Tools for enterprises collaboration in virtual enterprises

This item was submitted to Loughborough University’s Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: [https://dspace.lboro.ac.uk/2134/12512](https://dspace.lboro.ac.uk/2134/12512)

Publisher: © Sri Krishna Kumar

Please cite the published version.
TOOLS FOR ENTERPRISES COLLABORATION IN VIRTUAL ENTERPRISES

by

Sri Krishna Kumar

A Doctoral Thesis

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Loughborough University

2013

©Sri Krishna Kumar 2013
ABSTRACT

Virtual Enterprise (VE) is an organizational collaboration concept which provides a competitive edge in the globalized business environment. The life cycle of a VE consists of four stages i.e. opportunity identification (Pre-Creation), partner selection (Creation), operation and dissolution. The success of VEs depends upon the efficient execution of their VE-lifecycles along with knowledge enhancement for the partner enterprises to facilitate the future formation of efficient VEs. This research aims to study the different issues which occur in the VE lifecycle and provides a platform for the formation of high performance enterprises and VEs.

In the pre-creation stage, enterprises look for suitable partners to create their VE and to exploit a market opportunity. This phase requires explicit and implicit information extraction from enterprise data bases (ECOS-ontology) for the identification of suitable partners. A description logic (DL) based query system is developed to extract explicit and implicit information and to identify potential partners for the creation of the VE.

In the creation phase, the identified partners are analysed using different risks paradigms and a cooperative game theoretic approach is used to develop a revenue sharing mechanism based on enterprises inputs and risk minimization for optimal partner selection.

In the operation phases, interoperability remains a key issue for seamless transfer of knowledge information and data. DL-based ontology mapping is applied in this research to provide interoperability in the VE between enterprises with different domains of expertise.

In the dissolution stage, knowledge acquired in the VE lifecycle needs to be disseminated among the enterprises to enhance their competitiveness. A DL-based ontology merging approach is provided to accommodate new knowledge with existing data bases with logical consistency.
Finally, the proposed methodologies are validated using the case study. The results obtained in the case study illustrate the applicability and effectiveness of proposed methodologies in each stage of the VE life cycle.

**Keywords:** Virtual Enterprise (VE), Ontology, Description Logic (DL), Cooperative game theory, nonlinear programming.
When I started my PhD at Loughborough University in December 2009, I was looking forward to the opportunity to learn and to develop myself and I am greatly indebted to all who have helped and supported me during these years.

Firstly, I would like to thank my supervisor Prof. J A Harding for her great support, both financially and academically, motivation and extra ordinary guidance during the various stages of my PhD journey. She allowed me to think and act independently but guided and inspired me when I failed. It is my honour to pursue research under her learned supervision and my whole career will be the reflection of what I learnt from her.

My PhD journey could not have started without the guidance of Prof. M K Tiwari. He motivated me when I was most demotivated; he encouraged me when I was most discouraged in my life. I am also thankful to Mrs. Usha Tiwari for her lovely support and delicious foods and lovely Manu and Ashu for refreshing encouragement and chats.

I am also thankful to my brother Srinivas and my colleagues Rahul, Nitesh and Claire for the inspiring and constructive discussions.

Finally I would like to express my deepest regards to my parents, brothers and sister for their love and support and to the almighty God for giving me such wonderful people in my life.
# TABLE OF CONTENTS

ABSTRACT .......................................................................................................................... 1

ACKNOWLEDGEMENTS ................................................................................................. III

TABLE OF CONTENTS .................................................................................................. IV

LIST OF TABLES .......................................................................................................... IX

LIST OF FIGURES ......................................................................................................... X

GLOSSARY OF TERMS ................................................................................................. XII

LIST OF MATHEMATICAL SYMBOLS ........................................................................ XIII

CHAPTER 1 : INTRODUCTION AND RESEARCH OVERVIEW ....................................... 1

1.1 Background ............................................................................................................. 1

1.2 Research Context .................................................................................................. 4

1.3 Research Issues ..................................................................................................... 5

1.4 Thesis Organization: ............................................................................................ 7

CHAPTER 2 : LITERATURE REVIEW ............................................................................ 8

2.1 Introduction ............................................................................................................ 8

2.2 VE Definition ......................................................................................................... 8

  2.2.1 VE-Level benefits: ................................................................................... 9

  2.2.2 Enterprise-Level benefits: .................................................................... 10

2.3 VE-Lifecycle .......................................................................................................... 12

  2.3.1 Pre-Creation ............................................................................................ 13

  2.3.2 Creation ..................................................................................................... 17

  2.3.3 Operation ................................................................................................... 23

  2.3.4 Termination ............................................................................................... 30
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.4 Research Aims and Objectives</td>
<td>34</td>
</tr>
<tr>
<td>2.5 Research Methodology</td>
<td>35</td>
</tr>
<tr>
<td>2.6 Summary</td>
<td>37</td>
</tr>
<tr>
<td>Chapter 3: Description Logic and Ontology</td>
<td>40</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>40</td>
</tr>
<tr>
<td>3.2 Description Logic (DL)</td>
<td>40</td>
</tr>
<tr>
<td>3.2.1 TBox of DL</td>
<td>41</td>
</tr>
<tr>
<td>3.2.2 ABox of DL</td>
<td>42</td>
</tr>
<tr>
<td>3.2.3 Axioms in $\mathcal{KB}$</td>
<td>42</td>
</tr>
<tr>
<td>3.2.4 Interpretations of $\mathcal{KB}$</td>
<td>43</td>
</tr>
<tr>
<td>3.2.5 Reasoning in DL</td>
<td>44</td>
</tr>
<tr>
<td>3.3 Ontology and DL</td>
<td>45</td>
</tr>
<tr>
<td>3.4 Web Ontology Language (OWL)</td>
<td>46</td>
</tr>
<tr>
<td>Chapter 4: DL-based query in ECOS ontology for partner identification in VE</td>
<td>48</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>48</td>
</tr>
<tr>
<td>4.2 Ontology queries</td>
<td>49</td>
</tr>
<tr>
<td>4.2.1 TBox queries</td>
<td>49</td>
</tr>
<tr>
<td>4.2.2 ABox queries</td>
<td>50</td>
</tr>
<tr>
<td>4.3 ECOS query</td>
<td>51</td>
</tr>
<tr>
<td>4.3.1 Limitation of Queries</td>
<td>51</td>
</tr>
<tr>
<td>4.4 Query Rearrangement</td>
<td>54</td>
</tr>
<tr>
<td>4.5 Query Optimization</td>
<td>57</td>
</tr>
<tr>
<td>4.5.1 Concepts vs. Concepts</td>
<td>57</td>
</tr>
<tr>
<td>4.5.2 Role vs. Role</td>
<td>58</td>
</tr>
<tr>
<td>4.5.3 Concepts vs. Roles</td>
<td>59</td>
</tr>
</tbody>
</table>
### 4.6 Query Processing and Implementation ........................................................... 63

### 4.7 ECOS-Query Implementation ......................................................................... 66

### CHAPTER 5 : RISK ANALYSIS AND GAME THEORETIC REVENUE SHARING MECHANISM FOR PARTNER SELECTION IN VE ................................. 68

#### 5.1 Introduction ..................................................................................................... 68

#### 5.2 Risk Analysis in Partner Selection ................................................................. 68

#### 5.3 Risk evaluation in Partner selection ................................................................ 72
   - 5.3.1 Individual Performance risk ................................................................ 72
   - 5.3.2 Collaborative performance risk ........................................................... 73
   - 5.3.3 Imposed collaborative performance risk ............................................. 74
   - 5.3.4 Network risk ........................................................................................ 74

#### 5.4 Game Theoretic aspects in Partner Selection .................................................. 76
   - 5.4.1 Cooperative Game .............................................................................. 77
   - 5.4.2 Cooperative game theory and partner selection .................................. 77
   - 5.4.2 Mathematical representation ............................................................... 80

#### 5.5 Solution Methodology .................................................................................... 84

#### 5.6 Sensitivity Analysis ........................................................................................ 89

### CHAPTER 6 : DL BASED ONTOLOGY MAPPING FOR INTEROPERABILITY IN VE OPERATION PHASE ................................................................. 90

#### 6.1 Introduction ..................................................................................................... 90

#### 6.2 Semantic heterogeneity in ontology mapping ................................................. 92

#### 6.3 Bridging axioms and Ontology Mapping ....................................................... 95

#### 6.4 Logical derivation of bridging axioms in Ontology Mapping ......................... 96
   - 6.4.1 Concept level ...................................................................................... 97
   - 6.4.2 Role level .......................................................................................... 100
   - 6.4.3 Concept and role level ...................................................................... 102
   - 6.4.4 Combinations of concepts and roles ................................................. 104
CHAPTER 7: DL BASED ONTOLOGY MERGING FOR KNOWLEDGE ENHANCEMENT IN VE’S TERMINATION PHASE

7.1 Introduction
7.2 Ontology Similarity
   7.2.1 Similarity function
7.3 Ontology Similarity index (Matrix)
   7.3.1 Equivalent Similarity Matrix (ERM)
   7.3.2 Super Similarity Matrix (SRM)
   7.3.3 Sub Similarity Matrix (sRM)
7.4 Ontology Merging through Reconfiguration and restructuring
   7.4.1 Similarity and type of similarity of concepts
   7.4.2 Ontology merging procedure
7.5 Implementation Method

CHAPTER 8: HIGH PERFORMANCE VES: IMPLEMENTATION

8.1 Introduction
8.2 ECOS-Query implementation
8.3 Partner Selection in VE
8.4 Interoperability in VE
   8.4.1 Concepts vs. Concepts
   8.4.2 Role vs. role
   8.4.3 Concepts vs. Property
LIST OF TABLES

Table 2:1 Different Approaches in Ontology mapping .................................................... 28
Table 2:2 Relationships between research objectives and chapters................................. 39
Table 3:1 DL Syntax and Semantics.................................................................................. 44
Table 3:2 OWL Syntax and Semantics............................................................................. 47
Table 8:1 VE requirements ............................................................................................. 140
Table 8:2 ECOS-Query results: capable companies for the formation of the VE ........ 147
Table 8:3 Experiment results ......................................................................................... 153
Table 8:4 Partner selection problem results..................................................................... 155
Table 8:5 Results of problem set 1, 9 and 22................................................................. 156
Table 8:6 Selected partners for the VE ......................................................................... 157
Table 8:7 Merging process outcome ............................................................................. 172
LIST OF FIGURES

Figure 1:1 Life Cycle of VEs................................................................. 3
Figure 1:2 Research issues in VE ......................................................... 6
Figure 2:1 Research issues in VE-lifecycle ............................................. 35
Figure 2:2 Research Methodology ....................................................... 36
Figure 4:1 ECOS-Query system ............................................................ 54
Figure 4:2 ECOS-Query Steps ............................................................. 67
Figure 5:1 Partner selection process .................................................... 69
Figure 5:2 VE Risks ........................................................................... 70
Figure 5:3 Pseudocode ..................................................................... 88
Figure 6:1 Ontology mapping for interoperability in operation phase ....... 91
Figure 6:2 Heterogeneity in Ontology mapping ..................................... 94
Figure 6:3 Ontology mapping ............................................................. 97
Figure 6:4 Overall mapping procedures .............................................. 108
Figure 7:1 Ontology merging process for accommodating new knowledge .. 110
Figure 7:2 Case 1 ............................................................................. 118
Figure 7:3 Case 2 ............................................................................. 119
Figure 7:4 Case 3 ............................................................................. 120
Figure 7:5 Case 4 ............................................................................. 121
Figure 7:6 Case 5 ............................................................................. 121
Figure 7:7 Case 6 ............................................................................. 122
Figure 7:8 Case 7 ............................................................................. 123
Figure 7:9 Case 8 ............................................................................. 124
Figure 7:10 Case 9 .......................................................................... 124
Figure 7:11 Case $1 ......................................................................... 125
Figure 7:12 Case $2 ......................................................................... 126
Figure 7:13 Case $3 ......................................................................... 127
Figure 7:14 Case $4 ......................................................................... 127
Figure 7:15 Case $5 ......................................................................... 128
Figure 7:16 Case $6 ......................................................................... 129
Figure 7:17 Case $7 ......................................................................... 129
Figure 7:18 Case $8 ......................................................................... 130
Figure 7:19 Case $9 ......................................................................... 131
Figure 7:20 Merged ontology ............................................................. 133
Figure 7:21 Procedure for ontology merging ....................................... 134
Figure 8:1 ECOS Ontology in Protégé ............................................... 139
Figure 8:2 Query written in JAVA ...................................................... 141
Figure 8:3 Rearranged query according to the variables ....................... 142
Figure 8:4 Query Optimization ......................................................... 143
Figure 8:5 Rearranged Queries after Optimization .............................. 144
Figure 8:6 Final Results of the Query ............................................... 146
Figure 8:7 Two different ontologies ................................................... 159
Figure 8:8 Initial similarity input................................................................. 160
Figure 8:9 Equivalent class assertion........................................................... 161
Figure 8:10 Different levels of sub-concepts.................................................. 162
Figure 8:11 Relationship between EntA:Dimension and EntB:Dimension........ 162
Figure 8:12 Role definition in two ontologies............................................... 163
Figure 8:13 Property assertion...................................................................... 164
Figure 8:14 Class vs. Property comparison.................................................... 164
Figure 8:15 Class equivalent property assertion.............................................. 165
Figure 8:16 Class equivalent to property assertion......................................... 165
Figure 8:17 Mapped global ontology.............................................................. 166
Figure 8:18 Input Ontologies....................................................................... 168
Figure 8:19 Tbox input.................................................................................. 169
Figure 8:20 Merged / Reconfigured Ontology................................................. 173
## GLOSSARY OF TERMS

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABox</td>
<td>Assertion box</td>
</tr>
<tr>
<td>COR</td>
<td>Coalitionly rational</td>
</tr>
<tr>
<td>CR</td>
<td>Collectively rational</td>
</tr>
<tr>
<td>DL</td>
<td>Description logic</td>
</tr>
<tr>
<td>ECOS</td>
<td>Enterprise Competency Organization Schema</td>
</tr>
<tr>
<td>FOL</td>
<td>First order logic</td>
</tr>
<tr>
<td>ICT</td>
<td>Information and communication Technology</td>
</tr>
<tr>
<td>IR</td>
<td>Individually rational</td>
</tr>
<tr>
<td>KB</td>
<td>Knowledge base</td>
</tr>
<tr>
<td>KM</td>
<td>Knowledge management</td>
</tr>
<tr>
<td>MCP</td>
<td>Marginal contribution principle</td>
</tr>
<tr>
<td>OWL</td>
<td>Web ontology language</td>
</tr>
<tr>
<td>RBox</td>
<td>Role box</td>
</tr>
<tr>
<td>RDF</td>
<td>Resource description framework</td>
</tr>
<tr>
<td>SME</td>
<td>Small and medium size enterprises</td>
</tr>
<tr>
<td>TBox</td>
<td>Terminological box</td>
</tr>
<tr>
<td>XML</td>
<td>Extensive markup language</td>
</tr>
<tr>
<td>Symbol</td>
<td>Meaning</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>→</td>
<td>If then</td>
</tr>
<tr>
<td>⊢</td>
<td>Inference</td>
</tr>
<tr>
<td>✓</td>
<td>Cannot be inferred</td>
</tr>
<tr>
<td>⊑</td>
<td>Less general</td>
</tr>
<tr>
<td>⊒</td>
<td>More general</td>
</tr>
<tr>
<td>⊥</td>
<td>Disjoint</td>
</tr>
<tr>
<td>¬</td>
<td>Negation</td>
</tr>
<tr>
<td>R</td>
<td>Inverse role</td>
</tr>
<tr>
<td>T</td>
<td>Interpretation</td>
</tr>
<tr>
<td>C(a)</td>
<td>Concept assertion</td>
</tr>
<tr>
<td>?C(x)</td>
<td>Concept query</td>
</tr>
<tr>
<td>∧</td>
<td>And</td>
</tr>
<tr>
<td>L_r</td>
<td>Lower performance risk</td>
</tr>
<tr>
<td>α</td>
<td>Optimistic value</td>
</tr>
<tr>
<td>c</td>
<td>Cost of sub project</td>
</tr>
<tr>
<td>R</td>
<td>Collaborative performance risk</td>
</tr>
<tr>
<td>nr</td>
<td>Network risk</td>
</tr>
<tr>
<td>V</td>
<td>Project value</td>
</tr>
<tr>
<td>x</td>
<td>Partner selection variable</td>
</tr>
<tr>
<td>mc</td>
<td>Marginal contribution</td>
</tr>
<tr>
<td>N</td>
<td>Number of subprojects</td>
</tr>
</tbody>
</table>
CHAPTER 1: INTRODUCTION AND RESEARCH OVERVIEW

1.1 BACKGROUND

The pressures of global competitive markets are increasingly forcing enterprises towards automation and collaboration. Collaborations affect enterprises’ working environments as nowadays both products and services are commonly achieved by the collective inputs of different enterprises. Enterprise collaboration not only benefits large enterprises but can also provide the catalyst for the growth of small and medium size enterprises (SMEs) with single competency characteristics.

The evolution of enterprise collaboration networks started in the early decades of

the 19th century when the market started to shift from craft production to mass production (Womack et al., 1991). The vertically integrated supply chain was the consequence of

mass production and its aftermath (Jagdev and Thoben, 2001). However, due to large networks of non-comparable activities and without proper information and knowledge flow, enterprises failed to achieve their desired goals in vertically integrated supply chains (Prahalad and Hamel, 1990; Parker and Anderson, 2002). Quinn and Hilmer (1994) explained the importance of core competency and outsourcing in the current era of

business environments. Nowadays, enterprises are more focused on their core competency and outsource other activities which have little to no strategic value for their market position (Prahalad and Hamel, 1990) and outsourcing can be as high as 70% of

the final product value (Van Weele, 1994). In other words the combination of outsourcing and collaboration has helped enterprises to focus and develop their own core competency and skill.

An enterprise collaboration can aptly be defined as a network of enterprises working together in a form of synergistic whole with blurred boundaries and the aim of
achieving a common goal. The possible scope of enterprise collaborations is extensive (Jagdev and Thoben, 2001) and diverges from long term, high level cooperation in the form of an extended enterprise (EE) (Browne et al., 1995; Davidow and Malone, 1992) to short term, dynamic collaboration in the form of a virtual enterprise (VE) (Eversheim et al., 1998). A VE contains almost all the characteristics of an EE and further includes market-driven cooperation, complementarity, dynamic participation, non-hierarchy and process and resource sharing (Martinez et al., 2001; Camarinha-Matos and Afarmanesh, 2003).

In contrast to an EE, a VE is a temporary collaboration of enterprises or business organizations to exploit a market opportunity by focusing on combining and utilising each partner’s core competency to accomplish the common goal which the partner enterprises could not have satisfied individually. In other words, the VE paradigm suits the 21st century global business environment and exploits the prevailing market conditions by using the following characteristics:

1. Enterprises can focus on their core competencies yet still capture larger market segments (Sieber and Griese, 1998)

2. Enterprises, especially SMEs, are able to provide a wider range of product and services with limited individual resources (Neubert et al., 2001)

3. Great agility can be achieved in the network as a new VE can be formed in the context of a specific market scenario (Sieber and Griese, 1998).

Each VE runs in a cycle, called the VE life-cycle, consisting of four phases (Synergy, 2010) – 1. Identification or pre-creation 2. Creation 3. Operation and 4. Termination (Figure 1.1). The pre-creation phase starts with the identification of a market opportunity which requires appropriate core-competencies. The creation phase facilitates the formation of the VE and defines the individual roles required to achieve the common
goal. Scheduled tasks and goals are achieved in the operation phase and then the VE disbands in the termination phase. Individual enterprises can join new VEs and starts searching for their next opportunities and the VE lifecycle starts again.

![Figure 1:1 Life Cycle of VEs](image)

A VE must respond fully and quickly to the prevailing market conditions and therefore its success in achieving goals is a critical factor in order to gain all the benefits of market success. The success of a VE depends upon the efficient execution of each phase of its lifecycle and therefore close monitoring of the VE and its partners is essential (Yang and Liu, 2008). The effectiveness of a VE is achieved through the full cooperation of its individual partner enterprises working within a VE network. According to Camarinha-Matos and Afarmanesh (1999) the driving forces for the formation of VEs among SMEs include capturing bigger market share and utility by sharing risk, investment and resources and effectively allocating revenue and profit. Specifically the
execution of a VE should be based on the proper utilization of knowledge and skills of the partner enterprises. Therefore the success of a VE depends on the implementation of effective and efficacious knowledge management (KM) in every stage of the VE lifecycle (Chryssouris et al., 2008).

1.2 RESEARCH CONTEXT

The research context of this thesis is the provision of support for SMEs collaborating within VEs. As discussed in the last section, globalization has forced SME’s to collaborate to remain competitive in the global market. In VEs collaboration SMEs break their individual capacity and capability barriers and achieve greater benefits by increasing market share (Parker, 2000; McCarthy and Golicic, 2002; McLaren et al., 2002), increasing flexibility and capability (Parker, 2000; Holton, 2001; Tidd et al., 2002), achieving best practice (Manders and Brenner, 1995; Jagdev and Browne, 1998), reducing lead times and increasing market responsiveness (Parker, 2000; Martinez, Fouletier et al., 2001), leveraging resources across the supply chain (Bowersox et al., 2005) etc. Today more than 24% of SMEs in UK are looking for collaboration to gain these benefits (http://www.aldermore.co.uk/about/news-press-releases/2013/03/smes-embrace-collaboration-to-weather-tough-economic-conditions/) Collaboration enables SMEs to focus on enhancing their core competencies through innovation whilst benefiting from the expertise of others, with complementary capabilities in the collaboration.

VEs nowadays play a vital role in the world economy. In the UK alone there are around 4.5 million businesses providing employment for 13.8 million people and £1.5 trillion turnover. Therefore, VEs have become the key research issue in the field of enterprise collaboration. Various researchers and research projects have investigated VEs
to identify collaborative issues and to facilitate smooth collaboration among SMEs. Some of the most highly promoted ones are the North American NIIIP (National Industry Information Infrastructure Protocols) (NIIP Inc., 1998) and the European PRODNET II (Production Planning and Management in a Virtual Enterprise) (Camarinha-Matos and Afsarmanesh, 1999). These projects intend to develop information infrastructures and supporting platforms for VEs and efforts are focused on reusing, extending and integrating various industry standards and enabling technologies for developing Enterprise architecture, Enterprise architecture framework its model and modelling languages to facilitate collaboration and to integrate different business processes. Still, VEs need to address broader issues throughout their life cycle (Yang and Liu, 2008) such as risk, interoperability, partner selection, knowledge management etc. to achieve minimum risks with maximum resilience.

This thesis explores the various problems faced by SMEs and VEs in the VE-lifecycle. Next section presents the research issues in the VE-lifecycle and chapter 2 identifies the research gaps which need to be addressed in this thesis to facilitate the efficient formation and operation of VEs.

1.3 RESEARCH ISSUES

Rapid advancement in the field of Information and Communication Technology (ICT) has expedited VE collaboration through sharing of resources, processes, core competency, skills etc., through knowledge and information sharing. Therefore, knowledge management (KM) constitutes a critical factor for the sustainability and resilience of not only VEs (Moore and Manring, 2009) but also of individual enterprises (Erol et al., 2010) in their current and future collaborations.
Despite increased interest in the topic of VEs and the technological advancements which have been made reliable, guidelines for the proper configuration of a VE in the different phases of the VE-lifecycle are still lacking. A VE faces many different situations, difficulties and problems at each phase of its lifecycle from pre-creation to termination as follows (shown in figure 1.2):

1. How to identify potential partners (Afarmanesh et al., 2009).
2. How to determine risks associated with different partners (Li and Liao, 2007).
3. How to select optimal partners (Ip et al., 2003).
4. How to achieve seamless transfer of data information and knowledge (i.e. Interoperability) among partners during the operation phase (Vernadat, 2010).
5. How to enhance its knowledge and core-competencies and gain competitive advantage from the VE experience during the termination phase and thereby benefit future collaboration (Pollalis and Dimitriou, 2008).

![Figure 1:2 Research issues in VE](image)
1.4 Thesis Organization:

The outline of the thesis is as follows.

Chapter 2: This chapter presents the detailed literature review of the lifecycle of VE. It identifies the research gap in different phases of VE lifecycle and sets the research objectives in this thesis.

Chapter 3: This chapter introduces the concept of description logic (DL).

Chapter 4: This chapter addresses the first objective of the thesis and develops the DL based ECOS-Query system to extract implicit and explicit information from an ECOS-ontology and find the potential partners capable of forming an effective VE.

Chapter 5: This chapter addresses the second and third objectives of the thesis. It first identifies the different risks in the formation of a VE and then develops the cooperative game theoretic based revenue sharing mechanism for optimal selection of partners in the formation of the VE.

Chapter 6: This chapter deals with the fourth objective of the thesis and develops the DL-based ontology mapping technique for achieving interoperability among enterprises. Bridging axioms are derived in this method providing the logical consistency in the mapped ontology and avoiding any heterogeneity.

