SMI Report Number: SMI-2002-0936, Stanford Medical Informatics, p. 14.

- Noy, N.F., and Musen, M.A. "The PROMPT Suite: Interactive Tools for Ontology Merging and Mapping," *Stanfords Medical Informatics*, p. 69.
- Obrst, L. "Ontologies for semantically interoperable systems," *Proceedings of the twelfth international conference on Information and knowledge management*, New Orleans, LA, USA, 2003, pp. 366--369.
- Obrst, L., Wray, R.E., and Liu, H. "Ontological engineering for B2B E-commerce," *Proceedings of the international conference on Formal Ontology in Information Systems - , ACM Special Interest Group on Artificial Intelligence*, Ogunquit, Maine, USA, 2001, pp. 117 - 126.
- Osterwalderÿ, A. "An e-Business Model Ontology for the Creation of New Management Software Tools and IS Requirement Engineering," *Ecole des HEC, Université de Lausanne,, Lausanne*.
- O'Sullivan, D., and Lewis, D. "Semantically driven service interoperability for pervasive computing," *Proceedings of the 3rd ACM international workshop on Data engineering for wireless and mobile access*, {San Diego, CA, USA, 2003, pp. 17--24.
- Partridge, C. "What is a customer? The beginnings of a reference ontology for customer," *The Business Object Reference Ontology Program*, p. 12.
- Pearlson, K. *Managing and using information systems : a strategic approach*, (1st ed.) John Wiley & Sons, New York, 2001, pp. xviii, 278 p.
- Peat, B. "Becoming Net Centric - Leveraging an Information Network with Communities of Interests, Architectures, and Ontologies," *Eprocess Solutions*.
- Pollock, J.T., and Hodgson, R. *Adaptive information : improving business through semantic interoperability, grid computing, and enterprise integration* John Wiley & Sons, Hoboken, N.J., 2004.
- Popper, K.R. *The logic of scientific discovery*, ([3d ed.) Hutchinson, London,, 1968, p. 480 p.
- Popper, K.R. *Conjectures and refutations : the growth of scientific knowledge*, (5th ed.) Routledge, London ; New York, 1989a, pp. xiii, 431 p.
- Popper, K.R. *Conjectures and refutations : the growth of scientific knowledge*, (5th (rev.) ed.) Routledge, London ; New York, 1989b, pp. xiii, 431.

- Psillos, S. *Scientific realism : how science tracks truth* Routledge, London ; New York, 1999, pp. xxv, 341 p.
- Ram, S., and Park, J. "Semantic Conflict Resolution Ontology (SCROL): An Ontology for Detecting and Resolving Data- and Schema-Level Semantic Conflicts."
- Renner, S., Rosenthal, A., and Scarano, J. "Data Interoperability: Standardization or Mediation,," IEEE Metadata Workshop, Silver Spring, MD, 1996.
- Renner, S.A. "A "Community of Interest" Approach to Data Interoperability," Federal Database Colloquium, San Diego, 2001.
- Resnik, P. "Semantic Similarity in a Taxonomy: An Information-Based Measure and Its Application to Problems of Ambiguity In Natural Language," *Journal of Artificial Intelligence Research*) 1998.
- Resnikoff, H.L. *The illusion of reality* Springer-Verlag, New York, 1989, pp. x, 339 p.
- Riichiro Mizoguchi, M.I., Katherine Sinita "Roles of Shared Ontology in AI-ED Research -- Intelligence, Conceptualization, Standardization, and Reusability," Osaka University,, Osaka, pp. pp.537-544.
- Rosemann, M., and Green, P. "Integrating Multi-Perspective Views Into Ontological Analysis," ICIS 2000, Association for Information Systems, Brisbane Australia, 2000.
- Rosenthal, A., Renner, S., Seligman, L., and Manola, F. "Data Integration Needs an Industrial Revolution," *Submitted to VLDB 2001*) 2001.
- Schlenoff, C., Ivester, R., Libes, D., Denno, P., and Szykman:, S. "An Analysis of Existing Ontological Systems for Applications in Manufacturing and Healthcare," NISTIR 6301, National Institute of Standards and Technology, Gaithersburg, MD,.
- Schmelzer, R. "Solving Information Integration Challenges in a Service-Oriented Architecture," Zapthink, Waltham, p. 16.
- Shanks, G., Tansley, E., and Weber, R. "Using Ontologies to Validate Conceptual Models," *COMMUNICATIONS OF THE ACM* (Vol. 46:No 10), October 2003 2003, pp 85 - 90.