Chapter 7: This chapter achieves the fifth objective of the thesis and presents the ontology merging technique for knowledge enhancement of enterprises during the termination (dissolution) phase of VE.

Chapter 8: This chapter presents the implementation and results of the proposed methodologies (chapter4 - chapter7) in the form of a case study.

Chapter 9: This chapter discusses the objectives and contribution of this research and also describes the potential areas of future research.
CHAPTER 2: LITERATURE REVIEW

2.1 INTRODUCTION

This chapter presents an exploration of previous research on the critical aspects of successful VEs. A detailed analysis of the literature reviewed here firstly endorses the routes which should be followed to achieve a high performance VE and secondly sets the research objectives to address the research issues described in chapter 1. This chapter is organized as follows: section 2.2 introduces the concept of VE and explains the benefits at enterprise level and organization level. Section 2.3 and its subsections discuss the lifecycle of VE, its phases and research gaps in each phase. Section 2.4 establishes the research objectives of the thesis based on the research gaps discussed in section 2.2 and research issues discussed in chapter 1. Section 2.5 discusses the methodology adopted in this research and section 2.6 concludes the chapter with a summary.

2.2 VE DEFINITION

Both large enterprises and SMEs need to collaborate to achieve their business objectives in the current era of globalized environments (Huang et. al., 2005). Collaboration supports enterprises, especially SMEs enabling them to focus on their core competency and stay competitive in the global market (Sieber and Griese, 1998). The research by Li et al., (2006) further emphasizes the requirement of enterprise collaboration by confirming that an enterprise’s competency increases with collaboration by increasing its flexibility and agility to adopt new business ideas.

In literature, various definitions of VE have been proposed (Camarinha-Matos and Afsarmanesh, 1999; Travica, 1997; Martinez et. al., 2001). The general definition of a VE can be given as: “A VE is a collaborative network of independent enterprises,
possibly globally located, and linked by information technology (IT) to share skills, costs and to get access to the bigger markets (Eversheim et. al., 1998). In other words, a VE is the temporary union of enterprises, organizations or individuals for achieving common goals (Zhang et. al., 1997) and fit for the current market conditions (Goldman et al., 1995). All the other definitions provided in the literature about VEs (Kosanke et. al., 1999; Gibbon, 1999) agree with the following characteristics:

- It is a collaboration of independent enterprises.
- It is temporary and opportunity driven.
- Its aim is to grasp opportunity, improve global competitiveness and gain tangible and intangible benefits.
- It utilizes the skills, resources and core-competencies of member enterprises within the VE to achieve quality, reliability, cost effectiveness, shorter product time cycles and mass customization of the products i.e. common goals of the VE.
- It uses ICT for data, information and knowledge transfer.

All other characteristics can be seen as more specific illustrations of the above mentioned general characteristics. Enterprises involved in a VE, realize both short and long term benefits (Chen et. al., 1998) to tackle global and dynamic business pressures. Several researchers have discussed the benefits and advantages that VEs provide at both the collaborative level (VE level) and individual level (Enterprise level) (Filos and Ouzounis, 2003; Mowshowitz, 2002). Some of the general benefits can be listed as:

2.2.1 VE-Level benefits:

1. Agile and reconfigurable collaboration.
2. Customer focused collaboration as a VE is formed in accordance with the market scenario.
3. Shorter product development time, lead time and optimized costs due to the high competency level of member enterprises.
4. High quality of products and services due to the specialized enterprises collaborating in the VE.
5. High customer satisfaction due to the high quality and on time product delivery.

**2.2.2 Enterprise-Level benefits:**

1. Member enterprises in a VE enjoy the broader market and benefits, which would not be possible for an enterprise acting on its own.
2. Enterprises are more focused towards their core-competency.
3. Enterprises are able to provide a wide range of products and services with limited individual resources.

The aforementioned advantages of VEs compel enterprises especially SMEs to form VEs with the support of ICT not only to survive but also to achieve competitive edge and technological advancement (Jagdev and Thoben, 2001). The advantages gained by VEs are due to the high degree of collaboration and proper coordination among the enterprises (Tolle and Bernus, 2003), but paradoxically, lack of proper collaboration and coordination are also the major reasons for the failure of a VE (Filos and Ouzounis, 2003).

One of the major characteristics of a VE is independency and core-competency of its member enterprises where enterprises perform their specialized tasks and wait for others to achieve the final goals. On one hand, lack of trust, incompetent partners,
incapable or unwilling partners and other risk factors in selecting appropriate partners plays a major role in whether or not a VE is successful. On the other hand operational level difficulties, such as interoperability (Kim et. al., 2006) and knowledge transfer can hinder the VE during its operational phase. Apart from that, a VE can be characterized as opportunistic collaboration coupled with innovation, cost effectiveness and mass customization. Therefore the future of enterprises and consequently of VEs is directly dependent on the advancement of technology and enhancement of knowledge (Gunashekrann and Ngai, 2007) and enterprises should continuously enhance their knowledge to survive in the market.

Unlike traditional collaborations, enterprises in VEs face different collaborative problems due to the presence of autonomous, non-hierarchical and independent enterprises with different technological and cultural environments. Collaborative problems for both VEs and enterprises vary at different stages of their life-cycle and if not addressed adequately at the stage where the problem arises, further and different problems will arise at subsequent stages. For example, improper partner selection at the VE creation phase will also affect the different tasks at the operation phase. The success of a VE depends on the successful execution of its life-cycle and close monitoring is essential throughout its life-cycle (Yang and Liu, 2008). Therefore, the VE lifecycle needs to be evaluated from the perspectives of both the VE and its member enterprises i.e.:

1. To achieve a successful VE.
2. To achieve success and advancement of member enterprises.
2.3 VE-LIFECYCLE

In the literature many attempts have been made to classify the VE lifecycle (Nayak et. al., 2001; Parunak, 1997). An important characterization and classification was provided by Grenier and Metes (1995) and Tolle and Bernus (2003) used a holistic view approach. The holistic view approach analyzes the different requirements and functioning of a VE and the roles of individual enterprises in terms of cooperation, communication and coordination, since these factors are essential for the success of a VE.

Different authors have provided different views regarding the lifecycle of a VE, starting from three phases view (Parunak 1997): Design or creation, Management or Operation and Disbanding or dissolution to four phases views (Camarinha-Matos and Afsarmanesh 1999): Creation, operation, evaluation and dissolution. In general the process of formation of a VE starts with the identification of a market opportunity. Next, member enterprises with complementary competencies and a common goal to exploit the market opportunity form the VE. In the next stage, individual tasks are carried out by the enterprises with full coordination and cooperation in order to achieve common goals. Finally, the VE is disbanded after achieving its goals. The Synergy project (Synergy, 2010) presents a holistic view of the VE-lifecycle in terms of its functioning and requirements and divides the whole VE life cycle into four phases:

1. Pre-creation
2. Creation
3. Operation
4. Termination
2.3.1 Pre-Creation

The pre-creation phase of the VE lifecycle starts with the market opportunities identification. These opportunities are then evaluated from the view point of profit and a suitable opportunity is selected by a leading enterprise, sometimes called a broker (Bremer et. al., 2001; Katzy and Dissel, 2001). The broker enterprise further investigates the opportunity and identifies the different requirements such as competency, skill, technology, capacity, etc. for the formation of a VE. The final step of this phase is to determine a group of enterprises that can fulfill the requirements and move on to the next phase of the VE lifecycle i.e. creation for further negotiation and VE formation.

The Pre-Creation phase of a VE lifecycle can be divided into four major steps. These steps are:

1. Identification of the market opportunities.
2. Selection of the best opportunity and identification of its requirements i.e. core competency.
3. Identification of a group of enterprises fulfilling these requirements.
4. Start the process of VE creation and partner selection after gathering information about the potential member enterprises, which were identified in step 3.

The first two steps in this phase are the defining factors of a VE but ironically enterprises have little or no control over them. The last two steps are very critical for the formation of a suitable VE and exploitation of a market opportunity. Enterprises eager to participate in the VE should provide information regarding their competency, resources etc. for assessments against the VE requirements. Two different but overlapping approaches have been reported in the literature for the identification of suitable enterprises:
(i) Formation of collaboration pools (CP) (Synergy, 2010) and Virtual breeding environments (VBE) (Afsarmanesh et al., 2009). In the similar line of approach Bremer (2000) proposed the development of virtual industry cluster (VIC) for identifying SMEs competencies and virtual enterprise broker (VIB) responsible for business opportunity and partner selection. In CP and VBE, a number of enterprises are registered as members and a group of appropriate enterprises are selected from them for further negotiations in the creation phase. Although, various supporting mechanisms have been suggested for efficient VE creation using these methods (Afsarmanesh and Camarinha-Matos, 2005), these approaches still have two major drawbacks: (i) enterprises tend to make their information available using different unstructured means such as webpage, business card, industrial profile etc. and extracting useful information from these requires great human effort. Although some enterprises provide ontology based information, the lack of standardized terminology, structure or explicit data still greatly increases the required human efforts. (ii) Enterprises registered in a VBE or CP may not be the optimal choice for the formation of a VE i.e. some enterprises outside the VBE or CP may be more suitable partners for an efficient VE.

(ii) Development of an enterprise ontology for electronic based (web, computers etc.) sharing of information. Many important enterprise ontologies have been developed by researchers such as Toronto Virtual Enterprise (TOVE) (Fadel et al., 1994), Open Information Model (OIM, 1999), Computer Integrated Manufacturing Open System Architecture (CIMOSA) (Gransier and Schönewolf 1995; Kosanke 1995), Business Processing Modelling Language (BPML, 2002), Collaborative Network Organization (CNO) (Plisson et al.,
An Ontology defining the knowledge and information of enterprises needs to use well defined and common terminology which is accepted by fellow enterprises. Another important factor it needs to possess is to have a well-defined structure and formalism for automatic processing by computers.

In order to overcome these issues enterprises need to publish their individual profile information in a structured and explicit manner on the web. Such profiles should contain core-competency areas, knowledge, resources, skill, experience and other relevant information. For efficient sharing and reuse, the terminology used to describe a company profile should be standardized. The Enterprise Competency Organization Schema (ECOS) (Khilwani et al., 2011) provides the cornerstone for developing a standard and organized means of publishing enterprise information. ECOS provides ontology based knowledge and information representation using common and standard vocabularies understandable by both humans and machines. ECOS uses web ontology language (OWL), a DL based ontology, to develop, maintain and publish information in machine readable resource description format (RDF) or Extensible Markup Language (XML) format (Khilwani et al., 2011).

ECOS provides various benefits over other enterprise ontologies such as: (i) it captures enterprise competencies with well-defined and standardized vocabulary (ii) it is both human and machine readable due to RDF/XML + OWL format and OWL provides the DL based concepts and role interrelations (Baader et. al., 2003) (iii) it can be further expanded to suit individual requirements. However, querying of enterprises based on their published information and VE requirements is still a major challenge. The process of extracting information from an ECOS ontology is achieved by querying the ontology using the RDF graphs. Although an RDF based query can extract simple information from an ECOS ontology (Khilwani et. al., 2011), it will fail when the required
information is not explicitly mentioned within the ontology. To find suitable enterprises according to the VE requirement, the concepts and role relations in the ontology need to be evaluated to extract implicit knowledge and information. Therefore, a challenging issue for any ontology is how to optimize queries of its database (Haarslev and Moller, 2004) for extracting implicit knowledge and information.

Query answering in DL has received considerable attention in recent years. Efficiency of queries in DL largely depends on its expressivity. Expressive DL provides a rich means for knowledge representation but generally incurs high computational cost (Rosati, 2007). Basic query languages such as structured query language (SQL) or Datalog allow DL-databases to ask object instances or role instances and they return or join pieces of information to answer queries. Due to the lack of expressiveness queries cannot directly be expressed in DLs (Beeri et. al., 1997). Therefore a novel approach should be added to make the DL-query efficient (Perez-Urbina et. al., 2010). Different approaches have been adopted for efficient querying such as view based query answering approach in DL (Calvanese et. al., 2012), knots based query answering (Eiter et. al., 2012), query rewriting based on constraint handing approach (Pérez-Urbina et. al., 2010), optimization of query answering (Sirin and Parsia, 2006). Worldwide web consortium (W3C) suggested SPARQL as a RDF based query language for web based ontologies such as OWL. At present SPARQL-DL (Sirin and Parsia, 2007), a substantial subset of SPARQL, provides more expressive DL queries than existing DL-query languages (Kremen and Sirin, 2008). SPARQL-DL allows TBox, RBox and ABox (chapter 3) queries for exploring concept, role interrelation and instance checking and instance retrieval. Although SPARQL supports many of the DL-queries, it still does not support the data type complex queries and negation of concepts (roles) where negation is
not defined explicitly in the ontology. DL based queries therefore still need to extract implicit knowledge and to compute queries automatically.

2.3.1.1 Research Gap in pre-creation phase:

The detailed review of research relating to the Pre-Creation phase of VEs reveals that the identification of suitable partners requires the development of a DL-based query method (ECOS-DL query) which can extract both explicit and implicit information from an ECOS-ontology, since with a set of suitable partners, the VE can enter into its next lifecycle stage i.e. the creation phase for further negotiation.

2.3.2 Creation

The creation phase in the VE lifecycle determines the formation of a VE from the partner enterprises in response to the market opportunity and VE requirements. The pre-creation phase establishes the goal and objectives of the future VE according to the market conditions and also identifies different requirements which need to be fulfilled. The proposed ECOS-DL query system can identify a set of appropriate enterprises capable of fulfilling the requirements but in the creation stage only a few of them will be selected through a partner selection process to form the VE.

Partner selection occurs once the broker enterprise has identified the key core-competency, tasks, skills and capacities needed from the prospective partners and the ECOS-DL query system has identified the set of capable enterprises. In this context, the key issue at this phase is to determine the appropriate partner selection mechanism for VE formation. This is arguably the most critical step at this stage as selecting the appropriate partners is the central issue for the success of the VE (Crispim and Sausa, 2009) and inappropriate partners can lead to the failure of the VE (Wildeman, 1998) due to different risk factors associated with them (Hallikas et al., 2004).
Partner selection problems for VEs, using mathematical programming methods, have been tackled by optimizing from a single criterion (the overall cost of the VE) to multiple criteria (cost, due date, risks etc.). Ko et al. (2001), Ip et al. (2004) and Wu and Su (2005) proposed the integer programming models with the objective of minimizing the total costs including production, operation and transportation costs. Talluri et al. (1999) proposed a two-phase approach for partner selection by designing a VE and considering the factors of cost, time and distance. However, the precedence relations between the sub-projects were not considered. Davis and O’Sullivan (1999) pointed out that precedence relations between sub-projects are important factors, and from this viewpoint, project scheduling should be embedded in the partner selection (Brucker et al., 1999). Subsequently, Ip et al. (2004) considered the precedence of sub-projects with due date and cost factors for partner selection and solved it by a branch and bound technique. Zeng et al. (2005) proved that the partner selection problem embedded with sub-project activity and precedence relation is an NP-hard and used non-linear integer programming. Jarimo and Salo (2009) further added the variable cost of operation and transportation cost in the mixed integer linear programming (MILP) for partner selection. De Boer et al., (2001) divided the partner selection process in three subsequent stages, starting from criteria formulation stage to criteria qualification stage and finally partner selection stage based on best qualified ones.

Artificial intelligence (AI) techniques and qualitative approaches considering competency and management style (Hitt et al., 2004; Mitsuhashi, 2002) and knowledge, reputation and skill (Pidduck, 2006; Hitt et al., 2004) have also been adopted in partner selection problems. Different AI techniques used for optimizing the partner selection problems are Particle Swarm Optimization (PSO) (Wang, 2009; Gao et al., 2006), Ant Colony Optimization (ACO) (Wu and Liu, 2007; Kang et al., 2007), Genetic Algorithm
(GA) (Ip et al., 2003; Ma et al., 2007) and quantum evolutionary algorithm (Tao et al., 2010). Different Multi-criteria decision making techniques have also been used in partner selection frameworks such as Analytical Hierarchy Process (AHP) (Sari et al., 2007), Fuzzy set theory and AHP (Cao et al., 2004, Cao and Zhou 2006, Mikhailov 2002), Fuzzy set theory and clustering (Dai and Yang, 2005), Fuzzy set theory and critical path analysis (Huang et al., 2005), Fuzzy decision making (Ye and Li 2005, Ren et al., 2007) and a Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) (Ye and Li, 2009). TOPSIS and Fuzzy techniques rank the alternative VE configurations based on selection criteria, which increase the decision flexibility in changing environment conditions (Crispam and Sausa, 2009).

Another important aspect in partner selection is risk analysis (Huang and Chen, 2005; Li and Liao, 2007). As VEs are complex network composed of many stand-alone SMEs to explore market opportunity, inefficient knowledge sharing, changing of market condition, lack of cooperation and inefficient partners in terms of resources can cause different risks (Brunelle, 2009). Hence risk management is a key factor to ensure success of VEs (Huang et al. 2013). Ip et al. (2003) studied the risk factors along with due date, cost and precedence relation with conversion of risk factors into additional costs. The combination of various selection criteria has also been analyzed by researchers such as risk factors and due date (Yang et al., 2006; Zhao et al., 2006), operational cost (Goankar and Viswanadham, 2004), operational cost and transportation cost (Ko et al., 2001), operational cost and processing time (Wang et al., 2001; Wu and Su, 2005), operational cost and due date (Zeng et al., 2005; Huang et al., 2005), operational costs and risk factors (Gao et al., 2006).

The current approaches in partner selection problems for the VE paradigm use the different operational costs, due date, risks and other quantitative and qualitative methods.
in parallel with single organizational approaches. However, the selection of the right partners is a difficult and crucial process (Crispim and Sausa, 2009) due to the behavioral differences of VEs from traditional organizations. Member enterprises needing to synergize not only core competency but also management style, corporate cultures, cooperation, knowledge and information transfer etc. Another very important aspect in the partner selection is risk analysis. Although, at individual levels member enterprises of VEs are capable of exploiting the advantages which come from being small (i.e. they tend to be innovative, proactive and so forth), nonetheless, due to the cumulative effect of the risks of collaborating within a VE, higher degrees of risk are experienced by individual enterprises when compared to traditional organizations due to the following factors:

1. Complex connection and interaction among enterprises.
2. Dynamic environment (selection criteria changes with project).
3. Incomplete or non-availability of information.

Due to such peculiarities, risk analysis is of foremost importance in VE formation. Das and Teng (2001) stated that there are two main kinds of risks in alliances, i.e., performance risk and relational risk. Performance risk is related to the probability that a VE may fail even when member enterprises commit themselves fully. Relational risk is concerned with the probability that member enterprises lack commitment to the VE and that their possible opportunistic behaviors could undermine the prospects of the VE. Undoubtedly, enterprises collaborate to form a VE in order to reduce the performance risk but this can increase the relational risk. Without any controlling mechanism, it is highly likely that member enterprises will tend to be interested more in pursuing their self-interest than the common interest of the VE. If this is the case then an individual enterprise’s self-interest comes at the cost of other member enterprises’ interests and even the VE’s interest. Opportunistic behavior leads to shirking, appropriating the member
enterprise’s resources, distorting information, harboring hidden agendas, and delivering unsatisfactory products and services. As these activities seriously jeopardize the viability of VEs, analyzing overall risk is an important component of the successful VEs.

As explained earlier, the general structure of a VE is very different from traditional enterprises. The absence of a centralized control system increases complexity further with the absence of controlling mechanisms which can determine the level of effort which is needed by member enterprises. As a result partners are subjected to moral hazard and awarding each partner a fixed share of profit invites undersupply of effort. Therefore, an appropriate mechanism is needed to address the following issues from both VE and enterprise perspectives:

1. Different risk factors
2. Effect of risks on both VE and individual member enterprises.
3. Determining the effort put forward by enterprises in terms of risks and value addition.
4. Division of revenue based on the effort provided by the enterprises.

Since VEs are profit driven, a key issue for the successful running of VEs is the construction of a reasonable and efficient revenue sharing mechanism based on effort, risk shared and value addition which can prevent some members from gaining profit by harming others’. Such a mechanism will facilitate not only the formation of robust VEs but will also enable enterprises to analyze associated gains and risks.

A VE, which is characterized as a coalition of enterprises with a basic common goal, combined with a complex decentralized decision and control system is of major interest and relevancy in game theory, under the classical decomposition into cooperative (or coalitional) games and non-cooperative (or strategic) games. Models referred to as ‘non-cooperative’ are composed of players with different preference relations or utility
functions. Their actions obey a strategy that takes into account information (generally imperfect) of the other players’ actions and preferences. Although, different networks have been studied under game theory along with non-cooperative approaches (Cachon and Netessine, 2004), cooperative game theory seems to be more appropriate for the analysis of a VE in its partner selection process.

In general, the characteristics of cooperative games are based on the assumption that players can obtain a larger global benefit from pooling their resources than by acting separately. This characteristic is particularly important when enterprises are complimentary, as in the case of a VE and the maximum profit can only be achieved by proper cooperation. Several authors have used cooperative game theory to represent alliances in network analysis, Nagarajan and Soˇsic´ (2008) studied the benefits expected by retailers from price setting, pooling their market share and sharing risks. The works of Granot and Soˇsic´ (2003), Cachon and Netessine (2004) and Reinhardt and Dada (2005) provide convincing interpretations of supply chain design problems as cooperative games by focusing on manufacturing capacity and inventory pooling. Hsieh and Lin (2012) used the reverse auctions method to minimize the overall cost in VE partner selection process. In the light of cooperative game theory, VE partner selection can be modeled as a coalition of optimal partners pooling their resources and sharing the same utility function (profit) and can be termed as a cooperative game with transferable utilities (TU-game). The cooperative game is first characterized by global optimization of the total value (total expected profit) and this can only be achieved by optimal selection of partners. Next, a TU-game is realized, which decides the shares of the global profit acceptable to all of them based on their effort (value addition and risk sharing). Another practical advantage of this model is that it evaluates and compares different possible coalitions. The results
can thus be used as arguments in the VE partner selection stage, to convince partners to be part of the best possible coalition and set up the joint venture.

### 2.3.2.1 Research Gap in Partner Selection phase:

Based on the above literature review, partner selection in the VE paradigm requires analysis of the different risks from a collaborative network perspective and development of mechanism to determine the input put forward by individual enterprises. Although various authors have analyzed the risk factors for partner selection (Huang and Chen, 2005; Li and Liao, 2007), their analysis is all based on extensions of the single enterprise environment. As VEs are different from individual enterprises in terms of functions, structures and complex interrelations, it is important to analyze risk factors from both VE and individual enterprise perspectives i.e. (i) the effect of an enterprise’s risks on other member enterprises (VE perspective) (ii) the effect on an individual enterprise of risks from other member enterprises and (Enterprise perspective) (iii) the overall network risks (VE perspective). This approach to risk analysis will provide the measures of enterprises’ input in the VE and based on these inputs an effective revenue sharing mechanism needs to be developed from a cooperative game viewpoint, for efficient partner selection.

### 2.3.3 Operation

The operation phase in a VE is the business process phase where the virtual organization is producing an end product. The operation phase aims to execute the operational requirement of the VE to obtain the common goals realized in the pre-creation phase. The operation phase of a VE starts with task distribution, with analysis of task sequencing, task interdependency, precedence relations (if any) and other operational issues. It carries out its operational requirement through partners’ collaboration support
tools which facilitate the data, information and knowledge transfer among the member enterprises.

In a VE, each member enterprise will have a specific expertise, and will execute their own tasks but will have little or no influence over the way that other specific tasks are carried out. However, enterprises need to communicate, cooperate, collaborate and interoperate with other member enterprises situated locally or globally to address the interdependency issues. Therefore, a VE needs proper communication and cooperation for efficacious collaboration and successful execution of the operation phase.

The effective and efficient collaboration in a VE requires agility and interoperability (Chen et al., 2008) and in the present environment of VEs, interoperability is the major issue of concern (Kim et al., 2006). The interoperability means seamless communications among the business components which can be shared and exchanged in order to cope with unanticipated events in the business environments (Kim et al., 2006). In the context of networked enterprises, interoperability refers to the ability of interactions (exchange of information, services and data) between enterprise systems at these three levels: Physical, Application and Business (Chen et al., 2008). Interoperability enables VEs to respond quickly to the rapidly changing business environment (Kim et al., 2006). The advantages of interoperability in a VE have been reported by many researchers (Panetto and Molina, 2008). Interoperability is a major tool for enterprise coordination which aligns the business processes and resources so that the VE can efficiently and effectively execute its objectives as defined by common consent (Kosanke and Nell, 1997; Petrie, 1992; Vernadat, 2007). Interoperability provides the right information, knowledge and data at the right place at the right time with right meaning to support business process operations (Gold-Bernstein and Ruh, 2005) and therefore, enhances the overall productivity, flexibility, and reactivity of the VE.
According to the European interoperability framework (EIF, 2004), difficult stages for interoperability are at two levels: 1. Technical level and 2. Semantic level. Technical level interoperability is related to the transfer of knowledge and data, whereas semantic level interoperability is related to the meaning of the knowledge. ICT has enabled enterprises to transfer knowledge or data at a technical level (Vernadat 2010) but proper semantic interoperability is still a challenging issue (Tursi et al., 2009). Semantic interoperability ensures that the information transferred between the enterprises should be understood with the correct intension. However, in a VE it is likely that each enterprise has its own knowledge system and terminology, which provides particular meaning to its information. It is also possible that an enterprise may have different collaborations or networks including other VEs. Hence it is not possible to fully change their existing terminology and accept a new one in accordance with a new VE. Therefore, a flexible and interoperable approach is needed to accommodate this.

The first step adopted in the process to achieve semantically aligned knowledge transfer was to develop the ontology based knowledge base (Wajid et al., 2013, Tursi et al., 2009), i.e. ontology based information, data and knowledge acquisition and transfer. Ontologies facilitate the representation of shared concepts in a domain or across domains by specifying a set of terms to ensure proper communication between the enterprises. Ontologies explicitly represent the data along with their semantics to facilitate proper information transfer; however, it is very difficult and probably impossible to develop a single, universally accepted, ontology, defining the whole universal system (Pollalis and Dimitriou, 2008). Enterprises are developing their own ontologies based on individual requirements (manufacturing, logistics etc.) and use different terminologies for the concepts and properties. Collaboration among such cross domain enterprises (such as manufacturing, sales, marketing etc.) leads to semantic heterogeneity, in spite of ontology
based knowledge systems (Jung, 2008), and this still remains a key issue for VE interoperability.

As mentioned earlier, different enterprises are developing their own ontologies, in most cases independently, to describe the same, different and overlapping domains. Different kinds of heterogeneity exist such as, lexical heterogeneity (same concept defined in different terms or same terms defining different concepts), structural heterogeneity (difference in degree of detail or granularity) (Wang and Liu, 2009) etc. In order to achieve proper semantic interoperability, ontologies need to be synchronized i.e. ontology mapping and matching. Ontology mapping finds the correspondences between entities (concept, relation, individuals) among the different enterprises. In literature, various approaches have been reported for ontology mapping (sometimes referred to as alignment) (Chungoora and Young 2008) ranging from manual (Hu et al., 2008) to semi-automatic (Chen et al., 2011). Nonetheless, simple correspondence between entities can create an erroneous or inconsistent mapping.

Various heterogeneities have been reported in ontology mapping (Wang and Liu, 2009) and the reported types of mismatches are as follow:

1. Synonymy conflicts: Same concepts defined by different terms.
2. Polysemy conflicts: Different terms defined by the same term.
3. Subclass conflicts: Occur when the same class in different ontologies is divided into different subclass concepts (i.e. difference in the granularity).
4. Class-Role conflicts: Occur when a class in one ontology is described by a role or properties in another ontology.
5. Class Coverage conflicts: Occur when a class defines the same concept in two ontologies but one class covers a broader domain than the other.
6. **Role conflicts**: Occur when the same class in different ontologies is described by different properties (roles).

7. **Role Attribute conflicts**: Occur when a class and its role are the same in two ontologies but their value types (attributes) differ.

Considerable effort has been put into ontology mapping research in order to provide interoperability and resolve conflicts (as described in the previous section). A comprehensive review of current approaches in ontology mapping has been provided by Euzenat and Shvaiko (2007). Initial work on ontology mapping was focused mainly on the string distance and overall nomenclature of the ontologies. This approach commonly leads to synonymy and polysemy conflicts as cross domain ontologies or even similar domain ontologies often use different taxonomies. Ontology mapping systems, such as FCA-merge (Stumme and Madche, 2001) and T-Tree (Euzenat, 1993) tried to resolve this issue and explore the subclass- superclass relationships along with lexical similarity for ontology mapping. Various other approaches have also been applied in ontology mapping and alignment as shown in Table 2.1.

Nowadays, especially in the VE paradigm, enterprises can use OWL as a prominent tool for storing, using and transferring data and knowledge through the web. OWL is based on the (DL, a fragment of first order logic (FOL). CtxMatch (Bouquet *et al.*, 2006) and S-Match (Giunchiglia *et al.*, 2004) tried to determine semantic matching with inconsistency deduction using the DL axioms. In these approaches inconsistencies were detected using the unsatisfiability of the equivalence and sumsumption relation but still simple correlation between the terms may cause the heterogeneity even in the DL based ontology mapping (Dou and McDermott, 2006).
<table>
<thead>
<tr>
<th>Name</th>
<th>Approach</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLUE</td>
<td>Taxonomy and concept information based approach</td>
<td>Doan et al. (2003)</td>
</tr>
<tr>
<td>COMA</td>
<td>Structural similarity matcher based approach</td>
<td>Massmann et al. (2006)</td>
</tr>
<tr>
<td>Falcon-AO</td>
<td>Graph based mapping approach</td>
<td>Hu et al. (2006)</td>
</tr>
<tr>
<td>IF-Map</td>
<td>Formal concept analysis based approach</td>
<td>Kalfoglou and Schorlemmer (2003)</td>
</tr>
<tr>
<td>S-Match</td>
<td>Deductive technique based approach</td>
<td>Giunchiglia et. al. (2004)</td>
</tr>
<tr>
<td>PROMT Suite</td>
<td>Syntactic concept level matching</td>
<td>Noy and Musen (2003)</td>
</tr>
<tr>
<td>CtxMatch</td>
<td>Description logic based subsumption approach</td>
<td>Bouquet et al. (2006)</td>
</tr>
</tbody>
</table>

**Table 2:1 Different Approaches in Ontology mapping**

Mismatches in ontology mapping are due to assigning simple correspondence between entities which creates an erroneous or inconsistent mapping (Dou and McDermott, 2006). The simple correspondence between the terms of two ontologies
commonly causes inference problems. This can be explained as: assuming \( P, Q \) are two terms in two ontologies, with simple correspondence \( P \rightarrow Q \) then \( \{ \mathcal{KB}, P, P \rightarrow Q \} \vdash Q \) i.e. \( Q \) can be inferred using knowledge base \( \mathcal{KB} \), \( P \) and axiom \( P \rightarrow Q \), but \( \{ \mathcal{KB}, P, P \rightarrow Q \} \nvdash \neg P \rightarrow \neg Q \), i.e. \( \{ \mathcal{KB}, P, P \rightarrow Q \} \nvdash Q \rightarrow P \). In general terms the fact that the \( Q \) can be inferred from \( P \) does not automatically mean that \( P \) can be inferred from \( Q \) and trying to infer this causes the error in the ontology mapping. Such heterogeneity in the ontology mapping is caused by considering the correspondence but not analyzing their relationship, such as more general (\( \supseteq \)), less general (\( \subseteq \)), equivalence (\( \equiv \)), disjoint (\( \perp \)), overlapping (\( \cap \)) or union of other entities (\( e_1 \cup e_2 \cup \ldots \)). Defining such relationships in the mapping can prevent incorrect inference, For example if the mapping finds a correspondence such as \( P \subseteq Q \), this gives the \( \mathcal{KB} \) two axioms: \( P \rightarrow Q \) and \( \neg P \rightarrow \neg Q \) (not \( P \) does imply not \( Q \)) (Kumar and Harding, 2013).

2.3.3.1 Research Gap in Operation phase:

Current literature indicates that there is a spectrum of methods that rely on lexical similarity matching, which from a semantic interoperability viewpoint is not optimal (Dou and McDermott, 2006) as different words can define same concepts while the same words can define different concepts (Wang and Liu, 2009). Two similar concepts defined in different ontologies can have different domains of interpretation. The last section identified the different mismatches in ontology mapping and explained the reason for logical inconsistencies in simple correspondence between the entities. Therefore, it is important to develop a logic based ontology mapping technique to achieve interoperability among enterprises in VEs. Moreover, where VEs are relying on the web for knowledge and data transfer, it becomes imperative to consider ontology mapping in the DL paradigm. This is because the web ontology OWL, is based on DL,
hence concepts defined in different ontologies using OWL need to be brought under a single umbrella to achieve interoperability by exploring both semantic and logical definitions of different entities in ontologies.

2.3.4 Termination

VEs are terminated or disbanded once they have achieved their goals and they are no longer effective in the current market scenario. Members of the disbanded VE will look for other market opportunities for the formation of future VEs i.e. beginning of another VE lifecycle. One of the major issues at this stage is the dissemination of the knowledge acquired during the VE lifecycle (Pollalis and Dimitriou, 2008).

The previous section explained that a successful VE needs to develop a mechanism for seamless transfer of data, information and knowledge among member enterprises (Kim et al. 2006). ICT can help to achieve collaboration in VEs at a technical level (Singh, 2005) whilst ontologies have been proved to be important tools at the semantic level. Generally each individual enterprise builds its own ontology, based on the domain of their operation, to represent the enterprise’s knowledge. It is imperative that an enterprise ontology should possess two features: 1. Interoperable: to be able to collaborate with other enterprises and 2. Maintainable: to preserve current knowledge and to accommodate new knowledge. If an enterprise ontology is interoperable, information and knowledge which is transferred using ontologies can be understood accurately, i.e. with correct intention and extension (Gold-Bernstein and Ruh, 2005). However, as ontologies may be developed independently to suit personnel requirements, it is impossible to avoid heterogeneity in the terminology used for concepts and their relation and mappings between ontologies are required to interrelate different concepts and to achieve interoperability. Many mapping techniques have been adopted and
proposed, in the literature, to achieve uniformity and to tackle interoperability of the enterprise ontologies (Pollalis and Dimitriou, 2008). Last section discussed the interoperability issue in VE and this section discusses the maintainability issue.

Nowadays knowledge has become one of the most precious resources for any enterprise (Sorli et al., 2006). However, this knowledge is more valuable if it can be made inferable and deducible. The future success of enterprises is coupled with their knowledge assets so enterprises need to accumulate knowledge (or create knowledge) from information e.g. by updating their knowledge in the form of ontology. According to Mo and Zhou (2003), knowledge is power and its proper management is necessary to preserve valuable content, learn new things, solve problems, consolidate core competency and discover and implement new technologies. Enterprises should be able to maintain their ontologies to accommodate new knowledge to stay competitive and successfully collaborate in VEs not only in the current time but also in the future. For this reason, the maintenance of the ontology is a continuous or on-going aspect of VEs.

Ontologies definitely play important roles in KM (Ku et al., 2008), but the knowledge discovery possess is equally important to identify and accommodate new knowledge within existing ontologies. The discovered knowledge will not be useful unless it is mapped semantically and structurally with the existing ontologies. To merge knowledge correctly, both the syntax and semantics must be considered, in order to:

1. Deduce similar or new concepts
2. Deduce the possibility of merging concepts, i.e. by restructuring an ontology.
3. Achieve logically consistent mappings

Exhaustive surveys have been carried out on KM (Gunashekaran and Ngai, 2007; Kebede, 2010) and its tools (Liao, 2003). KM in enterprises is mostly tackled at the subjective level and this can be divided in three different stages: 1. Knowledge creation
2. Ontology development for new knowledge and 3. Merging new knowledge in the existing sources.

Due to the widespread application of different information systems, a large amount of different knowledge is accumulated during collaboration between enterprises. One of the most important factors in KM is knowledge discovery. Proliferation of data has created a completely new and different area of KM (Sun et al., 2008) requiring the extraction of knowledge from abundant data and the organization and merging of this knowledge with existing knowledge. Knowledge discovery includes discovering implicit knowledge from the data, often using data mining techniques to extract knowledge from data sources. Exhaustive literature surveys illustrate that knowledge management frameworks, knowledge-based systems (KBS), ICT, artificial intelligence and expert systems, database technology etc. have all been adopted by enterprises to exploit knowledge in order to solve their current problems and enhance their expertise. A detailed review has been done by Liao (2003). Pollalis and Dimitriou (2008) first proposed the different initiatives needed for knowledge creation and then developed the requirements at each stage of the KM-lifecycle.

Many researchers have proposed different methodologies for ontology creation from new knowledge. Huang and Diao (2008) proposed a methodology for creating a Concept Map based ontology construction method for knowledge integration. This accumulates knowledge in the business processes and rules and constraints are implemented using SWRL (semantic web rule language). However to implement this in the VE scenario, enterprises need to reconstruct their ontology every time they move to a new collaboration. Ling et al. (2007) proposed an ontology-based method to build an integrated knowledge base from heterogeneous sources operating in a single domain. Rajsiri et al. (2010) developed a knowledge based ontology model for the collaborative
business process model. A distributed enterprise system framework for KM is developed by Ho et al. (2004). Pirro et al. (2010) developed a framework for creating, managing and sharing knowledge within an organization with a distributed functional system. Mo and Zhou (2003) developed tools and methods for managing the intangible knowledge of a VE. Ling et al. (2007) proposed an ontology based method for knowledge integration in a collaborative environment. They used heterogeneous ontologies to build a domain ontology, i.e. by merging them and through inconsistency elimination. Chen et al. (2011) used Wordnet (Wordnet API) and fuzzy formal concept analysis for merging domain ontologies. Raunich and Rahm (2011) proposed the ATOM (Automatic Target-driven Ontology Merging) for integration of multiple ontologies. The process was based on the equivalent relation between source and target taxonomy and merging them preserving the target taxonomy. PROMPT (Noy and Musen, 2000) uses the class-name similarities and relies on users for specific merge operation whereas, OntoMerge (Dou et al., 2002) uses the bridge ontology concept for ontology merging.

2.3.4.1 Research Gap in Termination phase:

It is clear from the literature survey that knowledge enhancement or competency enhancement in enterprises, i.e. merging new knowledge in the existing ones in the VEs has been given little or no attention. Although various approaches have been adopted for knowledge merging (Raunich and Rahm, 2011; Chen et. al., 2011), these methods require ontologies to be built from the beginning. Another important drawback is that little or no attention has been given to logical structures of different concepts for ontology merging and consistency checks. As VEs are relying on a web ontology such as OWL (a DL based ontology), a DL-based ontology merging method is needed to map discovered knowledge with existing knowledge by exploring the semantic and logical structure of different
concepts and, if needed, by reconfiguring the ontology to accommodate new concepts in the existing ontology with logical consistency.

2.4 RESEARCH AIMS AND OBJECTIVES

The literature review on the lifecycle of VEs has identified the major research gaps in the different lifecycle stages. The success of a VE lies in the core of strengthening its lifecycle and this research intends to provide the tools and techniques to improve the performance of VE. Keeping the research issues listed in section 1.1 and the research gaps discussed in previous sections in mind, this research aims to enable enterprises to form a high performance VE and to improve the competency of member enterprises by addressing the following objectives:

1. To develop the DL-based query system, this can extract explicit and implicit information from an ECOS-ontology to identify the potential partners in the pre-creation stage.

2. To identify a method for the analysis of different risks, this may occur during the creation phase of a VE.

3. To develop a revenue sharing mechanism for optimal partner selection in the creation phase.

4. To develop a DL-based ontology mapping technique to achieve interoperability in the operation phase.

5. To develop a DL-based ontology merging technique for knowledge enhancement of enterprises during the termination phases.
The first four objectives intend to provide a high performance VE, whereas the fifth objective enhances the competency of member enterprises in different phase of VE lifecycle as shown in figure 2.1.

![Diagram](image)

**Figure 2:1 Research issues in VE-lifecycle**

### 2.5 RESEARCH METHODOLOGY

This section outlines the research methodology adopted in this research. The research methodology is defined as a set of techniques implemented to achieve predefined goals (Welke, 1983). The research methodology also describes the sequence in which different techniques need to be performed and checks their consistencies (van den Heuvel, 2002). Figure 2.2 graphically outlines the phases of the methodology adopted in this chapter.
Figure 2.2 Research Methodology

**Phase 1:**

The first step in any research is to identify and understand the problem under investigation. Phase 1, in this research, addresses research issues related to VEs using the detailed literature review and work from the synergy project (Synergy, 2010). Phase 1 concluded by setting the aims and objectives of this research (Chapter 1 and 2).

**Phase 2:**

The identified aims and objectives from phase 1 are addressed in phase 2 in the form of a solution design. Appropriate technologies were identified and chosen including

```
Research Issues → Aims and Objectives

Phase 1

Solution Design → Case Study

Phase 2

DL, OWL, Cooperative game, Non-Linear programming → DL-Reasoners JAVA API, JAVA, Protégé, Matlab

Contribution and future scope

Phase 3
```

```
Phase 1

Literature Review → Synergy Project

Phase 2

Synergy Project

Phase 3
```
DL (Chapter 3), OWL, Cooperative game theory and non-linear programming concepts. During phase 2 the solution methodologies were proposed to achieve the aims and objectives of this research (Chapter 4-7).

**Phase 3:**

Phase 3 validates the solution methodologies proposed in phase 2. Literature review and synergy projects were used to generate data for the case study. Protégé, an OWL editor, JAVA, JAVA library for DL-reasoners, matlab were used to implement prototype software applications and obtain results in order to validate this research (Chapter 8). Finally, in phase 3, the research contribution and future scope are discussed (Chapter 9).

**2.6 SUMMARY**

It is clear from the analysis of research work on VE and VE life-cycles that different support is needed at different stages of the VE lifecycle for the VE and for its member enterprises. The research aims and objectives (section 2.4) discussed the key issues which need to be addressed for the high performance VE and its member enterprises. As the importance of VEs is increasing in the current global environment, VEs and partner enterprises need to be strengthened in terms of cooperation, interoperability, competency etc.

This thesis identifies the major problems faced by VEs and enterprises at every stage of the VE lifecycle and provides approaches to strengthen enterprises and VEs in terms of competency, interoperability and cooperation. The DL based ECOS-query system, described in chapter 4, identifies the potential partners in accordance with the VE requirements. One of the benefits of the ECOS-ontology is that it can extract implicit
information from the ECOS ontology which would otherwise require restructuring the whole ontology.

Addition of the ECOS-Query facility enables better partner selection processes to be applied in the creation phase of the VE. A Game theory based approach has been applied, in chapter 5, to develop a revenue sharing mechanism based on risk and added value factors. One of the benefits of this mechanism is that it creates a robust VE in terms of risks and also provides a platform for enterprises to understand the risks associated with a VE for decision making about whether to join VE.

In the operation phase, enterprises carry out their tasks independently but they are still dependent on the inter-related tasks information and knowledge. Therefore, interoperability is a major issue at this stage. Simple correspondence or lexical similarity is erroneous as explained in the section 2.3.3. A DL based ontology mapping technique with bridging axioms is developed, in chapter 6, for logical and consistent mapping of heterogeneous ontologies. The mapping is carried out at class, role and class and role levels to provide interoperability in the VE.

Enterprises gain knowledge during the operation phase of the VE. It is therefore imperative for enterprises to merge gathered knowledge into the existing knowledge to enhance their core-competency and stay competitive in the market. DL- based ontology mapping, chapter 7, has been proposed for merging gathered knowledge into the existing knowledge. The process exploits the different axiomatic relations for deriving the logical consistency, position and relationships between the two ontologies. The implementation of proposed methodologies has been shown in a case study in chapter 8, to demonstrate their efficiency and efficacy. Table 2.2 summarizes the relationships between the research objectives and the chapters which address them.
<table>
<thead>
<tr>
<th>Research Objective</th>
<th>VE-lifecycle stage</th>
<th>Addressing Chapters</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Chap 4</td>
</tr>
<tr>
<td>1. ECOS Query</td>
<td>Precreation</td>
<td>X</td>
</tr>
<tr>
<td>2. Risk analysis</td>
<td>Creation</td>
<td></td>
</tr>
<tr>
<td>3. Revenue sharing</td>
<td>Creation</td>
<td></td>
</tr>
<tr>
<td>4. Ontology mapping for interoperability</td>
<td>Operation</td>
<td></td>
</tr>
<tr>
<td>5. Ontology merging for knowledge enhancement</td>
<td>Termination</td>
<td></td>
</tr>
</tbody>
</table>

Table 2:2 Relationships between research objectives and chapters
CHAPTER 3 : DESCRIPTION LOGIC AND ONTOLOGY

3.1 INTRODUCTION

This chapter introduces the concept of DL, construction of ontologies based on DL and the development of OWL in the process of achieving a semantic web (W3C, 2000). DL, a decidable fragment of FOL, builds the knowledge base using various constructors for the building of ontologies. A DL knowledge base (\( \mathcal{KB} \) or \( \mathcal{KB} \)) contains the explicit and implicit information about the ontology using Terminological box (\( \mathcal{TBox} \) or \( \mathcal{T} \)) and Assertion box (\( \mathcal{ABox} \) or \( \mathcal{A} \)) and is denoted as: \( \mathcal{KB} = \langle \mathcal{T}, \mathcal{A} \rangle \). The next section describes the theory of DL with the introduction of \( \mathcal{TBox} \) (section 3.2.1), \( \mathcal{ABox} \) (section 3.2.2), Axioms (section 3.2.3), Interpretation (section 3.2.4) and Reasoning (section 3.3.5). Section 3.3 describes the development of ontologies based on DL and section 3.4 introduces the OWL, a DL based web ontology.

3.2 DESCRIPTION LOGIC (DL)

DL is logic based formal structured representation of domain specific knowledge (Baader and Nutt, 2003). It is based on a common family of languages, called description languages, which support different constructors (Schmidt-Schaub and Smolka, 1991) to build classes (sometimes called concepts) and roles (sometimes called properties). The foundation of DL is description of concepts, roles and individuals. A concept represents the class of objects sharing some common characteristics which are also called unary predicates. A role represents a binary relation, binary predicates, between objects or between objects and data-type values. Individuals are instances of concepts and roles. The descriptions of concepts, roles and individuals, are also called \( \mathcal{KB} \), and are divided in two sections: \( \mathcal{TBox} \) and \( \mathcal{ABox} \).
3.2.1 TBox of DL

The TBox or Terminological box defines the terminology or vocabulary i.e. the names of the concepts and roles in the domain of the KB. The TBox building blocks are atomic concepts and roles. Atomic concepts and roles are self-explanatory terms and are not defined using other concepts, roles and DL constructors. Complex concepts and roles are built upon atomic concepts and roles using DL constructors. TBox also contains the statements about concepts and roles relations such as equivalent, sub, super, complement, union etc.

Some authors further divide TBox into two parts: TBox and RBox (Role box or R). Here, TBox exclusively deals with at the concept level whereas RBox deals with the roles and KB is defined as: KB = <T, R, A>. Atomic concepts, complex concepts and their relations are defined in TBox while atomic roles, complex roles and their relations are defined in RBox. In this thesis TBox means the combination of TBox and RBox.

The use of constructors defines the expressivity of the DL. Less expressive sets of DL constructors provide less expressive DL due to the inability of defining complex terms and relations. Some of the general constructors of DL are: top concept (⊤) bottom concept (⊥), negation (¬), union (∪), intersection (∩), existential quantifier (∃), universal quantifier (∀), cardinality restriction (≥, ≤) etc. TBox concepts (Ci) and roles (Ri) are built from atomic concepts (Ai), atomic roles (Pi) using constructors as follows:

\[ C_i :\rightarrow \ A_i (\text{Atomic concepts}) | \top (\text{Top concept, always true}) | \bot (\text{Bottom concept, always false}) | \neg A_i (\text{Negation of a concept}) | Ci \cup C_j (\text{Union of concepts}) | C_i \cap C_j (\text{Intersection of concepts}) | \exists R_i C_i (\text{Existential quantifier}) | \forall R_i C_i (\text{Universal quantifier}) | \geq n R_i C_i, \leq n R_i C_i (\text{Cardinality restriction}). \]
3.2.2 ABox of DL

The second component of a knowledge base is ABox which introduces individuals by their names and asserts concepts and properties of these individuals. There are two kinds of ABox assertion, concept assertion and role assertion. A concept assertion relates an individual with a concept. Assuming that there is a concept $C$ and an individual $a$, $C(a)$ means that the individual $a$ belongs to the concept $C$. Role assertion relates the individuals with a role. Assuming individuals $b$, $c$ and role $R$, $R(b,c)$ means that $c$ is the filler of role $R$ for $b$ or $b$ and $c$ are associated with role $R$. The set of all $b$ is called the domain of role $R$ and the set of all $c$ is called the range.

In other words, ABox defines the individuals by unary relations (concepts) and by binary relations (roles). A role relation can also be a data type where an individual is related to a data value (numerical or string value).

3.2.3 Axioms in KB

Axioms in KB define the interrelations between concepts and roles. In terms of concepts the most common axioms are:

$$R_i :\rightarrow P_i \text{ (Atomic roles)} \mid \neg P_i \text{ (Negation of roles)} \mid R_i \sqcup R_j \text{ (Union of roles)} \mid R_i \sqcap R_j \text{ (Intersection of roles)} \mid R_i^+ \text{ (transitive role)} \mid R_i^- \text{ (Inverse role)}.$$
• General concept inclusions (GCIs): These define the subsumption relations between concepts. Such as, with concepts \( C \) and \( D \), \( C \sqsubseteq D \) means \( C \) is more specific than \( D \).

• Concept equivalence: This defines the equivalence relation between concepts i.e. \( C \equiv D \) and uses the two GCI \( C \sqsubseteq D \) and \( D \sqsubseteq C \).