- Sheth, A., and Larson, J. "Federated database systems for managing distributed, heterogenous, and autonomous databases," *ACM Computing Surveys* (22:3) 1990, pp 183-230.
- Simon, H.A. *The new science of management decision*, (Rev. -- ed.) Prentice-Hall, Englewood Cliffs, N.J., 1977, pp. xi, 175.
- Small, S.L., Cottrell, G.W., Tanenhaus, M.K., and Adriaens, G. *Lexical ambiguity resolution : perspectives from psycholinguistics, neuropsychology, and artificial intelligence* Morgan Kaufmann Publishers, San Mateo, Calif., 1988, pp. ix, 518.
- Smith, B.C. *On the origin of objects* MIT Press, Cambridge, Mass., 1996, pp. xii, 420 p.
- Sowa, J.F. *Conceptual structures : information processing in mind and machine* Addison-Wesley, Reading, Mass., 1984, pp. xiv, 481.
- Sowa, J.F. *Principles of semantic networks : explorations in the representation of knowledge*, San Mateo, Calif. Morgan Kaufmann Pub., 1991, 1991, pp. xi, 582.
- Sowa, J.F. *Knowledge representation : logical, philosophical, and computational foundations* Brooks/Cole, Pacific Grove, 2000, pp. xiv, 594.
- Staab, S., and Studer, R. *Handbook on ontologies* Springer, Berlin ; New York, 2004, pp. xv, 660 p.
- Stewart, T.A. *The wealth of knowledge : intellectual capital and the twenty-first century organization*, (1st ed.) Currency, New York, 2001, pp. xx, 379 p.
- Stuckenschmidt, H., and Harmelen, F.v. *Information sharing on the semantic web*, (1st ed.) Springer, New York, NY, 2004.
- Swartout, B., Patil, R., Knight, K., and Russ, T. "Toward Distributed Use of Large-Scale Ontologies," In *Proceedings of the 10th Knowledge Acquisition, Modeling and Management Workshop (KAW'96)*, Banff, Canada, 1996.
- Tarski, A. *Introduction to logic and to the methodology of deductive sciences*, (Enl. and rev. ed.) Oxford University Press, New York, 1941, p. 239p.
- Tarski, A., and Tarski, J. *Introduction to logic and to the methodology of the deductive sciences*, (4th ed.) Oxford University Press, New York, 1994, pp. xxii, 229 p.
- Thagard, P. *How scientists explain disease* Princeton University Press, Princeton, N.J., 1999, pp. xviii, 263 p.

- Thagard, P. *Coherence in thought and action* MIT Press, Cambridge, Mass., 2000, pp. xiv, 312 p.
- Tiwana, A. *The essential guide to knowledge management : e-business and CRM applications* Prentice Hall PTR, Upper Saddle River, NJ, 2001, pp. xxiv, 315 p.
- Tozer, G.V. *Metadata management for information control and business success* Artech House, Boston ; London, 1999, pp. xxv, 318.
- Uschold, M., and Gruninger, M. "Ontologies and Semantics for Seamless Connectivity," *SIGMOD Record*, Vol. 33, No. 4, December 2004 (33:4), Dec 2004 2004, pp 58 - 63.
- Van Fraassen, B.C. *The empirical stance* Yale University Press, New Haven, CT, 2002, pp. xix, 282 p.
- Van Fraassen, B.C., Churchland, P.M., and Hooker, C.A. *Images of science : essays on realism and empiricism, with a reply from Bas C. van Fraassen* University of Chicago Press, Chicago, 1985, pp. viii, 309 p.
- Vergara, J.E.L.d., Villagr, V.A., and Berrocal, J. "Semantic Management: advantages of using an ontology based management information meta-model," E.T.S.I. de Telecomunicacin, Ciudad Universitaria, 28040 Madrid, Spain.
- Vergara, J.E.L.d., Villagr, V.A., and Berrocal., J. "Semantic Management: advantages of using an ontology based management information meta-model," Departamento de Ingeniera de Sistemas Telemticos, Universidad Politcnica de Madrid (DIT-UPM). Madrid, Spain.
- Visser, P.R.S., Jones, D.M., Bench-Capon, T.J.M., and Shave, M.J.R. "An Analysis of Ontology Mismatches; Heterogeneity Versus Interoperability," AAAI 1997 Spring Symposium on Ontological Engineering, 1997.
- Vogt, C. "Intractable ERP: A Comprehensive Analysis of Failed Enterprise-Resource-Planning Projects," *ACM SIGSOFT* (27:2), March 2002, pp 62-69.
- Wache, H., Vogele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., and Hubner, S. "Ontology-Based Integration of Information - A Survey of Existing Approaches," Intelligent Systems Group, Center for Computing Technologies.

- Wand, Y., Storey, V., and Weber, R. "An Ontological Analysis of the Relationship Construct in Conceptual Modeling," *ACM Transactions on Database Systems* (Vol. 24,:No. 4), December 1999, 1999, p 494–528.
- Wand, Y., and Wang, R.Y. "Anchoring Data Quality Dimensions in Ontological Foundations," *COMMUNICATIONS OF THE ACM* (Vol. 39,:No. 11) 1996, pp 85 - 95.
- Weick, K.E. *Sensemaking in organizations* Sage Publications, Thousand Oaks, 1995, pp. xii, 231.
- Weick, K.E. *Making sense of the organization* Blackwell Publishers, Oxford, UK ; Malden, MA, 2001, pp. xii, 483.
- Weinstein, P.C., and Birmingham, W. "Comparing Concepts in Differentiated Ontologies," Twelfth Workshop on Knowledge Acquisition, Modeling and Management, Banff, Alberta, Canada, 1999.
- Wiig, K.M. *Knowledge management foundations : thinking about thinking : how people and organizations create, represent, and use knowledge* Schema Press, Arlington, Tex., 1993, pp. xvi, 471 p.
- Zhao, Y. "Develop the Ontology for Internet Commerce by Reusing Existing Standards," Linköping University, Linköping, p. 7.
- Zhong, N. "Representation And Construction Of Ontologies for Web Intelligence," *Internaltional Journal Of Foundations of Computer Science* (13:4), Feb 2002 2002, pp 555-570.