• Concepts disjointedness: This defines that the concepts are unrelated i.e. \( C \cap D \sqsubseteq \bot \).

General role axioms are:

• Role inclusion axioms: \( P \sqsubseteq R \)

• Role equivalent axioms: \( P \equiv R \)

• Role disjointedness axioms: \( P \cap R \sqsubseteq \bot \)

There could be other role axioms, such as transitive, reflexive, irreflexive, symmetric, asymmetric etc. Concepts and role axioms depend on the expressivity of the DL and DL constructors. Some DL could have more axioms than the ones introduced in this section whereas some can have less. However, the DL-constructors introduced in the TBox, ABox and Axioms section here are common in most of the DLs.

3.2.4 Interpretations of KBs

The interpretations of a \( \mathcal{KB} \) define the semantics of the \( \mathcal{KB} \). Interpretations \( \mathcal{I} \) consist of a non-empty set \( \Delta^\mathcal{I} \), also called the domain of interpretations and an interpretation function \( \bullet^\mathcal{I} \). An interpretation, \( (\Delta^\mathcal{I}, \bullet^\mathcal{I}) \), of a concept \( C \) is a set \( C^\mathcal{I} \subseteq \Delta^\mathcal{I} \) and every interpretation of role \( R \) is \( R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I} \). The mapping descriptions of an interpretation function in the domain of interpretation are as follows (Table 3.1):
<table>
<thead>
<tr>
<th>Constructor</th>
<th>Syntax</th>
<th>Semantics/Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top</td>
<td>⊤</td>
<td>$\Delta^I$</td>
</tr>
<tr>
<td>Bottom</td>
<td>⊥</td>
<td>$\Phi$</td>
</tr>
<tr>
<td>Atomic Concept</td>
<td>$A$</td>
<td>$A^I \subseteq \Delta^I$</td>
</tr>
<tr>
<td>Concept negation</td>
<td>$\neg C$</td>
<td>$C^I \setminus \Delta^I$</td>
</tr>
<tr>
<td>Concept conjunction</td>
<td>$C \cap D$</td>
<td>$C^I \cap D^I$</td>
</tr>
<tr>
<td>Concept disjunction</td>
<td>$C \cup D$</td>
<td>$C^I \cup D^I$</td>
</tr>
<tr>
<td>Universal quantification</td>
<td>$\forall R.C$</td>
<td>${x \mid \forall y.(x, y) \in R^I \rightarrow y \in C^I}$</td>
</tr>
<tr>
<td>Existential quantification</td>
<td>$\exists R.C$</td>
<td>${x \mid \exists y.(x, y) \in R^I \land y \in C^I}$</td>
</tr>
<tr>
<td>Concept assertion</td>
<td>$C(a)$</td>
<td>$a \in C^I$</td>
</tr>
<tr>
<td>Role assertion</td>
<td>$R(a,b)$</td>
<td>$(a,b) \in R^I$</td>
</tr>
</tbody>
</table>

Table 3:1DL Syntax and Semantics

Expressive DL which contains constructors like at most restriction, at least restriction, nominal and data types have their own syntax and semantics.

3.2.5 Reasoning in DL

The $KB$ of a DL stores the axioms and assertions. Axioms inter relate the concepts and roles whereas, assertions map the individuals with concepts and roles. DL also provides the reasoning service to discover implicit knowledge or to answer any query. Reasoning services find the truth value of the queries related to TBox (concepts and roles interrelation) and ABox (concept and role relation with individuals) and also ABox instance retrieval i.e. individuals satisfying concepts or role assertion. The basic reasoning service includes:
- **KB Satisfiability**: Checks whether a model \( \mathcal{I} \) exists for \( \mathcal{KB} (\mathcal{KB} \models \mathcal{I}) \). Axioms could be contradictory such as \( C \sqsubseteq D \), \( C \cap D \sqsubseteq \perp \) and in that case no model will satisfy the \( \mathcal{KB} \).

- **Concepts (or roles) Satisfiability**: Checks whether a concept or role satisfies a model of \( \mathcal{KB} (\mathcal{KB} \models C(\text{or } R)) \) i.e. \( C^\mathcal{I} (\text{or } R^\mathcal{I}) \neq \emptyset \)

- **Concepts (or roles) Subsumption**: Checks whether a concept (or role) is more general than others \( (\mathcal{KB} \models C \sqsubseteq D(\text{or } P \sqsubseteq R)) \) i.e. \( C^\mathcal{I} \subseteq D^\mathcal{I} (\text{or } P^\mathcal{I} \subseteq R^\mathcal{I}) \).

- **Instance checking and retrieval**: Checks and retrieves all individuals satisfying concepts or roles assertion. e.g. \( \mathcal{KB} \models a : C \) i.e. \( a^\mathcal{I} \in C^\mathcal{I} \).

Different algorithms are used for reasoning services in a \( \mathcal{KB} \) which depends on the expressivity of the DL. Less expressive DLs use the structural subsumption algorithm that compares the syntactic structure of concepts but is not complete for DLs using negation and disjunction constructors. The Tableaux algorithm (Schmidt-Schaub and Smolka, 1991) is used by more expressive DLs to solve the satisfiability and subsumption problems.

### 3.3 Ontology and DL:

Ontology provides the semantics to the resources. The term ontology has been taken from philosophy where it is used to describe the entities in the world and their interrelations. In knowledge management terminology, ontology is a representation of a shared conceptualisation of a specific domain (Gruber, 1995; Uschold and Gruninger, 1996). Ontology provides the common vocabulary for the definition of concepts, roles and their constraints in the domain of application. The semantic ability of ontology
initiates proper communication between distributed and heterogeneous systems (and possibly people also).

Ontologies can be expressed in DLs using **TBox** and **ABox**. The vocabulary of an ontology can be divided into atomic concepts and roles and complex concepts and roles. The interrelations of the vocabulary can be written in terms of axioms. The benefit of DL based ontologies is that they can use reasoning mechanisms to extract the information and this facilitates machine understanding of resources. One of the important DL-based ontologies is OWL (Bechhofer et al., 2004) which facilitates internet based knowledge transfer.

### 3.4 Web Ontology Language (OWL)

OWL is the W3C recommendation for expressing ontologies in the semantic web. OWL provides greater machine understanding of web resources by providing more expressive vocabularies. OWL is based on the $\mathcal{SBOIN}(D^+)$ DL (Baader and Nutt, 2003). OWL uses the RDF(S)/XML based syntax and its mapping in the DL of some of its syntax has been shown in the table 3.2. (For a detailed discussion of OWL and DL see Patel-Schneider et al., 2004).
<table>
<thead>
<tr>
<th>OWL Syntax</th>
<th>DL Syntax</th>
<th>Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class( ( A ))</td>
<td>( A )</td>
<td>( A^T \subseteq \Delta^T )</td>
</tr>
<tr>
<td>Class (owl:Thing)</td>
<td>( \top )</td>
<td>( \Delta^T )</td>
</tr>
<tr>
<td>Class (owl:Nothing)</td>
<td>( \bot )</td>
<td>( \Phi )</td>
</tr>
<tr>
<td>intersectionOf (( C,D ))</td>
<td>( C \cap D )</td>
<td>( C^T \cap D^T )</td>
</tr>
<tr>
<td>unionOf (( C,D ))</td>
<td>( C \cup D )</td>
<td>( C^T \cup D^T )</td>
</tr>
<tr>
<td>complementOf (( C ))</td>
<td>( \neg C )</td>
<td>( C^T \setminus \Delta^T )</td>
</tr>
</tbody>
</table>

Table 3:2 OWL Syntax and Semantics
CHAPTER 4: DL-BASED QUERY IN ECOS ONTOLOGY FOR PARTNER IDENTIFICATION IN VE

4.1 INTRODUCTION

A VE lifecycle starts with the identification of a market opportunity and the consequent step of identifying the potential partners for the VE. The ECOS ontology provides a platform for enterprises to publish their information on the web. However, formation of VEs requires various functionalities from information support including the ability to search for enterprises with required characteristics such as competency, capacity, resources etc. Very often searching an ECOS database, according to the VE requirements, involves extracting implicit information and knowledge. The process of extracting information from an ECOS ontology is achieved by querying the ontology using the RDF graphs (Khilwani et al., 2011). Although an RDF based query can extract simple information from an ECOS ontology, it will fail when the required information is not explicitly mentioned. Furthermore, ECOS ontology is web based ontology (OWL) built using a subset of DL, and its queries have technical limitation due to the following reasons (explained in section 4.3.1):

1. DL-based reasoners (Pellet, Fact, etc.) cannot provide information if it is not explicitly mentioned in the ontology.
2. OWL does not support customized data type roles (Pan, 2004).

The objective of this chapter is therefore to develop a DL-based ECOS-Query system capable of extracting implicit knowledge and information in an efficient way and this addresses the research objective 1 set in the chapter 2. The query system developed here will extract implicit information in accordance with VE requirements without
changing the ECOS-ontology. In order to establish the ECOS-Query system this chapter focuses on:

1. An overview of DL based queries, ECOS-Query and shortcomings of the traditional DL based query methods (section 4.2 and section 4.3).
2. A query rearrangement procedure for the process of query optimization (section 4.4).
3. Query optimization techniques for efficient query answering (section 4.5).
4. A query processing and implementation procedure (section 4.6).

### 4.2 Ontology Queries

Chapter 3 briefly described DL, ontology and their reasoning procedure. An ontology reasoning service provides the ability to answer queries about the ontology. Ontology queries can be divided in two categories (1) **TBox** queries and (2) **ABox** queries.

#### 4.2.1 TBox queries

**TBox** queries explore the interrelations between concepts and roles. These can be divided into checking and retrieval of interrelations.

#### 4.2.1.1 TBox interrelation checking

**TBox** interrelation checking queries explore the different interrelations between concepts and roles such as: ?EquivalentClass(C1,C2), ?SubClass(C1,C2), ?DisjointWith(C1,C2), ?ComplementOf(C1,C2), ?EquivalentProperty(P1,P2), ?SubProperty(P1,P2), ?InverseOf (P1,P2), ?ObjectProperty (P), ?DataTypeProperty (P). The answers are Boolean i.e. True if the query assertion satisfies the **TBox** and False otherwise.
4.2.1.2 TBox interrelation retrieval

**TBox** interrelation retrieval queries find all related concepts and roles such as:

\( ?x : \text{EquivalentClass}(C,?x) \) provides all concepts who are equivalent to concept \( C \).

Similarly for \( ?x : \text{SubClass}(C,?x) \), \( ?x : \text{DisjointWith}(C,?x) \), \( ?x : \text{ComplementOf}(C,?x) \), \( ?x : \text{EquivalentProperty}(P,?x) \), \( ?x : \text{SubProperty}(P,?x) \), \( ?x : \text{InverseOf}(P,?x) \). Queries \( ?x : \text{ObjectProperty}(P,?x) \) and \( ?x : \text{DataTypeProperty}(P,?x) \) provide answers that are the object property or data property values of role \( P \).

4.2.2 ABox queries

**ABox** queries involve instance checking and instance retrieval of classes and roles.

4.2.2.1 ABox instance checking

The query: \( ? : C(a) \) provides the Boolean answer of whether there is an association between individual \( a \) with concept \( C \). The instance checking proceeds with adding \( \neg C(a) \) in the **ABox** and then checking the consistency/inconsistency of the **ABox**. If **ABox** is consistent with \( \neg C(a) \) then the answer of \( ? : C(a) \) will be false and true otherwise. Similarly, the query: \( ? : R(a,b) \) finds the Boolean answer for connectivity of instances \( a \) and \( b \) through role \( R \).

4.2.2.2 ABox instance retrieval

The query: \( ?x : C(?x) \) tries to find all individuals who are members of the concept \( C \). The instance retrieval queries bind the \( I \) for \( x \) (where \( I \in \text{individuals in A-Box} \)). The rest of the steps are similar to instance checking with \( \neg C(I) \). If the model is inconsistent then \( I \) is added in the answer set. In a similar manner the queries: \( ?x : R(?x,b), ?y : R(a,?y), ?x, ?y : R(?x,?y) \) find the respective values of individual or the data value associated with the respective role.
4.3 ECOS QUERY

The ECOS-query seeks the enterprises that can fulfil the VE requirements. Such queries can be written as:

\[ Q(x_1, x_2, \ldots, x_n) = C_1(x_1) \land \ldots \land C_i(x_i) \land \phi_1(x_i, y_1^i) \land \ldots \land \phi_j(x_j, y_j^j) \land \ldots \land \phi_n(x_n, y_n^j) \]

Where, \( x_i \)'s are the query variables which need to be mapped with the enterprise name. \( C \)'s are the concept formulas that can denote atomic concepts, union, intersection or negation of the concepts. Similarly, \( \phi \)'s define the characteristics of each \( x_i \) needed for the formation of a VE. \( \phi \)'s can be simple roles, union, intersection or negation of roles and data type properties. The required values \( y_j^i \) can be data type requirements (string or number), individuals or even one of \( x_i \).

4.3.1 Limitation of Queries

The ECOS ontology, which is OWL based built using a DL subset, contains all the information about enterprises i.e. type of enterprise, area of expertise, capacity or resources, location or distance etc. However some queries require answers that are not explicitly defined in the ontology structure. DL-based reasoners (Pellet, Fact) cannot answer queries which cannot be derived logically using ABox and TBox reasoning (see example 1 and 2 below) i.e. explicitly. Another shortcoming of OWL based ontologies is the inability to provide reasoning with customized datatype roles (see example 3 below). In such cases a DL-based query will not be able to provide the required answer for an ECOS-query. Examples of this include:

1. If a concept \( C \) is defined in the ontology, queries looking for the individuals who are not a member of concept \( C \) i.e. \(?x_i : \neg C(x_i)\) will return a null set or improper set unless the complement of concept \( C \) has also been explicitly defined in the ontology. The reason behind this is that
the ontology model will fail to find consistency\inconsistency with defined relations as not all consistent\inconsistent models can be inferred due to the lack of defined relations (¬\(C\) in this case) and hence the query result, in this case, will be improper.

2. The complement of role queries \(?x_i, y_i^j : ¬R(?x_i, y_i^j)\) will have the same problem.

3. Most of the data properties defined in the ontology will be represented by numerical or string values for example a laptop (LP1) with RAM 1 GB : 'RAM (LP1, 1GB)'. ECOS-Query might need to identify some of the numerical characteristics to be greater than, less than or even equal to (also applicable to string values) certain values which are not explicitly defined for example a query looking for a laptop with RAM greater than 512 MB : \(?x: RAM(?x, >512MB)\). DL-based reasoners will not be able to process this query and will not give LP1 as one of the answers as they cannot filter out such requirements (Pan, 2004).

4. Another important factor exists regarding the evaluation of optimized queries related to the order in which the queries are processed (Sirin and Parsia (2006), Ruckhaus (2004)). For example consider the query \(?x_i : C(?x_i) \land R(?x_i, y_i^j)\). Suppose \(N\) is the total number of individuals in which \(n_1\) belongs to concept \(C\) and \(n_2\) satisfies the role filler for \(x_i\) in \(R\) i.e. the answer of \(y_i^j\). If the sequence is: \(?x_i : R(?x_i, y_i^j) \land C(?x_i)\) then the number of evaluations would be \(N^2 + n_2\) but if the sequence is: \(?x_i : C(?x_i) \land R(?x_i, y_i^j)\) then the number of evaluations would be \(N + n_1 \times N\). It is very hard to determine the sequence in which the number of evaluations would be less as \(n_1\) and \(n_2\) are unknowns and can take any
value between 0 to N and there is no easy way to determine them. Different cost factors can be added to find the minimum number of evaluations (Motik et. al., 2003) however they are ontology dependent and can unnecessarily increase the complexity of the query. In order to simplify this process the ECOS query follows the class query, data type query and role query sequence.

The ECOS-Query system needs to extract the implicit information and knowledge from the ECOS ontology according to the VE requirement. In this process the first step is to identify the part of the query which requires implicit information to be extracted. The ECOS-Query system must therefore first rearrange the query according to the variables \((x_i, y_i)\) to identify the explicit and implicit parts of the query. The next step involves optimizing the query i.e. deleting the redundant parts of the query which do not affect the overall evaluation. In the final step query processing takes place. The ECOS-Query system can therefore be divided into three steps (as shown in the figure 4.1):

1. Rearrangement of the query.
2. Optimization of the query.
3. Processing the query with implicit and explicit information/knowledge retrieval.
4.4 QUERY REARRANGEMENT

An ECOS query is arranged according to the variables i.e. $?x_i$ etc. The query is categorized in two parts: 1. explicit 2. implicit for concepts and roles. The explicit part of the query is defined in the TBox and the negation of the implicit is defined in the TBox.

In case of concept satisfaction queries, an ECOS-Query may need to find the instances which are not related to a concept and there will be problems in achieving this if the negation of the concept is not defined in the ECOS ontology TBox. This can be shown by considering the concept level ECOS-Query which has been divided into four categories:

I. **ANDConcept** ($x_i$): This is the conjunctive query of instances with respect to the variable. For example $C_1(?x_i) \land \ldots \land C_n(?x_i)$.

II. **ORConcept** ($x_i$): This is the disjunctive query of instances with respect to a variable. For example $C_1(?x_i) \lor \ldots \lor C_n(?x_i)$. 

Figure 4:1 ECOS-Query system
III. **NOT-AND Concept** ($x_i$): This is the conjunctive query of negation of the instances of concepts where a concept is defined in the ontology but its negation is not. For example, $\neg C_i(?x_i) \land \ldots \ldots \land \neg C_n(?x_i)$, where $C_i$ is defined in ontology but $\neg C_i$ has not been included.

IV. **NOT-OR Concept** ($x_i$): This is the disjunctive query of negation of the instances of concepts where a concept is defined in the ontology but its negation is not. For example, $\neg C_i(?x_i) \lor \ldots \ldots \lor \neg C_n(?x_i)$, where $C_i$ is defined in the ontology but $\neg C_i$ has not been included.

The first two categories are the part of explicit concept queries whereas, the last two are the part of implicit concept queries.

For data-type role values three different categories are formed 1. Data-Greater, 2. Data-Less and 3. Data-Equal as follows:

I. **Data-Greater** ($x_i$): Where the value required from the query needs to be greater than the specified value. For example, $V < R_D(?x_i, ?d)$ is the query about all instances whose data value (?d) in data-type role ($R_D$) is greater than $V$, where $V$ is the numerical value.

II. **Data-Less** ($x_i$): Where the value required from the query needs to be less than the specified value. For example, $V > R_D(?x_i, d)$ is the query about all instances whose data value (?d) in data-type role ($R_D$) is less than $V$, where $V$ is the numerical value.

III. **Data-Equal** ($x_i$): Where the value required from the query needs to be equal to the specified value. Unlike Data-Greater and Data-Less queries, the specified value in Data-Equal categories can be both numerical and string. For example, $V \equiv R_D(?x_i, d)$ is the query about all instances whose data value (?d) in data-type role ($R_D$) is equal $V$, where $V$ can be both
numerical and string values. For example $V$ could be a capacity equal to 1000 units or a location equal to London.

Object properties (other than data-properties) are arranged in a pair of (2-tuple) variables and categorized as: 1. **ANDRole**, 2. **ORRole**, 3. **AND-NOTRole** and 4. **OR-NOTRole** as follows:

I. **ANDRole** ($x_i, y_i'$): This is the conjunctive query of roles specifying variables $x$ and $y$. For example $R_i(\? x_i, \? y_i') \land \ldots \land R_n(x_i, y_i')$.

II. **ORRole** ($x_i, y_i'$): This is the disjunctive query of roles specifying variables $x$ and $y$. For example $R_i(\? x_i, \? y_i') \lor \ldots \lor R_n(x_i, y_i')$.

III. **NOT-ANDRole** ($x_i, y_i'$): This is the conjunctive query of negation of roles specifying variables $x$ and $y$, where roles are part of the ontology but the negation of their roles are not, e.g. $\neg R_i(\? x_i, \? y_i') \land \ldots \land \neg R_n(x_i, y_i')$ where $R_i$ is defined in the ontology but not $\neg R_i$.

IV. **NOT-ORRole** ($x_i, y_i'$): This is the disjunctive query of negation of roles specifying variables $x$ and $y$, where roles are part of the ontology but the negation of their roles are not, e.g. $\neg R_i(\? x_i, \? y_i') \lor \ldots \lor \neg R_n(x_i, y_i')$ where $R_i$ is defined in ontology but not $\neg R_i$.

The query rearrangement step arranges the query according to the required variables. The next step involves query optimization which finds the interrelationship between a set of (part of) the query and deletes the redundant part of the query. Here the redundant part means that the deletion of that part of the query which does not affect the overall answer.
4.5 Query Optimization

Rearranged queries can be simplified using concepts and roles interrelationships i.e. sub, super or equivalent relations. Considering this fact, in many cases, the redundant part of the query can be safely removed without affecting the overall result. For example, if \( C_1 \subseteq C_2 \) then the query \( C_1(x_i) \lor C_2(x_i) \) w.r.t. variable \( x_i \) can easily be proved to be logically equivalent to the query \( C_2(x_i) \). Such simplification checks the TBox query about relational coupling between concepts and avoids unnecessary instance checking (\( C_1(x_i) \) in the above case). In the ECOS-query simplification has been carried out at three levels: 1. Concepts level, 2. Role level and 3. Class and Role level. The detailed description and redundant query removal process is described next.

4.5.1 Concepts vs. Concepts

As described before, class-instance queries have been classified as ANDConcepts, ORConcepts, NOT-ANDConcepts and NOT-ORClass. The redundancy checks at all sublevels are carried out as follows:

I. **ANDConcepts**: An ANDConcepts query with respect to variable \( x \) is \( C_1(x_i) \land C_2(x_i) \land \ldots \land C_j(x_i) \ldots \land C_n(x_i) \). If \( C_i \subseteq C_j \) then \( C_j \) can be removed from the query and the optimized query will be: \( C_1(x_i) \land C_2(x_i) \land \ldots \land C_j(x_i) \ldots \land C_n(x_i) \).

II. **ORConcepts**: An ORConcepts query with respect to variable \( x \) is: \( C_1(x_i) \lor C_2(x_i) \lor \ldots \lor C_j(x_i) \lor \ldots \lor C_n(x_i) \). If \( C_i \subseteq C_j \) then \( C_i \) can be removed to form the optimized query:

III. **NOT-ANDConcepts**: A query in this form will be: \( \neg C_1(x_i) \land \neg C_2(x_i) \land \ldots \land \neg C_j(x_i) \land \ldots \land \neg C_n(x_i) \) where concepts \( C_k \)'s are part of the ontology. If \( C_i \supseteq C_j \), then, \( \neg C_i \supseteq \neg C_j \) and
\( \neg C_i \) can be removed from the query. The optimized query will be:

\[ \neg C_1(x_i) \land \neg C_2(x_i) \land \ldots \land \neg C_j(x_i) \land \ldots \land \neg C_n(x_i). \]

**IV. NOT-OR Concepts:** A query in this form will be:

\[ \neg C_1(x_i) \lor \neg C_2(x_i) \lor \ldots \lor \neg C_i(x_i) \ldots \lor \neg C_j(x_i) \lor \ldots \lor \neg C_n(x_i) \]

where concepts \( C_k \)'s are part of the ontology. If \( C_i \subseteq C_j \) then, \( \neg C_i \supseteq \neg C_j \) and \( \neg C_j \) can be removed from the query to get:

\[ \neg C_1(x_i) \lor \neg C_2(x_i) \lor \ldots \lor \neg C_i(x_i) \lor \ldots \lor \neg C_n(x_i). \]

A similar line of argument can be added to accommodate equivalent, disjoint, complement and other types of concepts relation.

**4.5.2 Role vs. Role**

**ANDRole, ORRole, NOT-ANDRole** and **NOT-ORRole** queries for a pair of variables \( (x, y) \) can be simplified and optimized in a similar manner to that shown in the Concepts vs. Concepts section (Section 4.5.1) if relational attributes (sub role, super role, equivalent role etc.) are explicitly described in the ontology. If role relations are not described then domain and range of roles can be compared to find the relation as follows.

Let \( C^R_{R_i} \) and \( C^D_{R_i} \) be the range and domain concepts for role \( R_i \), Then

A. \( R_1 \equiv R_2 \) iff: \( C^D_{R_1} \equiv C^D_{R_2} \) and \( C^R_{R_1} \equiv C^R_{R_2} \)

B. \( R_1 \subseteq R_2 \) iff: (i) \( C^D_{R_1} \subseteq C^D_{R_2} \) and \( C^R_{R_1} \subseteq C^R_{R_2} \) or

(ii) \( C^D_{R_1} \subseteq C^D_{R_2} \) and \( C^R_{R_1} \equiv C^R_{R_2} \) or

(iii) \( C^D_{R_1} \equiv C^D_{R_2} \) and \( C^R_{R_1} \subseteq C^R_{R_2} \)

C. \( \neg R_1 \equiv \neg R_2 \) iff: \( C^D_{R_1} \equiv C^D_{R_2} \) and \( C^R_{R_1} \equiv C^R_{R_2} \)

58
D. \( \neg R_1 \sqsubseteq \neg R_2 \) iff: (i) \( C_{R_1}^D \sqsubseteq C_{R_2}^D \) and \( C_{R_1}^R \sqsupseteq C_{R_2}^R \) or

(ii) \( C_{R_1}^D \sqsupseteq C_{R_2}^D \) and \( C_{R_1}^R \equiv C_{R_2}^R \) or

(iii) \( C_{R_1}^D \equiv C_{R_2}^D \) and \( C_{R_1}^R \equiv C_{R_2}^R \)

4.5.3 Concepts vs. Roles

The process of **AND**, **OR** and **NOT** query optimization in Concepts vs. Role for a pair of variables is an extension of Role vs. Role optimization as explained next.

1. **ANDConcepts** vs. **ANDRole** query with respect to variables \((x_i, y_i^j)\) can be written as: \( C_i(x_i) \land R(x_i, y_i^j) \land C_2(y_i^j) \). If \( C_R^D \) and \( C_R^R \) are the domain and range of the role \( R \). Then

   (i) \( C_i(x_i) \land R(x_i, y_i^j) \land C_2(y_i^j) \equiv C_i(x_i) \land C_2(y_i^j) \) iff (any of the following conditions holds)

   a. \( C_1 \equiv C_R^D \) and \( C_2 \equiv C_R^R \)

   b. \( C_1 \sqsubseteq C_R^D \) and \( C_2 \equiv C_R^R \)

   c. \( C_1 \equiv C_R^D \) and \( C_2 \sqsupseteq C_R^R \)

   d. \( C_1 \sqsubseteq C_R^D \) and \( C_2 \sqsubseteq C_R^R \)

   (ii) \( C_i(x_i) \land R(x_i, y_i^j) \land C_2(y_i^j) \equiv R(x_i, y_i^j) \) iff (any of the following conditions holds)

   a. \( C_1 \sqsupseteq C_R^D \) and \( C_2 \equiv C_R^R \)

   b. \( C_1 \equiv C_R^D \) and \( C_2 \sqsupseteq C_R^R \)

   c. \( C_1 \equiv C_R^D \) and \( C_2 \sqsubseteq C_R^R \)

   (iii) \( C_i(x_i) \land R(x_i, y_i^j) \land C_2(y_i^j) \equiv C_i(x_i) \land R(x_i, y_i^j) \) iff (either of the following conditions holds)

   a. \( C_2 \equiv C_R^R \)

   b. \( C_2 \sqsupseteq C_R^R \)
(iv) \( C_1(x_i) \wedge R(x_i, y'_i) \wedge C_2(y'_i) \equiv C_2(y'_i) \wedge R(x_i, y'_i) \) iff (either of the following holds)
  a. \( C_1 \equiv C^D_R \)
  b. \( C_1 \equiv C^R_R \)

2. **ORConcepts** vs. **ORRole** query with respect to variables \((x_i, y'_i)\) can be written as: \( C_1(x_i) \lor R(x_i, y'_i) \lor C_2(y'_i) \). If \( C^D_R \) and \( C^R_R \) be the domain and range of the role R. Then

(i) \( C_1(x_i) \lor R(x_i, y'_i) \lor C_2(y'_i) \equiv C_1(x_i) \lor C_2(y'_i) \) iff (any of the following conditions hold)
  a. \( C_1 \equiv C^D_R \) and \( C_2 \equiv C^R_R \)
  b. \( C_1 \supseteq C^D_R \) and \( C_2 \equiv C^R_R \)
  c. \( C_1 \equiv C^D_R \) and \( C_2 \supseteq C^R_R \)
  d. \( C_1 \subseteq C^D_R \) and \( C_2 \equiv C^R_R \)

(ii) \( C_1(x_i) \lor R(x_i, y'_i) \lor C_2(y'_i) \equiv R(x_i, y'_i) \) iff (any of the following conditions hold)
  d. \( C_1 \subseteq C^D_R \) and \( C_2 \equiv C^R_R \)
  e. \( C_1 \equiv C^D_R \) and \( C_2 \subseteq C^R_R \)
  f. \( C_1 \subseteq C^D_R \) and \( C_2 \subseteq C^R_R \)

(iii) \( C_1(x_i) \lor R(x_i, y'_i) \lor C_2(y'_i) \equiv C_1(x_i) \lor R(x_i, y'_i) \) iff (either of the following holds)
  c. \( C_2 \equiv C^R_R \)
  d. \( C_2 \subseteq C^R_R \)

(iv) \( C_1(x_i) \lor R(x_i, y'_i) \lor C_2(y'_i) \equiv C_2(y'_i) \lor R(x_i, y'_i) \) iff (either of the following holds)
c. \( C_1 \subseteq C_R^D \)

d. \( C_1 \equiv C_R^D \)

3. **NOT-ANDConcepts vs. NOT-ANDRole** query with respect to variables (\( x_i, y_{i'} \)) can be written as: \( \neg C_1(x_i) \land \neg R(x_i, y_{i'}) \land \neg C_2(y_{i'}) \). If \( C_R^D \) and \( C_R^R \) be the domain and range of the role \( R \) and concepts \( C_1 \) and \( C_2 \) are part of the ontology but not their negation. Then

(i) \( \neg C_1(x_i) \land \neg R(x_i, y_{i'}) \land \neg C_2(y_{i'}) \equiv \neg C_1(x_i) \land \neg C_2(y_{i'}) \) iff (any of the following conditions hold)

a. \( C_1 \equiv C_R^D \) and \( C_2 \equiv C_R^R \)

b. \( C_1 \supseteq C_R^D \) and \( C_2 \equiv C_R^R \)

c. \( C_1 \equiv C_R^D \) and \( C_2 \supseteq C_R^R \)

d. \( C_1 \supseteq C_R^D \) and \( C_2 \supseteq C_R^R \)

(ii) \( \neg C_1(x_i) \land \neg R(x_i, y_{i'}) \land \neg C_2(y_{i'}) \equiv \neg R(x_i, y_{i'}) \) iff (any of the following conditions hold)

a. \( C_1 \subseteq C_R^D \) and \( C_2 \equiv C_R^R \)

b. \( C_1 \equiv C_R^D \) and \( C_2 \subseteq C_R^R \)

c. \( C_1 \subseteq C_R^D \) and \( C_2 \subseteq C_R^R \)

(iii) \( \neg C_1(x_i) \land \neg R(x_i, y_{i'}) \land \neg C_2(y_{i'}) \equiv \neg C_1(x_i) \land \neg R(x_i, y_{i'}) \) iff (either of the following holds)

a. \( C_2 \equiv C_R^R \)

b. \( C_2 \subseteq C_R^R \)

(iv) \( \neg C_1(x_i) \land \neg R(x_i, y_{i'}) \land \neg C_2(y_{i'}) \equiv \neg C_2(y_{i'}) \land \neg R(x_i, y_{i'}) \) iff (either of the following holds)

a. \( C_1 \subseteq C_R^D \)
4. **NOT-ORConcepts** vs. **NOT-ORRole** query with respect to variables \((x_i, y_i')\) can be written as: \(\neg C_1(x_i) \lor \neg R(x_i, y_i') \lor \neg C_2(y_i')\). If \(C_D^R\) and \(C_R^R\) be the domain and range of the role \(R\) and concepts \(C_1\) and \(C_2\) are part of the ontology but not their negation. Then

(i) \(\neg C_1(x_i) \lor \neg R(x_i, y_i') \lor \neg C_2(y_i') \equiv \neg C_1(x_i) \lor \neg C_2(y_i')\) iff (any of the following conditions hold)

a. \(C_1 \equiv C_D^R\) and \(C_2 \equiv C_R^R\)

b. \(C_1 \subset C_D^R\) and \(C_2 \equiv C_R^R\)

c. \(C_1 \equiv C_D^R\) and \(C_2 \subset C_R^R\)

d. \(C_1 \subset C_D^R\) and \(C_2 \subset C_R^R\)

(ii) \(\neg C_1(x_i) \lor \neg R(x_i, y_i') \lor \neg C_2(y_i') \equiv \neg R(x_i, y_i')\) iff (any of the following conditions hold)

a. \(C_1 \supset C_D^R\) and \(C_2 \equiv C_R^R\)

b. \(C_1 \equiv C_D^R\) and \(C_2 \supset C_R^R\)

c. \(C_1 \supset C_D^R\) and \(C_2 \supset C_R^R\)

(iii) \(\neg C_1(x_i) \lor \neg R(x_i, y_i') \lor \neg C_2(y_i') \equiv \neg C_1(x_i) \lor \neg R(x_i, y_i')\) iff (either of the following holds)

a. \(C_2 \equiv C_R^R\)

b. \(C_2 \supset C_R^R\)

(iv) \(\neg C_1(x_i) \lor \neg R(x_i, y_i') \lor \neg C_2(y_i') \equiv \neg C_2(y_i') \lor \neg R(x_i, y_i')\) iff (either of the following holds)

a. \(C_1 \supset C_D^R\)

b. \(C_1 \equiv C_D^R\)
4.6 QUERY PROCESSING AND IMPLEMENTATION

The query processing starts by defining the empty sets $\Theta(\sigma)$, where $\sigma$ is variables $x_i, y_j$ etc. Clearly the set $\Theta(x_i)$ should contain the set of individuals which satisfies all the requirements imposed by the query for $x_i$. To obtain the concluding set $\Theta(x_i)$ the following steps are taken:

1. **ANDConcepts query**: Let $\Psi(x_i)$ be the set of concepts, which need to be satisfied by variable $x_i$ which is in the conjunctive form.

   **Initialize**: $\Theta(x_i) = I(C_i(x_i))$ where $I(C_i(x_i))$ is the set of individuals satisfying the concept $C_i$ and $C_i \in \psi(x_i)$ . For the rest i.e. $i \in [2, N]$ where $N = |\psi(x_i)|$ (cardinality).

   $\Theta(x_i) = \text{similar}(\Theta(x_i), I(C_i(x_i)))$, where the function $\text{similar}(,,)$ contains the similar individuals from both sets. After $N$ steps the set $\Theta(x_i)$ will contain all individuals satisfying $\Psi(x_i)$.

2. **ORConcepts query**: Let $\Lambda(x_i)$ be the set of concepts which need to be satisfied by variable $x_i$ which are in disjunctive form then

   $\Theta(x_i) = \text{all}(\Theta(x_i), I(C_i(x_i)))$, where $i \in [2, N]$ with $N = |\Lambda(x_i)|$ (cardinality) and the function $\text{all}(,,)$ contains all the individuals from both sets without duplication.

3. **NOTConcepts query**: If $-C_i(x_i)$ is in the NOTClass query in both conjunctive and disjunctive forms where $C_i$ is part of the ontology but its negation is not then the following steps are taken:

   $\Theta(x_i) = \text{remove}(\Theta(x_i), I(C_i(x_i)))$ where the function $\text{remove}$ will delete all the elements from the first set which are also in second set.
The next step tries to satisfy the data type role requirement of the queries. The data requirement in the query (greater, less or equal) cannot be directly inferred from the ECOS ontology. This is dealt with at an individual level and is described below.

4. **Data-Type**: Let $D_g(x,d)$ be the set of data type role requirements of the queries which require the data value to be greater than the requirement. A temporary set of individuals $\Gamma(x)$ is formed according to the following requirement:

$$\Gamma(x) = \{I_i(\Theta(x)) | R_i(\Theta(x)), D), D > d_i, R_i \in D_g(x)\}.$$ 

Clearly, the set $\Gamma(x)$ contains all the individuals which fulfil the **Data-Greater** query. Finally, $\Theta(x)_{ij} = Similar(\Theta(x), \Gamma(x))$.

The process of **Data-Less** and **Data-Equal** query evaluation is the same but the selection criteria are different.

For **Data less** it is:

$$\Gamma(x) = \{I_i(\Theta(x)) | R_i(\Theta(x)), D), D < d_i, R_i \in D_g(x)\}$$

$\Theta(x)_{ij} = Similar(\Theta(x), \Gamma(x))$

For **Data equal** it is:

$$\Gamma(x) = \{I_i(\Theta(x)) | R_i(\Theta(x)), D) = d_i, R_i \in D_g(x)\}$$

$\Theta(x)_{ij} = Similar(\Theta(x), \Gamma(x))$

The next step deals with the role type query which looks for a set of two individuals which are connected through a particular required role. Two different cases can arise at this time: 1. Variables seeking for role answers are also seeking for concept or data type role answer. For example variables $x$ and $y$ seeking for a set of individuals satisfying the role assertion $R(x_j,y_{j'})$ also have $C_i(?x_j)$ and $C_j(?y_{j'})$ type queries. 2.
One or both variables $x_i$ and $y_i'$ appear only in a role query $R(x_i, y_i')$ but not in a concept or data query.

For the first type of role query, concept and data query have already found the set of individuals satisfying different variables. The next step in the role query is to identify and or remove those individuals in the set which also satisfies the role queries.

5. Starting with **ANDRole** query set $AR(x_i, y_i')$ this method first identifies the temporary set of individuals such that

$$\{\Gamma(x_i), \Gamma(y_i')\} = \{(I_i(\Theta(x_i)), I_i(\Theta(y_i'))), R_i(\Theta(x_i)), R_i(\Theta(y_i'))), R_i \in AR(x_i, y_i')\}$$

$$\Theta(x_i) = \text{Similar}(\Theta(x_i), \Gamma(x_i)), \quad \Theta(y_i') = \text{Similar}(\Theta(y_i'), \Gamma(y_i'))$$

In the conjunctive queries of roles (**ANDRole**) the process needs to check that all the individuals also satisfy the role queries.

In the case of a disjunctive query (**ORRole**) the process needs to add all individuals who satisfy this **ORRole** query set $(OR(x_i, y_i'))$ as follows:

$$\{\Gamma(x_i), \Gamma(y_i')\} = \{(x_i, y_i') | R_i(x_i, y_i'), R_i \in OR(x_i, y_i')\}$$

$$\Theta(x_i) = \text{All}(\Theta(x_i), \Gamma(x_i)), \quad \Theta(y_i') = \text{All}(\Theta(y_i'), \Gamma(y_i'))$$

In the case of not type roles either conjunctive (**NOT-ANDRole**) or disjunctive (**NOT-ORRole**) queries the first part will be same in defining the temporary set in line with conjunctive and disjunctive type. The final step is to remove all those individuals which satisfy them. It is easy to realize that if the process removes one of the non-satisfying sets the other will automatically be satisfied. To explain this, let us assume:

$$\Theta(x_i) = \text{remove}(\Theta(x_i), \Gamma(x_i)), \quad \Theta(y_i') = \text{remove}(\Theta(y_i'), \Gamma(y_i'))$$

Then the two sets $\{\Theta(x_i), \Theta(y_i')\}$ and $\{\Theta(x_i), \Theta'(y_i')\}$ will satisfy the requirement and hence can have two different sets of solution. The set $\{\Theta(x_i), \Theta(y_i')\}$
is the intersection of the previous two sets and hence does not require separate consideration.

The above steps will provide the final answers for variables \( x \) and \( y \). In the second type of role query, the process starts by initializing the \( x_i \) with any role requirement and then follows the same procedure.

### 4.7 ECOS-QUERY IMPLEMENTATION

W3C suggested SPARQL as a RDF based query language for web based ontologies such as OWL. At present SPARQL-DL (Sirin and Parsia, 2007), a substantial subset of SPARQL, provides more expressive DL queries than existing DL-query languages (Kremen and Sirin, 2008). SPARQL-DL allows TBox and Abox queries for exploring concept, role interrelation and instance checking and instance retrieval. The ECOS-Query system uses the SPARQL-DL for querying ECOS ontology.

The implementation process starts with ECOS-query rearrangement. The query rearrangement arranges the ECOS-query according to the variables and their requirement (i.e. concepts, data or role fulfilment). The next step starts with query optimization, which exploits the concepts and role interrelation and deletes the unnecessary part of the query without influencing the overall result. Finally the ECOS-query processing step finds the set of individuals satisfying the query requirement. The overall step involved in this approach has been shown in the figure (4.2).

Utmost care must be taken in order to initialize the result set for each variable to avoid incomplete or null results. The method for ECOS-query has been coded in the JAVA API using SPARQL-DL (SPARQL-DL API) query library for JAVA. The SPARQL-DL interface does the limited **TBox** and **ABox** queries (section 4.3.1) and has
been expanded further to be suitable for the ECOS-query and this will be shown in the case study example in section 8.1.

Figure 4:2 ECOS-Query Steps
CHAPTER 5: RISK ANALYSIS AND GAME THEORETIC REVENUE SHARING MECHANISM FOR PARTNER SELECTION IN VE

5.1 INTRODUCTION

Partner selection is an important component in the formation of a VE. Selection of the right partners in the formation of a VE is a difficult task due to its dynamic environment, risk factors and decentralized control. The ECOS-Query system (Chapter 4) can identify the set of potential partners according to the VE requirements but traditional methods of partner selection are still insufficient to incorporate the effect of different risk factors, lack of trust, cultural differences, management style and ability to cooperate etc. This chapter addresses the research objectives 2 and 3 (as listed in section 2.4) and presents the partner selection problem in VEs with risk analysis. Section 5.2 discusses the different types of risks in the formation of VEs. The application of cooperative game theory provides a better control mechanism among the enterprises through revenue sharing contracts. Revenue sharing is based on the efforts provided by the enterprises in terms of value addition and risk sharing (Section 5.3). A non-linear solution methodology is discussed in section 5.4 and sensitivity analysis, to consider the effect of changes in risk factors on the VE, is presented in the section 5.5. The overall process has been shown in the figure 5.1.

5.2 RISK ANALYSIS IN PARTNER SELECTION

In literature, risk has been defined as “the variance of probability distribution of outcomes” (March and Shapira, 1987). The success of a VE and its achievement of objectives depend on the individual partners’ capabilities and their cooperative
relationships. Thus a VE possesses multi-dimensional risk which can negatively affect the desired outcomes of the VE. In an early study of risk factors, Baird and Thomas (1990) concluded that generally risk emerges from eight different perspectives. In the collaborative or network environment, Das and Teng (2001) divided the risk factors into performance and relational risks. In another study, Harland et al. (2003) proposed a supply network risk tool to identify, assess and manage risk to support the single partner decision making process concerning network evaluation. In a collaborative environment, according to Link (2001), risk is higher than the risk related to the same business run by a single company. It is apparent from the findings of researchers that, although, return of investment, opportunities and risk sharing abilities are higher in VEs, they still operate in higher risk environments than single enterprises and therefore care must be taken in the

Figure 5.1 Partner selection process

(1990) concluded that generally risk emerges from eight different perspectives. In the collaborative or network environment, Das and Teng (2001) divided the risk factors into performance and relational risks. In another study, Harland et al. (2003) proposed a supply network risk tool to identify, assess and manage risk to support the single partner decision making process concerning network evaluation. In a collaborative environment, according to Link (2001), risk is higher than the risk related to the same business run by a single company. It is apparent from the findings of researchers that, although, return of investment, opportunities and risk sharing abilities are higher in VEs, they still operate in higher risk environments than single enterprises and therefore care must be taken in the
formation (including partner selection) of VEs as this plays an important role in their success.

In the formation of a VE, the whole project is divided into subprojects and for each subproject a single enterprise is selected, therefore, in the formation of VE, risks need to be analysed from two perspectives: direct influence and indirect influence. The overall division of risks has been shown in the figure 5.2 and is explained in the rest of this section.

To clarify the proposed VE risk decomposition, the overall risk can be considered as a combination of performance and network risk. Network risk is related to the interdependency in which lack of trust, inaccurate information sharing etc. hinders the effective collaboration (Ojala and Hallikas, 2006). Performance risk can be further
subdivided into individual, collaborative and imposed collaborative performance. Individual risk is associated with quality and capacity constraints of individual enterprises. Network and individual performance risk affect the VE directly i.e. they are considered when awarding a sub-project to an enterprise. Collaborative performance and imposed collaborative performance risk affect the VE indirectly i.e. they measure the overall contribution and overall risky environment and are used in determining revenue sharing as explained in the next section.

Risk analysis and risk management are generally composed of risk identification, risk evaluation or quantification in terms of costs, losses etc. and risk mitigation strategies. Various different methods have been proposed in the literature for risk evaluation or quantification such as stochastic risk modelling based on the risk preferences in different scenarios (Lu et al., 2008), fuzzy based risk evaluation under uncertain information based on VE requirements such as due date, cost, quality, precedence tasks etc. (Huang et al., 2011), PSO based combinatorial optimization to find the extreme value of risks i.e. minimum and maximum, interpretive structural modelling of risks sources in VEs, which is described as network risk here, (Alawamleh and Popplewell, 2011), ANP-based risk assessment (Ergu et al., 2011). Enterprises can use any of the appropriate risk quantification methods suitable for their VE requirements to calculate the individual performance risk and network risk. Collaborative and imposed collaborative performance risks are calculated based on the individual performance risk value. This research assumes that the enterprises have quantified their risks values and the next section introduces the mathematical part of the risk evaluation in the partner selection process.
5.3 **Risk Evaluation in Partner Selection**

The partner selection problem in terms of mathematical notation can be described as follows: A project is divided into $N$ subprojects, each of which requires a different competency and can be undertaken by a single enterprise. Subprojects are indexed as $i$, $i \in \{1, 2, ..., N\} = \mathbb{N}$. Let $m_i$ is the set of enterprises bidding for the subproject $i$, $i \in \mathbb{N}$. The decision variables: $x_{ik} \in \{0, 1\}: i \in \mathbb{N}, k \in m_i$, determine the subproject and enterprise relation which imposes a constraint that a subproject cannot be awarded to more than one enterprise. These notations are used to define various risks in VE formation as follows:

**5.3.1 Individual Performance Risk**

Every enterprise, bidding for the subproject provides information in accordance with the requirement about its ability, capacity, technology, competency and other related information. This information is used in calculating the individual performance risk. As explained earlier, individual performance risk is the measure of an enterprise’s ability to finish the subproject, in terms of time, quantity, quality and other aspects. In most cases, with improper and incomplete information regarding requirement and enterprises capability, individual performance risk can be given by interval values as: $[Lr_{ik}, Ur_{ik}]$, where $L$ and $U$ correspond to the lower and upper limit of the individual performance risk associated with the $k^{th}$ enterprise and the $i^{th}$ subproject. Depending upon the optimism present in the partners this risk can vary between its lower and upper limit and can be given as:

$$r_{ik} = \alpha Lr_{ik} + (1 - \alpha)Ur_{ik}$$

With $\alpha \in [0,1]$ defines the degree of optimism. Hence, a more optimistic network will have a lower risk value consideration when compared to a pessimistic network. The
analysis of $\alpha$ also provides decision robustness, which will be explained in the section 5.4.

If $c_i$ is the cost of subproject $i$, then after considering the individual performance risk, which will increase the cost of operation, the estimated cost of the subproject will be:

$$E[c_i] = \sum_k (1 + r_i^k)x_i^k c_i$$

and the cost of operation for the $k^{th}$ enterprise in the $i^{th}$ subproject will be:

$$C_i^k = (1 + r_i^k)c_i x_i^k$$

A single cost assumption for each sub project ($c_i$ here) may not be feasible in all cases where the cost of operation is different for different enterprises, however relaxing this assumption will not affect the procedure adopted in this paper. Individual performance risk of all subprojects is a joint measure of the ability of the VE to be successful. An enterprise generally bids for any sub-project based on its ability to complete it, whilst indirectly it suffers from the performance level of other member enterprises. Analysis of collaborative risk addresses this problem. Collaborative risk is a measure of the degree of risk in a VE which does not affect the direct risk in the VE, rather it calculates the indirect contribution of the VE due to the risk in its environment and provides a datum for determining the distribution of revenue using a game theoretic paradigm. Risk analysis in VE formation measures the collaborative risk in two parts: 1. **Collaborative performance risk** and 2. **Imposed collaborative performance risk**.

5.3.2 Collaborative performance risk

Collaborative performance risk analyses the risk level for each enterprise due to the presence of the other member enterprises in the VE. Based on the collaborative
performance risk factor, an individual enterprise can decide whether or not to join the consortium. Collaborative performance risk ($R^k_i$) can be calculated as:

$$R^k_i = (1 - r^k_i) x^k_i \left[ 1 - \prod_{i \neq i} \prod_{k \neq k} (1 - r^k_{i'} x^k_{i'}) \right]$$

The first part of the formula calculates the probability that the $k^{th}$ enterprise will be successful for the $i^{th}$ project (if assigned) and second part calculates the probability that at least one of the other subprojects will fail. The collaborative performance risk will indirectly increase the cost of investment for each enterprise as the higher the risk in the environment, the higher will be the risk on return of investment.

5.3.3 **Imposed collaborative performance risk**

The collaborative performance risk is related to the effect of member enterprises on the individuals, however, a converse study is also very important. Imposed relational risk is defined here, as risk imposed by individual enterprises on the member enterprises and this can be given by the following expressions:

$$IR^k_i = \left\{ \prod_{i \neq i} \prod_{k \neq k} (1 - r^k_{i'} x^k_{i'}) \right\} r^k_i x^k_i$$

The expression is simply calculated by the probability that an individual enterprise will fail whilst the other members succeed. This is the measure of how risky the VE will become by including any enterprise i.e. this will indirectly increase the cost of investment of member enterprises.

5.3.4 **Network risk**

Till now, risk analysis was based on the individual performance risk of candidates and the consequences of their inclusion on the other enterprises (partners), while it seldom incorporates the network risk in the formation of VE. In a VE, members have to
transcend their traditional enterprise boundaries and build a common platform to share their ideas and concerns. Collaboration and cooperation among members of a VE plays a significant role since the overall performance of the VE depends on a good combination of information, interpersonal interaction, cohesiveness, member satisfaction, mutual understanding and trust (Msanjila and Afsarmanesh, 2008) and absence of these can create network risk. While understanding network risk, some researchers have argued that firms favour past coalitions when choosing team members (Kaihara and Fujii, 2008).

In a VE, which is a temporary and short term coalition, enterprises may need to coordinate with others without prior experience. In this chapter network analysis is studied on a bilateral level, and consideration is given to past relationships (if any).

The first three types of risk, discussed above were related to individual risk factors and their consequences on the other partners. However, in a VE, where partners need to be seamlessly interoperable and cooperative, lack of communication, social, technological or cultural factors may hinder the desired output. This type of risk has been categorized as network risk in a VE, and this can be defined as:

$$m_r^k = \frac{\sum_{i',i \neq i // k',k \neq k} (1 - \gamma^{kk'}) x_i^k x_{i'}^{k'}}{2(n-1)}$$

With $\gamma^{kk'} \in [0,1]$. The value of $\gamma^{kk'}$ determines the affinity of collaboration between two enterprises ($k$ and $k'$) in the VE. If the value is close to 1, there will be a high affinity of collaboration due to a low risk factor. The network risk calculator assumes that the affinity of collaboration between two enterprises does not depend on or is not influenced by the presence of any third enterprise. Moreover, the affinity of collaboration between two enterprises is related to two subprojects, the network risk is
therefore divided by two to distribute it between the two sub projects undertaken by the
two enterprises and is divided by \((n-1)\) to make the value in the range 0 to 1.

The network risk can be given as:

\[
NR = \sum_k \sum_i nr_i^k
\]

Network risk generally does not affect the individual enterprises’ operational
costs but rather increases the risk of revenue generation i.e. the project value as explained
in the next section.

5.4 Game Theoretic Aspects in Partner Selection

The characteristics of a VE, basic common goals and decentralized decision and
control system, are analogous to cooperative game strategy. Cooperative game strategy
assumes that players can obtain a larger global benefit from pooling their resources than
by acting separately. This characteristic is particularly important when enterprises are
complimentary, as in the case of a VE and the maximum profit can only be achieved by
proper cooperation. One of the important aspects of cooperative game theory is its
stability and allocation of shares accepted by all. In cooperative game theory, several
concepts have been introduced for approaching the stability issue. A necessary condition
for the stability of a coalition is that no set of players is able to increase its members’
profits by forming a different coalition. The set of payoff profiles that verifies this
property is known as the core of the TU-game (Gillies, 1959). It is the set of non-
dominated feasible payoff profiles (also called imputations) covering all the possible
coalitions. This chapter next introduces the general concept of cooperative game theory.
5.4.1 Cooperative Game

Cooperative games with transferable utility are often defined by a set of players and a characteristics function, which specifies the outcome that each coalition can achieve. Let $N$ be the finite set of players and $v(S)$ be the characteristics function which determines the money or utility that coalition $S$ can divide, where $S$ is the nonempty subset of $N$ and the cooperative game with transferable utility can be given as: $(N,v)$.

For a detailed description of cooperative games and their different aspects see Peleg and Sudholter (2003). The next section (section 5.3.2) maps the partner selection problem of a VE in the cooperative game theory domain.

5.4.2 Cooperative game theory and partner selection

This section introduces the following further notations: let $V = \text{overall project value}$, with individual sub-project value as $v^i$, it is trivial that $\sum v^i = V$. Let $\beta^k_i$ be the minimum acceptable profit for the $k^{th}$ enterprise if engaged in the $i^{th}$ subproject and $T^k_i$ the amount received by the $k^{th}$ enterprise for the $i^{th}$ sub-project if engaged. With the value of network risk, the overall project value can be given as: $V' = (1 - \text{NR})V$ and at an individual level this can be given as $v'^i = v^i (1 - nr^k_i)$. The network risk will decrease the effective project value.

As discussed in the previous sections, circumstances that involve partner selection can be analysed by means of game theory as: players are the corresponding enterprises, enterprises collaborate to form a VE to obtain a common objective which is similar to cooperative games, and each enterprise has its own benefit in joining the VE when compared to utility functions in cooperative games. Hence, game theory can be applied in partner selection based on the revenue sharing mechanism as it has been widely applied to model the decision-making in inter-firm relations (Cousins 2002)
If $x$ and $T$ are the two decision factors defining the formation of a VE and distribution of the economy. Then the overall problem can be defined as the cooperative game $G(x, T)$, with the following definition.

**Definition 1:** A partner selection game is *individually rational (IR)* iff, $T_i^k \geq (c_i^k + \beta_i^k) x_i^k$. This term defines that if an enterprise is engaged in any subproject the enterprise’s expected profit should be more than the minimum acceptance level.

**Definition 2:** A partner selection game is *collectively rational (CR)* iff, $\sum_i \sum_i T_i^k \geq V'$. This term defines that the allocation is more than the speculated value (after considering network risk). A collectively rational group is one which does not want to deviate from the group as the overall utility is more than the speculated value.

**Definition 3:** A partner selection game is *coalitionally rational (COR)* iff, $\sum_i \sum_i T_i^k \leq V$. This term defines that the allocation should be less than or equal to the overall value of the project. Coalitionally rational solutions are also feasible solutions.

**Definition 4:** Core: The core of a partner selection game is the set of feasible payoff and allocation profiles that satisfy definitions (1), (2) and (3). Namely, it is the set of vectors $(x, T)$ that satisfy IR, CR and COR.

Analysis of $G(x, T)$ mostly involves analysis of the core of the game. The core contains the set of all feasible solutions. As explained earlier the core needs to satisfy IR, CR and COR in which IR provides the lower bound of the solution whereas COR and CR deal with the feasibility and optimal group selection. In order to construct a feasible core the concept of marginal contribution is introduced which provides the upper bound of the feasible solutions as explained next.

**Definition 5:** The marginal contribution of any enterprise in a VE is the sum of its direct and indirect contributions and this can be given as:

$$mc_i^k = v''\left\{ (1 + r_i^k + R_i^k - IR_i^k) x_i^k \right\}$$
This can be explained as the contribution of an enterprise (if engaged), \( mc_i^k \), which is the sum of direct contribution through sub project \( v'^i \) after considering the network risk, individual performance risk \( r_i^k \) and relational risk \( R_i^k \) undertaken with negative impact of imposed relational risk \( IR_i^k \). Marginal contribution defines the maximum input provided by any enterprise and is a very important factor while calculating the profit distribution.

**Property 1:** In a VE partner selection game, with a non-empty core, the vector \((x, T)\) that lies in the core also satisfies the marginal contribution principle (MCP) with \( T_i^k \leq mc_i^k \).

**Proof:**

It is clear that MCP satisfies COR and also IR for a non-empty core otherwise all solutions will be infeasible and for CR, \( V' \leq \sum_i \sum_k mc_i^k \) must be true and the proof is as follows:

From the definition of MCP and from the definition of \( R_i^k \) and \( IR_i^k \), the equation becomes,

\[
mc_i^k = v'^i x_i^k \{1 + r_i^k + R_i^k - IR_i^k\}
\]

\[
mc_i^k = v'^i [1 + r_i^k x_i^k + \{(1-r_i^k)x_i^k(1-\prod_{i'=i}^{k' \neq k} (1-r_{i'}^{k'}x_{i'}^{k'}))\} - \{(\prod_{i'=i}^{k' \neq k} (1-r_{i'}^{k'}x_{i'}^{k'}))r_i^k x_i^k\}]
\]

Assuming, \( z_i^k = (1-\prod_{i'=i}^{k' \neq k} (1-r_{i'}^{k'}x_{i'}^{k'})) \), the equation becomes

\[
mc_i^k = v'^i x_i^k [1 + r_i^k + (1-r_i^k)z_i^k - (1-z_i^k)r_i^k]
\]

and after rearrangement, the equation becomes:

\[
mc_i^k = v'^i x_i^k (1 + z_i^k), \quad z_i^k \geq 0 \quad \text{this means} \quad mc_i^k \geq v'^i x_i^k.
\]
Now the next step is to show that: $V' \leq \sum_i \sum_k v_i^k x_i^k$. From the definition of $v_i^k$, the term can be expressed as:

$$\sum_i \sum_k v_i^k = \sum_i \sum_k v_i^k (1 - nr_i^k) = V - \sum_i \sum_k v_i^k nr_i^k$$

As, $\sum_i \sum_k v_i^k nr_i^k \leq (\sum_i \sum_k v_i^k)(\sum_i \sum_k nr_i^k)$, since all the terms are non-negative. This means

$$(V - \sum_i \sum_k v_i^k nr_i^k) \geq V - (\sum_i \sum_k v_i^k)(\sum_i \sum_k nr_i^k)$$

or, $\sum_i \sum_k v_i^k \geq V - V(\sum_i \sum_k nr_i^k)$

or, $\sum_i \sum_k v_i^k \geq V(1 - \sum_i \sum_k nr_i^k)$

$$\Rightarrow \sum_i \sum_k v_i^k \geq V' \quad QED.$$ 

Hence, a core allocation is such that in any feasible coalition, an enterprise cannot obtain a payoff greater than its marginal contribution. With the objective of partner selection based on a revenue sharing mechanism, marginal contribution defines the upper bound whilst IR provides the lower bound.

5.4.2 Mathematical representation

Mathematically the revenue sharing mechanism in partner selection for a VE needs to solve $x$ and $T$ where $x$ determines the enterprises for the formation of VE and $T$ determines the revenue shared among those enterprises. The overall process can be defined as:

**The Mathematical formulation:**

$$\text{Max } (T) \quad \text{----------------------------------------------- (1)}$$

with the following constraints:
The objective function (1) maximizes the payoff of the member enterprises. Constraint (2) and (3) restrict the solution under IR and hence, a core allocation is such that in any feasible coalition, an enterprise cannot obtain a payoff greater than its marginal contribution. Constraint (4) is the feasibility constraint which defines that the overall payoff cannot be more than the net project value. Constraint (5) is the allocation constraint, which assures that a subproject should be awarded to one and only one enterprise.

Using the equations (1) and (4) the dual of the problem (1)-(5) can be stated as

$$\text{Min} \{ V - \sum_i \sum_k T_i^k \} \quad \text{----------------------------- (6)}$$

Subject to constraints (2), (3) and (5).

The primal and dual of the problem are solved to find optimal candidates and the range of revenue sharing (i.e. defining the core of the game). Due to constraint (3), this problem is of binary polynomial type. However, the problem can be subdivided in two parts, the first one will decide the optimal allocation ($x$) and second one decides the optimal payoff ($T$). To get the partition or subdivision of the problem, the following theorem has been used.

**Theorem 1** - *The optimal candidate in the VE formation satisfies the following equations:*

$$x^* = \text{arg max}_x \{ V' - \sum_i \sum_k C_i^k \}$$

**Proof:**
It is obvious that each project will select the optimal member enterprises i.e. 
\[ \max_{mc_i} \{ v^i - E(c_i) \} \] as the problem is to maximize \( T \). Thus the optimal candidates will provide \( \sum_i \max_{mc_i} \{ v^i - E(c_i) \} \) i.e. \( \max \{ V' - \sum_i \sum_k C_{ik}^k \} \).

As \( V' \leq \sum_i \sum_k mc_i^k \), \( \max \{ V' - \sum_i \sum_k C_{ik}^k \} \) provides the greatest upper bound for \( \sum_i \sum_k (mc_i^k - C_{ik}^k) \) and thus enterprises get the maximum profit in this case. This proves that optimal candidates satisfy the equation \( x' = \arg \max \{ V' - \sum_i \sum_k C_{ik}^k \} \). Therefore, \( x^* \) gives the position for each enterprise, where they can maximize their profit. Theorem 1 provides useful insight for problem solving as it leads to corollary 1 as stated below.

**Corollary1:**

The optimal partners provide the minimum risk associated cost.

**Proof:**

From Theorem 1, the optimal VE satisfies the following condition:

\[
\max \{ V' - \sum_i \sum_k C_{ik}^k \} \\
= \max \{ V(1 - NR) - \sum_i \sum_k (1 + r_i^k + c_i^k) \} \\
= \max \{ V - \{ \sum_i \sum_k \{ (1 + r_i^k) c_i^k \} + V.NR \} \}
\]

As \( V \) is constant, the equation turns out to be:

\[
\min \{ \sum_i \sum_k (1 + r_i^k) c_i^k \} + V.NR \quad \text{Proved.}
\]

Using theorem 1 and corollary 1 the problem can be divided into two phases, the first phase will select the optimal candidate while the second phase will determine the payoff allocation, as described below.

**Phase 1:** Identifying the optimal candidate or project allocation problem.

Using the corollary 1 this problem can be defined as:

\[
x^* = \arg \min_x \{ \sum_i \sum_k \{ (1 + r_i^k) c_i^k x_i^k \} + V.NR \} , \text{along with constraint (5)}
\]
$x^*$ is the optimal solution for the project allocation problem. Once the VE is formed, the next phase determines the range of payoffs for which they need to negotiate and is described below.

**Phase 2:** Revenue sharing or payoff allocation problem can be defined as:

$$\min \sum_i \sum_k (mc_i^k - T_i^k)$$  \hspace{2cm} (1b)

With the following constraints:

$$T_i^k \geq (C_i^k + \beta_i^k)x_i^k$$  \hspace{2cm} (2b)

$$\sum_i \sum_k T_i^k \leq V$$  \hspace{2cm} (3b)

$$x_i^k \in x^*$$  \hspace{2cm} (4b)

From the solution of the first phase $mc_i^k$ can be calculated and three different cases can be possible for phase 2:

**Case 1:**

$$\exists i, k \mid mc_i^k < (C_i^k + \beta_i^k)x_i^k$$

In this case the marginal contribution is less than the minimum expected profit. The second phase solution will be infeasible. In this situation the first phase solution is again carried out after removing the infeasible candidate.

**Case 2:**

$$\forall i, k \ mid mc_i^k > (C_i^k + \beta_i^k)x_i^k \text{ and } \sum_i \sum_k mc_i^k \leq V$$
In this case without carrying out the phase 2 calculation the solution will be \( mc_i^k \) and the rest of the profit \( (V - \sum_i \sum_k mc_i^k) \) requires further negotiation between the partners.

Case 3:

\[ \forall i, k \quad mc_i^k > (C_i^k + \beta_i^k)x_i^k \quad \text{and} \quad \sum_i \sum_k mc_i^k > V \]

This case gives the unique optimal solution after solving the second phase problem. The next section describes the solution methodology adopted for solving both phase problems.

5.5 Solution Methodology

It is clear that the first phase is a binary quadratic programing problem. The general approach for these kinds of problems is based on the relaxation of the binary constraint i.e. letting \( x \in [0,1] \) (continuous variable) instead of \( x \in \{0,1\} \) (binary variable) and using the branch and bound techniques in a similar fashion to the case of integer programming. This process solves the series of problems with a different combination of binary values of some of the \( x \) depending upon the lower bound of the particular problem. However, due to the constraint (5) where only one \( x \) is allowed to take the value one and rest of them will be zero for each sub-project, the following characteristics of the relaxed problem (relaxing binary constraint of \( x \)) are analyzed and presented next.

**Theorem 2**: If \( x \) is the optimal solution of the relaxed problem for any subproject \( i \), if
\[ x_i^k > x_i^{k'} \quad \forall i', k' \in m_i, k' \neq k \, , \text{then the optimal candidate for } i^{th} \text{ sub-project is } k \text{ i.e. } x_i^k : x_i^k = 1 \text{ and } x_i^{k'} = 0, \forall k' \in m_i, k \neq k' \]

This theorem tells that:
1. If $x^k_i$ is the optimal solution for the $i^{th}$ subproject i.e. if $x^k_i$ is the solution vector of the relaxed problem for the $i^{th}$ sub-project such that $x^k_i > x^k_i$ for $\forall k, k' \in m_i, k' \neq k$ then the optimal feasible solution for $i^{th}$ subproject will be $x^k_i$ where $x^k_i = 1, x^k_i = 0, \forall k' \in m_i, k \neq k'$.

2. $x^k_i$ is the global optimal solution for $i^{th}$ sub-project w.r.t. other sub-projects i.e. let $x^k_i$ be the solution vector with $x^k_i = 0, x^k_i = 1, x^k_i = 0 \forall k' \in m_i, k' \neq k$ and furthermore if $x^{k'}_i$ and $x^{k''}_i$ are the solution vectors when some other project is awarded to any other candidate in the same process with $x^k_i$ and $x^{k'}_i$ as feasible solutions then:
   1. $f(x^k_i) < f(x^{k'}_i)$
   2. $f(x^{k'}_i) < f(x^{k''}_i)$

**Proof:**

The first part of the theorem uses the tightest lower bound to get the feasible solution for $i^{th}$ subproject and the second part iterates that the solution obtained in the first part is still the best while making a solution feasible w.r.t. some other sub-project.

Let $x$ be the optimal solution of the relaxed problem, then:

$$x = \arg \min_x \{ \sum_k \sum_j (1 + r^k_j)c^k_j x^k_j + V.NR \}$$

and let $x^k_i > x^k_i' \forall k' \in m_i, k' \neq k$. Without loss of generality the following equation can be written:

$$f_{\min}(x) = \sum_{k \in m_i} (1 + r^k_j) x^k_i + \sum_{j, j' \in m_j} \sum_{k \in m_i} (1 + r^k_j) x^k_j +$$

$$\sum_{k \in m_i} \sum_{i' \neq i} \sum_{k' \in m_i} (1 - r^{k' k'}) x^k_i x^k_i'$$

Since in first part, only the value of $x$ corresponding the $i^{th}$ sub project is changed (from real to binary to make the solution feasible) and the rest are kept the same, the above equation can be written as:
\[ f_{\min}(x) = \sum_k A_i^k x_i^k + \sum_k B_i^k x_i^k + C = \sum_k (A_i^k + B_i^k) x_i^k + C = \sum_k \Lambda_i^k x_i^k + C \]

Where, \[ 1 + r_i^k = A_i^k, \quad \sum_{k \in m_i} \sum_{i \neq i'} \sum_{k \in m_i'} (1 - \gamma^k x_{i}') x_i = B_i' \]

and \[ \sum_{i,j=1} \sum_{k \in m_j} (1 + r_i^k) x_j^k = C \]

Now, changing the solution from real to binary w.r.t. sub-project \( i \), the following are obtained:

\[ f(x_i^k) = \sum_{z \in m_i} (\Lambda_i^z x_i^z) + C + \Lambda_i^k (1 - x_i^k) - \sum_{k' \in m_i, k' \neq k} \Lambda_i^{k'} x_i^{k'} \]

\[ f(x_i^{k'}) = \sum_k (\Lambda_i^k x_i^k) + C + \Lambda_i^{k'} (1 - x_i^{k'}) - \sum_{k' \in m_i, k' \neq k} \Lambda_i^{k'} x_i^{k'} \]

\( f(x_i^k) \) is obtained by making \( x_i^k = 1 \) and all others equal to 0 and \( f(x_i^{k'}) \) is obtained by making \( x_i^{k'} = 1 \) and all others equal to 0. In order to get \( f(x_i^k) < f(x_i^{k'}) \) this proof needs to show that

\[ \sum_k (\Lambda_i^k x_i^k) + C + \Lambda_i^{k'} (1 - x_i^{k'}) - \sum_{k' \in m_i, k' \neq k} \Lambda_i^{k'} x_i^{k'} \leq \sum_k (\Lambda_i^k x_i^k) + C + \Lambda_i^{k'} (1 - x_i^{k'}) - \sum_{k' \in m_i, k' \neq k} \Lambda_i^{k'} x_i^{k'} \]

should be true for all \( k' \in m_i, k' \neq k \) and after rearrangement it should be \( \Lambda_i^k \leq \Lambda_i^{k'} \) for all \( k' \in m_i, k' \neq k \).

Now as \( f_{\min}(x) = \sum_k (\Lambda_i^k x_i^k) + C \), interchanging the \( x_i^k \) and \( x_i^{k'} \) and keeping everything else the same, the following can be inferred:

\[ \Lambda_i^{k'} x_i^{k'} + \sum_{k' \in m_i, k' \neq k} \Lambda_i^{k'} x_i^{k'} + C \leq \Lambda_i^k x_i^k + \sum_{k' \in m_i, k' \neq k} \Lambda_i^{k'} x_i^{k'} + C \]

\[ \Rightarrow \Lambda_i^k x_i^k + \Lambda_i^{k'} x_i^{k'} \leq \Lambda_i^{k'} x_i^{k'} + \Lambda_i^{k'} x_i^k \]

\[ \Rightarrow \Lambda_i^k (x_i^k - x_i^{k'}) \leq \Lambda_i^{k'} (x_i^k - x_i^{k'}) \]

by the assumption \( x_i^k > x_i^{k'} \) the proof of the first part is complete.

For the proof of the second part let the \( j^{th} \) sub-project be awarded to the \( l^{th} \) enterprise then: \( \Lambda_i^j \leq \Lambda_j^j, \forall l', l \neq l' \in m_j \). Now

\[ f(x^{j'}) = \sum_{z \in m_j} (\Lambda_j^z x_j^z) + C + \Lambda_j^j (1 - x_j^j) - \sum_{z \in m_j, z \neq l'} \Lambda_j^z x_j^z \]

\[ f(x^{j'}) = \sum_k (\Lambda_j^k x_j^k) + C + \Lambda_j^j (1 - x_j^j) - \sum_{z \in m_j, z \neq l'} \Lambda_j^z x_j^z \]

...
Now, if \( f(x^{*^k}) < f(x^{*^{k'}}) \)
then
\[
\Lambda_i^k + \sum_{z \in m_j} (\Lambda_j^z x^z_j) + C + \Lambda_j^l (1 - x^l_j) - \sum_{z \in m_j, z' \neq l'} \Lambda_j^z x^z_j' \leq \\
\Lambda_i^{k'} + \sum_{z \in m_j} (\Lambda_j^z x^z_j) + C + \Lambda_j^l (1 - x^l_j) - \sum_{z \in m_j, z' \neq l'} \Lambda_j^z x^z_j'.
\]

after arrangement it becomes:
\[
\Lambda_i^k + \Lambda_j^l \leq \Lambda_i^{k'} + \Lambda_j^{l'}
\]

which is true from the first part of the proof and concludes the proof of the second part.

Using this theorem the problem needs to solve the \( n \)-relaxed problem where \( n \) is the number of projects. At each stage the problem finds solutions which have values closest to one and awards that subproject to that enterprise. The reduced problems are then again solved. After getting the optimum value of \( x \), i.e. for the first part of the problem, the second part of the problem is linear in revenue distribution mechanism. If the second part of the problem is infeasible, the constraint that is not satisfied i.e. that enterprise is removed from the list and the optimum value of \( x \) is again calculated. The overall process has been shown in the figure (5.3).
Figure 5.3 Pseudocode
5.6 SENSITIVITY ANALYSIS

In section 5.2, the individual risk of an enterprise is defined as:
\[ r_i^k = \alpha L r_i^k + (1 - \alpha) U r_i^k \],
where \( L \) and \( U \) refer to the lower and upper limit of the risk value and \( \alpha \in [0,1] \) as the degree of optimism. The calculation of phase 1 and phase 2 problems was carried out with the particular value of \( \alpha \), however, this section analyses the range of \( \alpha \) for which the current enterprise formation is stable. In other words, sensitivity analysis is carried out in terms of the degree of optimism which determines the resilience of the formed VE.

Let \( r_i^k(\alpha) \), \( C_i^k[\alpha_i^k(\alpha)] \) and \( mc_i^k(\alpha) \) be defined as risk, cost and marginal contribution as a function of \( \alpha \). The value of \( mc_i^k(\alpha) \) and \( x \) are known after the phase 1 and phase 2 calculations. The current optimum solution will become infeasible if the marginal contribution of any enterprise becomes less than its expected profit. This can be written as:
\[ mc_i^k(\alpha) < C_i^k[\alpha_i^k(\alpha)] + \beta_i^k \]. The critical value of \( \alpha \) can be given as:
\[ \alpha_c = \arg \max_{\alpha} \{ mc_i^k(\alpha) < C_i^k[\alpha_i^k(\alpha)] + \beta_i^k \}, \forall x_i \in x | \alpha \in [0,1] \}

If \( \alpha_c \notin [0,1] \), then VE is stable for all range of risk, otherwise the VE is unstable for \([0,\alpha_c]\). This analysis enables the VE to understand how stable they are when their risk environment changes.
CHAPTER 6: DL BASED ONTOLOGY MAPPING FOR INTEROPERABILITY IN VE OPERATION PHASE

6.1 INTRODUCTION

A VE, which is a temporary network of enterprises, is created after the identification of business opportunities and member enterprises work together in the operation phase to achieve desired goals. In a VE enterprises generally perform their tasks independently yet they still rely on other member enterprises, due to the interdependency of different tasks, for data, information and knowledge. The essential requirement for the success of VEs in the operation phase is effective and efficient communication, collaboration and cooperation among the member enterprises. Thus interoperability becomes the most important part of the operation phase.

Recent developments in ICT have provided a platform for enterprises to transfer data, information and knowledge during the operation phase to achieve proper coordination. ICT based transfer of data, information and knowledge can be divided into two levels: technical level and semantic level. The technical level is related to the contents or representation while the semantic level is related to the meaning of contents. Current ICT tools provide interoperability at a technical level but functionality is still lacking at the semantic level.

VEs therefore need to address the semantic interoperability issue in the operation phase. Semantic interoperability is required to ensure that the intention and meaning of information transferred between the enterprises will be understood correctly. In order to achieve semantic interoperability enterprises can use an ontology for acquisition and transfer of data, information and knowledge. Section 2.3.3 described the different standards of ontologies developed for enterprises, but these are generally based
on individual requirements and may use different terminologies. As it is very difficult to
develop a single, universally accepted ontology defining the whole universal system,
semantic heterogeneity remains a key issue in achieving VE interoperability.

This chapter addresses the 4th objective set in chapter 2 and discusses the
semantic interoperability issue by using DL based ontology mapping (figure 6.1).
Enterprises’ ontologies are merged to form the global ontology and DL based logical
derivation is used for defining bridging axioms between the entities of different
ontologies. This chapter focuses on:

1. Reasons for semantic heterogeneity and types of semantic heterogeneity
   (Section 6.2).
2. The role of bridging axioms in ontology mapping (Section 6.3).
3. Logical derivation of bridging axioms at concepts and roles level (Section 6.4)
4. Implementation methods of the proposed ontology mapping procedure
   (Section 6.5)

![Figure 6:1 Ontology mapping for interoperability in operation phase.](image)
6.2 SEMANTIC HETEROGENEITY IN ONTOLOGY MAPPING

Different forms of semantic heterogeneity may arise during ontology mapping such as lexical heterogeneity (same concept defined by different terms or the same terms being used to define different concepts), structural heterogeneity (difference in degree of details or granularity) etc. Various heterogeneities have been reported in ontology mapping (Wang and Liu, 2009) and the reported types of mismatches are as follow:

1. **Synonymy conflicts**: Same concepts defined by different terms.
2. **Polysemy conflicts**: Different terms defined by the same term.
3. **Subclass conflicts**: Occur when the same class in different ontologies is divided into different subclass concepts (i.e. difference in the granularity).
4. **Class-Role conflicts**: Occur when a class in one ontology is described by a role or properties in another ontology.
5. **Class Coverage conflicts**: Occur when a class defines the same concept in two ontologies but one class covers a broader domain than the other.
6. **Role conflicts**: Occur when the same class in different ontologies is described by different properties (roles).
7. **Role Attribute conflicts**: Occur when a class and its role are the same in two ontologies but their value types (attributes) differ.

The mismatches in the ontology mapping are due to assigning simple correspondence between entities which creates an erroneous or inconsistent mapping (Dou and McDermott, 2006). Figure 6.2 shows an example in which two different ontologies have been mapped, assuming consistent ontologies, simple similarity measures between them will provide the following correspondence:

O1: Operation → O2: Operation …………………….. (i),
O1: Transportation → O2: Transportation ……………..(ii).
Using the sub-concept relationship in the ontology O1, the following can be established:

O1: Transportation → O1: Operation .................. (iii)

Using equations (i), (ii) and (iii), the following can be inferred:

O2: Transportation → O2: Operation, but this is incorrect as they are disjoint concepts. Hence, according to the mapping, Transportation can be inferred as a sub-concept of the Operation in O2, therefore a logical inconsistency has occurred and therefore the mapping becomes erroneous. Such inconsistencies in mapping not only exist at the concept level but also at the role level. Differences may also exist at the level of granularity, which can be demonstrated by considering further detail of a product specification in two ontologies. This for example might be given as: O1: hasBore(x,y) and O2: hasDiameter(x,y), which means product x has bore (or diameter) y. Translating or mapping the role from O1 to O2 gives hasBore(x,y) → hasDiameter(x,y). This mapping is consistent but referring back to O1 from O2, hasDiameter(x,y) → hasBore(x,y) may be inconsistent because anything having a diameter does not necessarily imply having a bore.
As shown in the above examples, the simple correspondence between the terms of two ontologies commonly causes inference problems. This can be explained in terms of DL as: assuming \( P, Q \) are two terms in two ontologies, with simple correspondence \( P \rightarrow Q \) then \( \{KB, P, P \rightarrow Q\} \models Q \) i.e. \( Q \) can be inferred, but \( \{KB, P, P \rightarrow Q\} \not\models \neg P \rightarrow \neg Q \), i.e. \( \{KB, P, P \rightarrow Q\} \not\models Q \rightarrow P \). In other words, the fact that the \( Q \) can be inferred from \( P \) does not automatically mean that \( P \) can be inferred from \( Q \) and trying to infer this causes the error in the ontology mapping as described earlier. Such heterogeneity in the ontology mapping is caused by considering the correspondence but not analysing their relationship, such as more general (\( \supseteq \)), less general (\( \subseteq \)), equivalence (\( = \)), disjoint (\( \perp \)), overlapping (\( \cap \)) or union of other entities (\( e_1 \cup e_2 \cup \ldots \)). Defining such relationships in the mapping can prevent incorrect inference, For example if the mapping finds a correspondence such as \( P \subseteq Q \), this gives the KB two axioms: \( P \rightarrow Q \)
and $-P \rightarrow -Q$, for example $\text{hasBore}(x,y) \subseteq \text{hasDiameter}(x,y)$ mapping with defining relationship will not infer $\text{hasDiameter}(x,y) \rightarrow \text{hasBore}(x,y)$ whereas considering only the correspondence between the terms will produce the wrong inference.

### 6.3 Bridging Axioms and Ontology Mapping

Ontology mapping (sometimes called translation) is a difficult task especially across domains as in the case of a VE. Every enterprise develops its own terminology and axioms relating the terminology. In this research, ontology mapping is obtained by first merging the ontologies together by taking the union of terms and axioms, whilst preserving their namespace. Secondly bridging axioms are built between the terms in the two ontologies in order to build a global ontology which is then ready to merge with further ontologies. The reason for forming a global ontology lies in the fact that only $n-1$ mappings are required for $n$ ontologies whereas, $\binom{n}{2}$ mappings are required in the case of one to one mappings. Furthermore, any change in an ontology is easier to incorporate using a global ontology than separate one to one mappings.

Ontology mapping through ontology merging and bridging axioms can be given by the following function: $f: \{O_1, O_2, \ldots\} \rightarrow \{\text{GO}, BR_{ij}\}$. Where $O_i$ denotes the merging ontologies, $\text{GO}$ denotes the global ontology and $BR_{ij}$ is the set of rules (Bridging rules or axioms) inter-relating the entities of $O_i$ and $O_j$. The bridging rule $BR_{ij}$ w.r.t. $O_i$ and $O_j$ is said to be consistent if the following equation holds:

$$\{KB_{GO}: O_i(e)\} \models_{BR_{ij}} O_j(e) \quad \text{and} \quad \{KB_{GO}: O_j(e)\} \models_{BR_{ij}} O_i(e),$$

where $KB_{GO} = \bigcup_i (T_i \cup A_i)$ is the union of TBoxes and ABoxes of the merging ontologies. The above equation ensures that the mapping should be consistent locally, i.e. mapping from $O_i$ to $O_j$ using $BR_{ij}$ should be
consistent w.r.t. \( O_j \) and vice versa. The next section describes the DL based logical derivation of bridging axioms.

### 6.4 Logical Derivation of Bridging Axioms in Ontology Mapping

The process of ontology mapping, proposed in this chapter, has been depicted in figure 6.2. Although, great effort has been put into achieving fully automatic ontology mapping, human intervention is still needed for the final verification. In this research, the developed mapping technique takes a step closer towards automation by reducing human mediation. Ontologies are generally built upon atomic roles and concepts which are self-defined terms and depend on the domain of the ontology. Complex concepts are built upon atomic ones using a DL-Signature (union, intersection, quantification etc.). The first step in the ontology mapping method proposed here requires the relationships between atomic concepts and roles of the two ontologies to be found. As shown in the figure 6.3, human input is required for defining the relationships between atomic concepts in the two ontologies (\textbf{TBox} input) or providing individuals and their roles as \textbf{ABox} input. This process can be assisted by using wordnet (Wordnet API) as explained in Section 6.5. The rest of the mapping is then carried out, automatically by first deriving the bridging axioms between the entities of different ontologies and then verifying and validating the mapping to form a valid global ontology using DL reasoners.

In general, as described in chapter 3, ontologies are described by concepts (unary relations) and their roles (binary relations between concepts) and therefore an ontology’s interpretation \( I \) consists of the non-empty set \( \Delta^I \), the domain of interpretation. Every interpretation of a concept \( C \) is a set \( C^I \subseteq \Delta^I \) and every interpretation of role \( R \) is \( R^I \subseteq \Delta^I \times \Delta^I \). Using the prefix \( i \) and \( j \) for respective ontologies, the derivations of
bridging axioms between entities (concepts and roles) determine their interrelation. The next part of this section describes the DL based logical derivation for ontology mapping.

6.4.1 Concept level

At the concept level the bridging axiom $BR_{ij}$ is the relation from $\Delta^I_i$ to $\Delta^I_j$ and is the subset of $\Delta^I_i \times \Delta^I_j$. The process of identifying relationships between the concepts of two ontologies can be obtained by identifying relationships between the atomic concepts of two ontologies. Atomic concepts (sometimes also called base concepts) are used to build the complex concepts (sometimes also called name concepts or defined concepts).
Therefore, concept level bridging axioms are divided into two levels: atomic concept level and complex concept level.

6.4.1.1 Atomic concept level bridging axioms

Atomic concepts are self-defined concepts in the domain of knowledge or interpretation, therefore, TBox reasoning for bridging axioms derivation is not possible for atomic concepts. It can only be achieved manually or by expert’s interpretation i.e. manual entry of atomic concept relationships as bridging axioms or rules. However, with ontologies in similar domains or data bases ABox reasoning can be used to derive atomic level bridging axioms as follows:

a. **Equivalence relation**: Atomic concepts $A$ and $B$ in two ontologies are said to be equivalent if the following holds:

$$\{GO, BR\} \models i : A \equiv j : B \text{ iff } \forall x, A(x) \leftrightarrow B(x)$$

and the bridging rule is $BR_{ij} \rightarrow i : A \equiv j : B$

b. **Subsumption-Supersumption relation**: Atomic concept $A$ and $B$ in two ontologies can be inferred as a Subsumption-Supersumption relation if the following holds:

$$\{GO, BR\} \models i : A \subseteq j : B \text{ iff } (i) \forall x, A(x) \rightarrow B(x) \text{ and } (ii) \exists y, B(y) \land \neg A(y)$$

and the bridging rule is $BR_{ij} \rightarrow i : A \subseteq j : B$

c. **Overlapping relation**: Two atomic concepts $A$ and $B$ are in an overlapping relationship if the following holds:

$$\{GO, BR\} \models (A \cap B) \text{ iff } \exists x, A(x) \land B(x)$$

and the bridging rule is $BR_{ij} \rightarrow i : A \cap j : B$

d. **Disjoint relation**: Two concepts are said to be in a disjoint relationship if the following holds:
\{GO, BR\} \vdash (A \perp B) \iff \forall x, A(x) \leftrightarrow \neg B(x)

and the bridging rule is \( BR_i \rightarrow i : A \equiv j : \neg B \)

6.4.1.2 Complex concept level bridging axioms

For complex concepts, which are built from atomic concepts and roles, bridging axioms can be determined by analysing the atomic concepts and roles relationships. To illustrate the method applied in this research, consider the complex concepts \( C_i \) and \( C_j \) which are defined as:

\[ C_i = A_i \cap P.A_2 \quad \text{and} \quad C_j = B_i \cap R.B_2 \]

where the concepts and roles in the definition are atomic ones. Now the bridging axioms can be determined as:

a. **Equivalence relation:** Concepts \( C_i \) and \( C_j \) as defined above, in two ontologies are said to be equivalent if the following holds:

\[ \{GO, BR\} \vdash i : C_i \equiv j : C_j \quad \iff \quad \{(A_i \equiv A_2) \land (B_i \equiv B_2) \land (P \equiv R)\} \]

b. **Subsumption-Suorsumption relation:** Concept \( C_i \) and \( C_j \) in two ontologies can be inferred to be in a Subsumption-Supersumption relationship if the following holds:

\[ \{GO, BR\} \vdash i : C_i \subseteq j : C_j \quad \iff \quad \{(i) \{(A_i \equiv A_2) \land (B_i \equiv B_2) \land (P \equiv R)\} \quad \text{or} \quad (ii) \{(A_i \equiv A_2) \land (B_i \subseteq B_2) \land (P \equiv R)\} \quad \text{or} \quad (iii) \{(A_i \equiv A_2) \land (B_i \equiv B_2) \land (P \equiv R)\} \quad \text{or} \quad (iv) \{(A_i \subseteq A_2) \land (B_i \equiv B_2) \land (P \equiv R)\} \quad \text{or} \quad (v) \{(A_i \equiv A_2) \land (B_i \equiv B_2) \land (P \equiv R)\} \quad \text{or} \quad (vi) \{(A_i \equiv A_2) \land (B_i \equiv B_2) \land (P \equiv R)\} \quad \text{or} \quad (vii) \{(A_i \equiv A_2) \land (B_i \subseteq B_2) \land (P \equiv R)\} \]

c. **Overlapping relation:** Two concepts \( C_i \) and \( C_j \) are in an overlapping relationship if the following holds:

\[ \{GO, BR\} \vdash (C_i \cap C_j) \quad \iff \quad \{(A_i \land B_j) \lor (A_2 \land B_j) \lor (P \land R)\} \]
**d. Disjoint relation:** Two concepts $C_i$ and $C_j$ are said to be in disjoint relation if the following holds:

$$\{GO, BR\} \models (C_i \perp C_j) \iff \{(A_1 \equiv \neg B_1) \lor (A_2 \equiv \neg B_2) \lor (P \equiv \neg R)\}$$

### 6.4.2 Role level

The Role level mapping between two ontologies can be given by the bridging axioms $BR_{ij}$, which are relations in the subset of $(\Delta_i \times \Delta_i) \times (\Delta_j \times \Delta_j)$. Roles are also divided into two levels, atomic role level and complex role level. Unlike atomic concepts, bridging axioms in the atomic roles can be deduced using the concepts attached with domain and ranges of roles. The process of atomic and complex roles bridging axioms using TBox and ABox reasoning is presented next.

#### 6.4.2.1 TBox reasoning for bridging axioms in roles

DL roles are defined by binary relations, showing the relationships between two concepts. Role $R(x,y)$ defines the relationship $R$ between entities $x$ and $y$. Concept $C_d$, where $x \in C_d$ is the domain concept while $C_r$, where $y \in C_r$ is the range concept for role $R$. Using the concept level relation, role level bridging axioms can be deduced as:

**a. Equivalence relation:** Roles $P$ and $R$ in two ontologies can be said to be equivalent if the following holds:

$$\{GO, BR\} \models P \equiv R \iff C^P_D \equiv C^R_D \text{ and } C^P_R \equiv C^R_R.$$  

The bridging rule in this case is: $BR_{ij} \rightarrow i : P \equiv j : R$

**b. Subsumption-Supersumption relation:** Roles $P$ and $R$ in two ontologies can be inferred to be in a Subsumption-Supersumption relation if the following holds:
The bridging axioms will be $BR_{ij} \rightarrow i : P \subseteq j : R$

c. **Overlapping relation:** Two roles $P$ and $R$ are in an overlapping relationship if the following holds:

$\{GO, BR\} \models (P \cap R)$ iff $\{(C^p_D \equiv C^R_D) \land (C^p_R \equiv C^R_R)\} \lor
\{(C^p_D \subseteq C^R_D) \land (C^p_R \equiv C^R_R)\} \lor
\{(C^p_R \subseteq C^R_R) \land (C^p_R \subseteq C^R_R)\}$

and the bridging axiom is: $BR_{ij} \rightarrow i : P \cap j : R$

d. **Disjoint relation:** Two roles are said to be in a disjoint relationship if the following holds:

$\{GO, BR\} \models (P \perp R)$ iff $\{(C^p_D \equiv \neg C^R_D) \lor (C^p_R \equiv \neg C^R_R)\}$

and the bridging axiom is: $BR_{ij} \rightarrow i : P \equiv j : \neg R$

### 6.4.2.2 ABox reasoning for bridging axioms in role

Using the instances of database ABox reasoning can be used to determine the bridging axioms between the roles of two ontologies. The process is as follows

a. **Equivalence relation:** Roles $P$ and $R$ are said to be equivalent if the following holds:

$\{GO, BR\} \models P \equiv R$ iff $\forall x, y \ P(x, y) \leftrightarrow R(x, y)$

The bridging axiom in this case is: $BR_{ij} \rightarrow i : P \equiv j : R$

b. **Subsumption-Supersumption relation:** Roles $P$ and $R$ can be inferred

Subsumption-Supersumption relationship if the following holds:

$\{GO, BR\} \models P \subseteq R$ iff (i)$\forall x, y, P(x, y) \rightarrow R(x, y)$ and (ii)$\exists x, y, R(x, y) \land \neg P(x, y)$

The bridging axioms will be $BR_{ij} \rightarrow i : P \subseteq j : R$
c. **Overlapping relation:** Two concepts A and B are in an overlapping relationship if the following holds:

\[
\{GO, BR\} \models (P \cap R) \text{ iff } \exists x, y \ P(x, y) \land R(x, y)
\]

and the bridging axiom is: \( BR_{ij} \rightarrow i : P \land j : R \)

d. **Disjoint relation:** Two concepts are said to be in a disjoint relation if the following holds:

\[
\{GO, BR\} \models (P \perp R) \text{ iff } \forall x, y \ P(x, y) \leftrightarrow \neg R(x, y)
\]

and the bridging axiom is: \( BR_{ij} \rightarrow i : P \equiv j : \neg R \)

In a similar manner other role relations such as transitive, inverse etc. can also be deduced in the form of bridging axioms.

### 6.4.3 Concept and role level

In ontology development, it is possible that a concept in one ontology is described as a role in another ontology (Ghidini and Serafini, 2006). Concept to role mapping between two ontologies is achieved by the bridging axiom \( BR_{ij} \), finding a relation from \( \Delta^{i} \) to \( \Delta^{i} \times \Delta^{j} \) a subset of \( \Delta^{i} \times \Delta^{j} \times \Delta^{j} \). In terms of TBox and ABox reasoning the deduction of bridging axioms can be achieved by the following process:

#### 6.4.3.1 TBox reasoning for bridging axioms in concepts vs. roles

Assuming concepts \( C_i \) and \( C_j \) in an ontology \( i \), if a notion of \( C_i \) (as concept) and \( R_j \) (as role) in ontologies \( i \) and \( j \) are interrelated then bridging axioms can be deduced as:

a. **Equivalence relation:** Concept \( C_i \) and Role \( R_j \) are said to be equivalent if the following holds:

\[
\{GO, BR\} \models C \equiv R \text{ iff } (C_i \equiv C^{R_j}_D) \land (C_r \equiv C^{R_j}_R)
\]

where \( C^{R_j}_D \) and \( C^{R_j}_R \) are domain range concepts of role \( R_j \) and the bridging axioms can be given as: \( BR_{ij} \rightarrow i : C_i \equiv j : R_j.C^{R_j}_R \)
b. **Subsumption-Supersumption relation:** Concept $C_j$ and Role $R_j$ can be inferred to be in a Subsumption-Supersumption relationship if the following holds:

\[
\{GO, BR\} \models C_i \subseteq R_j \text{ iff (i) } \{(C_i \subseteq C_D^R) \wedge (C_i \subseteq C_R^R)\} \vee \\
\text{(ii) } \{(C_i \subseteq C_D^R) \wedge (C_i \equiv C_R^R)\} \vee \\
\text{(iii) } \{(C_i \subseteq C_D^R) \wedge (C_i \subseteq C_R^R)\}
\]

The bridging axioms will be $BR_{ij} \rightarrow i : C_i \subseteq j : R_j C_R^R$

c. **Overlapping relation:** Concept $C_i$ and role $R_j$ are in an overlapping relationship if the following holds:

\[
\{GO, BR\} \models (C_i \cap R_j) \text{ iff } \{(C_i \cap C_D^R) \wedge (C_i \cap C_R^R)\}
\]

and the bridging axiom is: $BR_{ij} \rightarrow i : C_i \cap j : R_j C_R^R$

d. **Disjoint relation:** Concept $C_i$ and $R_j$ are said to be in a disjoint relationship if the following holds:

\[
\{GO, BR\} \models (C_i \perp R_j) \text{ iff } \{(C_i \equiv \neg C_D^R) \vee (C_i \equiv \neg C_R^R)\}
\]

and the bridging axiom is: $BR_{ij} \rightarrow i : C_i \equiv j : \neg R_j C_R^R$

### 6.4.3.2 ABox reasoning for bridging axiom in concepts vs. roles

Using the instances from database, Abox reasoning can be used to determine the bridging axioms between concepts and roles of two ontologies. The process is given below.

a. **Equivalence relation:** Concept $C_i$ and Role $R_j$ are said to be equivalent if the following holds:

\[
\{GO, BR\} \models A \equiv R_j \text{ iff } \forall x, y \ C_i(x) \wedge C_j(y) \leftrightarrow R_j(x, y)
\]

and the bridging axiom is: $BR_{ij} \rightarrow i : C_i \equiv j : R_j C_R^R$

103
b. **Subsumption-Supersumption relation:** Concept $C_i$ and $R_j$ can be inferred as in a Subsumption-Supersumption relationship if the following condition holds:

\[
\{GO, BR\} \models C_i \subseteq R_j \text{ iff } (i) \forall x, y, C_i(x) \land C_i(y) \rightarrow R_j(x, y) \text{ and }
(ii) \exists x, y, R_j(x, y) \land \neg(C_i(x) \land C_i(y))\}
\]

The bridging axiom will be $BR_{ij} \rightarrow i : C_i \subseteq j : R_j, C^R_{ij}$

c. **Overlapping relation:** Concepts $C_i$ and Role $R_j$ are in an overlapping relationship if the following holds:

\[
\{GO, BR\} \models (C_i \cap R_j) \text{ iff } \exists x, y, C_i(x) \land C_i(y) \land R_j(x, y)
\]

and the bridging axiom is: $BR_{ij} \rightarrow i : C_i \cap j : R_j, C^R_{ij}$

d. **Disjoint relation:** Concept $C_i$ and role $R_j$ are said to be in a disjoint relationship if the following holds:

\[
\{GO, BR\} \models (C_i \perp R_j) \text{ iff } \forall x, y, C_i(x) \land C_i(y) \leftrightarrow \neg R_j(x, y)
\]

and the bridging axiom is: $BR_{ij} \rightarrow i : C_i \equiv j : \neg R_j, C^R_{ij}$

### 6.4.4 Combinations of concepts and roles

So far, this mapping approach has considered the one to one mappings between concepts, roles and concept-role. However, it is highly likely that a concept or role in one ontology is equivalent to subclass – superclass of a combination of concepts and roles in another ontologies as different ontologies may use different levels of granularity for their definitions. Such relationships or bridging axioms can be determined by one to many (or inversely by many to one) mappings. These can be done between a concept in one ontology to its sub-concepts in another ontology or between a role in one ontology and its sub-roles in another ontology or a mixture of both. The process of deducing bridging axioms or relationships is as follows:
a. **Concepts vs. Concepts:** Suppose a concept $A$ in one ontology has many subsumption relations with concepts $B_1, B_2, B_3, \ldots, B_n$ in another ontology. The equivalence relations between them can be found by both **TBox** and **ABox** reasoning. The process of **TBox** reasoning is to analyse relations at an atomic level as mentioned in the concept level mapping section and this can be given by the following equation:

\[
(A \equiv A_1 \cup A_2 \cup \ldots \cup A_m)
\]

Let \(\{GO, BR_p\} \models (A \equiv B_i \cup \ldots \cup B_n)\) iff \(A_i \equiv B_1, \ldots, A_n \equiv B_n\)

Through **ABox** reasoning the equivalence relation can be established by the following equation:

\[
\{GO, BR\} \models (A \equiv B_i \cup B_2 \cup \ldots \cup B_n)\) iff \(\forall x, A(x) \leftrightarrow B_i(x) \lor B_2(x) \lor \ldots \lor B_n(x)\)

Similar lines of argument and equations (**TBox** and **ABox**) can be given for subsumption-supersumption and other relations as previously described.

b. **Roles vs. roles:** If a role $P_i$ in one ontology has many sub role properties $R^i_1, R^i_2, \ldots$ in another ontology. The equivalence relation between them can be deduced using **TBox** and **ABox** reasoning as follows:

**TBox reasoning:**

\[
\{GO, BR_{ij}\} \models P_i \equiv \bigcup_k R^i_k \text{iff } (\forall i)\{(C^p_D \equiv \bigcup_i C^i_D) \land (C^p_R \equiv \bigcup_i C^i_R)\}
\]

Where, $C^p_D$ and $C^p_R$ are the domain and range of role $R^i_k$, $C^i_D$ and $C^i_R$ are the range and domain of concepts of role $P_i$.
**ABox Reasoning:**

\[ \{GO, BR\} \models (P \equiv R_1 \cup R_2 \cup \ldots \cup R_m) \iff \forall x, y P(x, y) \leftrightarrow R_1(x, y) \lor R_2(x, y) \lor \ldots \lor R_m(x, y) \]

Similar lines of argument and equations (TBox and ABox) can be given for subsumption-supersumptions and other relations as previously described.

### 6.5 Implementation Method

Ontology mapping is needed when enterprises working in similar or overlapping domains wish to collaborate. The proposed approach therefore assumes that there is some similarity in the terminology used between the enterprises, and then, ontology mapping can provide interoperability for transferring information and knowledge by comparing the complex terms at an atomic level. However, if the ontologies used by the enterprises are distinct, i.e. from completely orthogonal domains, the atomic terms used for building them will be completely different and there will not be any relationships between them. In that case the proposed methodology will form a global ontology which includes the distinct concepts and roles of the two original ontologies. Hence the proposed approach will still work in this case, although it must be noted that orthogonal domain ontologies do not generally require mapping techniques to be joined.

In this mapping approach, concepts and roles are compared at the atomic level which will resolve the ontology mapping conflicts described in the introduction section. Atomic level comparison between the concepts will resolve the subclass and class-coverage conflicts as the atomic level relationships are validated using human mediation (TBox/ABox input). As for the class-role conflicts, role conflicts and role attribute conflicts, these can be resolved by comparison of the domain and range of roles at the atomic level (similar to class conflicts). Synonymy and polysemy conflicts, which occur
due to linguistic characteristics, can be resolved by first using wordnet (Wordnet API) and then by comparing them at the atomic level. The next section describes the implementation method.

The process of implementation or deducing bridging axioms between ontologies is summarized in the figure 6.4. The first step starts with identifying concepts, roles (or properties) in ontologies using an ontology API (e.g. Jena) and providing them with different namespaces. The second step identifies the lexical similarity using Wordnet and the final step uses the DL reasoning (\textbf{ABox} and \textbf{TBox}) to deduce the bridging axioms. In this process, a global ontology is formed by incorporating all the entities of all the ontologies for mapping and identifying all their possible relationships.

In addition to using wordnet, the process of finding lexical similarities (i.e. synonyms, hyponyms etc.) can be enhanced by providing user interfaces to help the user identify similar words for any concepts and roles. As mentioned in the previous section, \textbf{TBox} reasoning can be used for concepts, and therefore their relationships, can be deduced if the relationships between atomic concepts of different ontologies can be provided by experts or users. Alternatively, \textbf{ABox} reasoning can be applied if the ontologies provide the same \textbf{ABox} assertions.

In the final step, a DL reasoner is used to find the relationships between the entities (concepts and roles) and to establish the bridging rules between the ontologies. A detailed description of implementation has been presented in the section 8.4.
Figure 6:4 Overall mapping procedures
CHAPTER 7 : DL BASED ONTOLOGY MERGING FOR KNOWLEDGE ENHANCEMENT IN VE’S TERMINATION PHASE

7.1 INTRODUCTION

In the termination phase a VE disbands after achieving its goals in the operation phase and the member enterprises continue their individual business and/or participate in further VEs as they wish. Although, technically the termination phase does not contribute much directly towards the success of VE, the acquisition, extraction and integration of knowledge, information and experience gained in the VE by member enterprises can boost their competitiveness through knowledge enhancement.

The success of a VE depends on its competitive edge over other large organizations or other VEs. The competitive advantage of a VE is the integrated accumulation of competitiveness provided by member enterprises in their domain of expertise. Therefore, the success of a VE directly depends upon the advancement of its member enterprises. Knowledge is a major resource which effectively boosts the competitiveness of enterprises. Thus enterprises need to be knowledge intensive and adopt efficient KM tools not only for the success of their VE but also to remain competitive independently and be able to collaborate proficiently in the future VEs.

Nowadays, enterprises are using ontologies to represent their knowledge base (Section 2.3.4). Ontologies provide the structural decomposition of concepts and their relation (Chapter 3). As knowledge is more valuable if it can be made inferable and deducible, discovered knowledge will not be useful unless it is mapped semantically and structurally with the existing ontologies. It is not advisable to construct an ontology from the beginning simply to accommodate new knowledge, therefore ontology merging techniques are required to identify and accommodate new knowledge within existing
ontologies. The ontology merging technique should consider both syntax and semantics and be able to perform the following tasks:

1. Deduce similar or new concepts.
2. Deduce the position where concepts should be merged in the ontology by either reconfiguring or restructuring.
3. Achieve logically consistent mappings.

This chapter addresses the 5th objective set in chapter 2 and develops the DL based ontology merging technique to provide a platform for enterprises to accommodate new knowledge and to enhance their competitiveness (figure 7.1). This chapter focuses on:

1. Process of finding similarity between the concepts of two ontologies.
2. Process of finding the similarity index (matrix) between the concepts of two ontologies.
3. Finding the hierarchical position of the new concept in the existing ontology and adding it through reconfiguration and restructuring of existing ontology.
4. Logically verifying and validating the position of a new concept in the ontology.

Figure 7:1 Ontology merging process for accommodating new knowledge
7.2 Ontology Similarity

An ontology is the explicit specification of shared conceptualization (Gruber, 1995). In simple words, an ontology is a domain specific knowledge representation specified in terms of concepts and their relations. An ontology can be represented as \( O := \{ C, R, A \} \) where \( C \) is the set of concepts, \( R \) is the set of roles and \( A \) is the set of axioms. The process of ontology merging starts by defining the similarity function for similarity value calculation. For this, let an ontology \( O_1 \) be defined as the existing knowledge and \( O_2 \) as the new knowledge. Let \( C^i_j \) and \( C^j_2 \) be the \( i^{th} \) and \( j^{th} \) concepts of two ontologies \( O_1 \) and \( O_2 \) respectively such that \( C^i_j \in O_1 \) and \( C^j_2 \in O_2 \). All other notations used in this section are as follows:

\[
\begin{align*}
    SynC^i_j : & \quad \text{Synonym set of } C^i_j, \\
    HyperC^i_j : & \quad \text{Hypernym set of } C^i_j, \\
    HypoC^i_j : & \quad \text{Hyponym set of } C^i_j, \\
    SC^i_j : & \quad \text{Set of super concepts of } C^i_j, \\
    sC^i_j : & \quad \text{Set of sub concepts of } C^i_j,
\end{align*}
\]

\[
\begin{align*}
    SynC^j_2 : & \quad \text{Synonym set of } C^j_2, \\
    HyperC^j_2 : & \quad \text{Hypernym set of } C^j_2, \\
    HypoC^j_2 : & \quad \text{Hyponym set of } C^j_2, \\
    SC^j_2 : & \quad \text{Set of super concepts of } C^j_2, \\
    sC^j_2 : & \quad \text{Set of sub concepts of } C^j_2.
\end{align*}
\]

7.2.1 Similarity function

A similarity function calculates both semantic similarity and structural similarity. A semantic similarity function is related to the calculation of linguistic association and a structural similarity function calculates the hierarchical association in the ontology. The process is as follows:

7.2.1.1 Semantic similarity function

A semantic similarity function determines how closely two concept names are linguistically associated. In language two words can be related to each other in various
ways e.g. same root, antonyms etc. However, the synonyms, hypernyms and hyponyms of two words imitate the equivalent, super and sub relationship of ontological concepts, and therefore only synonym, hypernym and hyponym relations have been taken into account for calculating the semantic similarity. Wordnet API is used to find the synonyms, hypernyms and hyponyms set of a word. Wordnet (wordnet API), created by Princeton university, is a dictionary of semantically similar English words, arranged structurally. Words are characterized based on the parts of speech- noun, verb, adjective etc. and linked together and categorized as synonyms, hyponyms etc.

The semantic similarity function $\psi$ maps the linguistic relation between two words and provides the numerical value i.e. $\psi : \{C_i^1, C_j^2\} \rightarrow [0,1]$, where $C_i^1 \in O_1$ & $C_j^2 \in O_2$ are the name of concepts. The semantic similarity function $\psi$ is sub-divided in three parts: Synonym, Hypernym and Hyponym and their calculations are given below:

**Synonym:**

$\psi_1(C_i^1, C_j^2) = 1$, if $\exists t_1, t_2 | t_1 = t_2 \ & \ t_1 \in SynC_i^1 \ & \ t_2 \in SynC_j^2$

$\psi_1(C_i^1, C_j^2) = 0$, otherwise

**Hypernym:**

$\psi_2(C_i^1, C_j^2) = \beta_2$, if $\exists t_1, t_2 | t_1 = t_2 \ & \ t_1 \in HyperC_i^1 \ & \ t_2 \in HyperC_j^2$

$\psi_2(C_i^1, C_j^2) = 0$, otherwise

where $\beta_2 \in [0,1]$

**Hyponym:**

$\psi_3(C_i^1, C_j^2) = \beta_3$, if $\exists t_1, t_2 | t_1 = t_2 \ & \ t_1 \in HypoC_i^1 \ & \ t_2 \in HypoC_j^2$

$\psi_3(C_i^1, C_j^2) = 0$, otherwise

where $\beta_3 \in [0,1]$
Here, $\beta_2$ and $\beta_3$ are weights given to the Hypernym and Hyponym relations. The final semantic similarity function will be the maximum of all.

$$\psi(C_i^1, C_j^2) = \max\{\psi_1(C_i^1, C_j^2), \psi_2(C_i^1, C_j^2), \psi_3(C_i^1, C_j^2)\}$$

### 7.2.1.2 Structural Similarity function

The structural similarity between the concepts of two ontologies is the measurement of their association in terms of equivalence, super and sub relationships. The structural similarity is measured at three levels: equivalence ($ER$), super ($SupR$) and sub relation ($subR$) as explained next.

**EQUIVALENCE RELATION SIMILARITY FUNCTION (ER)**

In structural similarity, an equivalence relation between two concepts is closely associated with the equivalence between their super and sub concepts respectively. An equivalence relation similarity function calculates the similarity between the super and sub concepts respectively of the two concepts. Mathematically, the equivalence relation between concept $C_i^1$ and $C_j^2$ can be given as:

$$ER(C_j^2, C_i^1) = 0.5 \left\{ \frac{Sim(SC_i^1, SC_j^2)}{|SC_i^1 \cup SC_j^2|} \right\}^2 + 0.5 \left\{ \frac{Sim(sC_i^1, sC_j^2)}{|sC_i^1 \cup sC_j^2|} \right\}^2$$

Here, function $Sim(.,.)$ determines the number of similar elements in the two sets and $|A|$ is the cardinality of the set $A$. The first part of the equation calculates the similarity in terms of super concepts and the second part calculates the similarity in terms of sub concepts. Squaring the function gives more weightage to the structurally equivalent concepts as the ratio will never exceed the value 1. Equal weightage has been given to both the parts as the two concepts are equivalent if their super and sub concepts are equivalent respectively.
**SUPER RELATION SIMILARITY FUNCTION (SUPR)**

A new concept is said to be in a super-concept relationship with any of the existing concepts if sub-concepts of the new concept match with the super-concepts of the existing concepts. This can be explained as the new concept is a super-concept of the existing concept, if in the hierarchy it should be above the existing concept. If a concept $C_i^1$ is the super-concept of $C_j^2$ then $sC_i^1$ should match with $SC_j^2$. Mathematically the SupR function can be given as:

$$\text{SupR}(C_i^2, C_j^1) = \left\{ \text{Sim}(\{sC_i^2 \cup C_j^2\}, \{SC_i^1 \cup C_j^1\}) \right\}^2$$

The super relation function (SupR) includes both concepts ($C_i^1$ and $C_j^2$) and their super and sub concepts and measures the proximity of the two concepts in terms of the super relation.

**SUB RELATION SIMILARITY FUNCTION (SUBR)**

A new concept is said to be in a sub-concept relationship with any of the existing concepts if super-concepts of the new concept match with the sub-concepts of the existing concepts. This, in contrast to the super relation similarity function, explores the hierarchical structure in which the new concept should be below the existing concept. If a concept $C_i^1$ is the sub-concept of $C_j^2$ then $SC_i^1$ should match with $sC_j^2$. Mathematically the subR function can be given as:

$$\text{subR}(C_j^2, C_i^1) = \left\{ \text{Sim}(\{sC_i^1 \cup C_j^1\}, \{SC_j^2 \cup C_j^2\}) \right\}^2$$

Similar to the argument of SupR function, subR function includes both concepts ($C_i^1$ and $C_j^2$) in the numerator and finds the proximity of the two concepts in terms of a sub relation.
7.3 Ontology Similarity Index (Matrix)

This section further explores the similarity function and calculates the similarity matrix of a new concept with all of the existing concepts. The calculation of the similarity matrix has been divided in three parts: Equivalent, Super and Sub.

7.3.1 Equivalent Similarity Matrix (ERM)

Using the semantic and equivalence relation similarity function, the equivalence relation \((\mathcal{ER})\) between two concepts \(C_i\) and \(C_j\) can be given as:

\[
\mathcal{ER}(C_i, C_j) = k \psi(C_i, C_j) + (1 - k) \mathcal{ER}(C_i^2, C_j^2)
\]

where \(k \in [0,1]\) is the weight given to the semantic similarity function.

The equivalence relation \((\mathcal{ER})\) value correlates the two concepts in terms of the equivalence relation. Equivalence relation matrix \((\mathcal{ERM})\) depicts the \(\mathcal{ER}\) values of new concepts with all the concepts of the existing ontology. Assuming \(n\) is the total number of concepts in ontology 1, in mathematical terms \(\mathcal{ERM}\) \((C_j)\) can be given as:

\[
\mathcal{ERM}(C_j) = [\mathcal{ER}(C_j, C_1^1), \mathcal{ER}(C_j, C_2^1), \ldots, \mathcal{ER}(C_j, C_n^1)]
\]

7.3.2 Super Similarity Matrix (SRM)

Similar to the \(\mathcal{ERM}\), \(\mathcal{SRM}\) uses the semantic and super relation similarity function to calculate the super relation \((\mathcal{SR})\) between two concepts \(C_i\) and \(C_j\) and it can be given as:

\[
\mathcal{SR}(C_i, C_j) = k \psi(C_i, C_j) + (1 - k) \mathcal{SupR}(C_i^2, C_j^2)
\]

where \(k \in [0,1]\) is the weight given to the semantic similarity function.

The super relation \((\mathcal{SR})\) value correlates the two concepts in terms of the Super-Sub relationship. \(\mathcal{SRM}\) represents the \(\mathcal{SR}\) values of new concepts with all the concepts.
of the existing ontology. Assuming \( n \) is the total number of concepts in ontology 1, in mathematical terms \( \text{SRM}(C_j^2) \) can be given as:

\[
\text{SRM}(C_j^2) = \left[ \text{SR}(C_j^2, C_{i_1}^1), \text{SR}(C_j^2, C_{i_2}^1), \ldots, \text{SR}(C_j^2, C_{i_n}^1) \right]
\]

### 7.3.3 Sub Similarity Matrix (sRM)

\( \text{sRM} \) also uses the semantic and sub relation similarity function to calculate the sub relation (\( \text{sR} \)) value between the two concepts \( C_i^1 \) and \( C_j^2 \) and can be given as:

\[
\text{sR}(C_j^2, C_i^1) = k \psi(C_j^2, C_i^1) + (1 - k) R(C_j^2, C_i^1)
\]

where \( k \in [0,1] \) is the weight given to the semantic similarity function.

The sub relation (\( \text{sR} \)) value correlates the two concepts in terms of the Super-Sub relationship. \( \text{sRM} \) represents the \( \text{sR} \) values of new concepts with all the concepts of the existing ontology. Assuming \( n \) is the total number of concepts in ontology 1, in mathematical terms \( \text{sRM}(C_j^2) \) can be given as:

\[
\text{sRM}(C_j^2) = \left[ \text{sR}(C_j^2, C_{i_1}^1), \text{sR}(C_j^2, C_{i_2}^1), \ldots, \text{sR}(C_j^2, C_{i_n}^1) \right]
\]

The relational matrix (equivalence, super and sub) obtained here serves two purposes as it not only relates the closeness of two concepts from different ontologies but also explores the kind of relationship (equivalence, super and sub) which exists and is used for ontology merging and reconfiguration. The next section describes the process of ontology merging through reconfiguration and reconstruction.

### 7.4 Ontology Merging through Reconfiguration and Restructuring

The final step of this procedure merges the new concepts in the existing ontology through reconfiguration and restructuring with logical validation. The relational matrix
obtained in the previous section is used for ontology merging. The first part of this process determines the greatest similarity in terms of equivalence, super and sub relations between the new concept and existing ontology concepts.

**7.4.1 Similarity and type of similarity of concepts**

Assuming ontology $O_1$ is the existing ontology and ontology $O_2$ is the new ontology. If $C^2_j \in O_2$ is a new concept, the first stage is to find the existing concept with the greatest similarity in terms of equivalence, super or sub. Let $i^{th}$ concept of $O_1$, $C^1_i \in O_1$, be the existing concept with greatest similarity value. Mathematically, $i$ can be determined as:

$$i = \arg\max_{\alpha_i} \{ \text{ERM} \left( C^2_j \right), \text{SRM} \left( C^2_j \right), \text{sRM} \left( C^2_j \right) \}$$

$$= \arg\max_{\alpha_i} \left\{ \left[ \text{ER}(C^2_j, C^1_i), \text{ER}(C^2_j, C^1_i), \ldots, \text{ER}(C^2_j, C^1_i) \right], \right. \left. \left[ \text{SR}(C^2_j, C^1_i), \text{SR}(C^2_j, C^1_i), \ldots, \text{SR}(C^2_j, C^1_i) \right], \left[ \text{sR}(C^2_j, C^1_i), \text{sR}(C^2_j, C^1_i), \ldots, \text{sR}(C^2_j, C^1_i) \right] \right\}.$$

In cases where two or more concepts have the maximum value, arbitrary selection is carried out. The next step involves establishing logical consistency, i.e. the formation of a logically consistent merged ontology. The process of establishing the DL-based relation (equivalent, super, sub) between the concepts was explained in the chapter 6. The same process is now used to find the relationship of the new concept with existing concepts. The process of merging is explained next, with reasoning to establish logical consistency. The highest valued relationship is selected to determine how the new concept should be inserted into the existing ontology. In case no relation is found it is added as a new concept. The detailed explanation is given next.
7.4.2 Ontology merging procedure

The relational matrix determines the structural and semantic similarity between the concepts. Finding the maximum value relating a new concept with existing concepts not only reveals the closest concept in the existing ontology but also the type of relation i.e. equivalence, super or sub. Once both have been determined, logical consistency is checked next to determine the position of merging the new concept in the existing ontology.

7.4.2.1 Equivalence relation merging

Equivalence relation merging between the concepts is carried out when the type of similarity relation is equivalence. The simplest case is when the equivalence relation matrix value is one. In this case (Case 1) the new concept \( C_j^2 \) can be established in the existing ontology as equivalent to concept \( C_j^1 \) as shown in the figure 7.2.

![Figure 7:2 Case 1](image-url)
When the equivalence relation matrix value is less than one then a possible position for the new concept ($C_j^2$) is as a sibling of concept ($C_j^1$) (**Case 2**). This case arises when DL does not establish an equivalence relation between the two and furthermore, $C_j^2$ is the sub-concept of $SC_j^1$ but $C_j^1$ and $C_j^2$ do not have any common sub-concepts. As a result the equivalence relational matrix will have a greater value than the sub and super relational matrix in this case. The position of $C_j^2$ in the merged ontology has been shown in figure 7.3.

![Logical Consistency check:](image)

Case 1 and Case 2 add the new concepts in the existing ontology i.e. reconfigure the existing ontology to accommodate the new concept with logical consistency.
7.4.2.2 Super relation merging

This is where the new concept is in a super-concept relationship according to the relational matrix i.e. \( C_j^2 \supseteq C_j^1 \). In this case, three positions are possible where the new concept can be merged in the ontology, as shown in the figures 7.3 to 7.5.

The first condition (Case 3) arises when the new concept \( (C_j^2) \) is equivalent to the immediate super concept \( (SC_j^1) \) of the compared concept \( (C_j^1) \) as shown in figure 7.4.

Figure 7:4 Case 3
The second condition (case 4) arises when the new concept ($SC^i_j$) is a super-concept of the compared concept ($C^i_j$) and is also a sub-concept of the super concept ($SC^i_j$) as shown in the figure 7.5. This situation arises when a concept in an ontology is further subdivided or refined.

Figure 7:5 Case 4

Figure 7:6 Case 5
The third condition (case 5) arises when the new concept ($C_j^2$) is super concept of ($SC_i^1$), as shown in the figure 7.6. In this case, the position of ($C_j^2$) is above ($SC_i^1$), but to get the exact place $C_j^2$ must be compared with $SC_i^1$ and then the conditions (3) and (4) should be checked again.

7.4.2.3 Sub relation merging

This condition arises when the existing concept is in a super-concept relation according to the relational matrix i.e. $C_j^2 \subseteq C_i^1$. In this case, possible positions where the new concept can be merged in the ontology are shown in figures 7.7 to 7.10.

The first condition (case 6) arises when $C_j^2$ is a sub concept of $C_i^1$ and a super concept of $SC_i^1$ as shown in figure 7.6.

![Diagram](image)

**Logical Consistency check:**

$C_j^2 \subseteq C_i^1$

$SC_j^1 \subseteq C_j^2$

**Figure 7:7 Case 6**
The second condition (case 7) is when $C_j^2$ is a sub concept of $C_j^1$ and is equivalent to $sC_j^1$ as shown in figure 7.8.

![Logical Consistency check:](image)

The third condition (case 8) arises when $C_j^2$ is a sub concept of $C_j^1$ and is disjoint with $sC_j^1$ as shown in figure 7.9. This scenario describes the condition when a concept is redefined with the addition of new concepts (or new characteristics).

The last condition (case 9) describes the situation when $C_j^2$ is a subclass of $sC_j^1$ (figure 7.10). In this condition, to get the exact position of $C_j^2$, it must be compared with $sC_j^1$ and further evaluated for conditions 6-8.
Logical Consistency check:
\[ C^0_j \subseteq C^i_j \]
\[ C^0_j \land sC^i_j \equiv \bot \]

**Figure 7:9 Case 8**

Logical Consistency check:
\[ C^0_j \subseteq C^i_j \]
\[ C^0_j \subseteq sC^i_j \]

**Figure 7:10 Case 9**
Although this approach has considered all possible conditions for equivalence, sub and super relations, it may also be possible that the new concept has no defined position or possibly has no relation with existing concepts (including case 5 and 9). In this scenario merging and reconfiguration is carried out using the super concept of the new concept. Let $SC_j$ be the super concept of $C_j$ and the relational matrix is obtained in the same manner as in the case of $C_i$ and $C_j$. The following conditions can be obtained in line with the previous explanations (Prefix ‘Super’ (Ṡ) has been used to emphasise that the super concept of a new concept is compared to get the relational matrix):

### 7.4.2.4 Super equivalence relation

In a super-equivalence relation, the mapping of a new concept ($C_j$) in terms of its super-concept ($SC_j$) with respect to $C_i$ follows the same procedure as the mapping between $C_i$ and $C_j$. The simplest condition is when $SC_j$ is equivalent to $C_i$ (Condition Ṣ1) as depicted in figure 7.11. As no relation is found between $C_j$ and the existing ontology, this is simply added as a sub concept of $SC_j$ in the merged ontology.

![Logical Consistency check: $C_i = SC_j$, $C_j$ is new concept](Figure 7:11 Case Ṣ1)
Similar to condition 2, condition Š2 arises (figure 7.12) when a new concept $SC^2_j$ is added in the ontology as a sub-concept of $SC^1_i$ and $C^2_j$ is added in the ontology accordingly.

![Figure 7.12 Case Š2](image)

### 7.4.2.5 Super-super relation

Super-Sub-relation mapping is carried out when the relational matrix reveals that $SC^2_j$ is closer to the sub concept of the $C^1_i$. Condition Š3 (figure 7.13) and Š4 (figure 7.14) have same logical base as cases 3 and 4 respectively. Similar to condition 5, in the case of condition Š5 (figure 7.15), the super concept of $SC^2_j$ is checked with $C^1_i$ for conditions Š3 and Š4.
Logical Consistency check:
$SC_i = SC_j$

Figure 7:13 Case S3

Logical Consistency check:
$C'_i \subseteq SC^2_j$
$SC^2_j \subseteq SC^1_i$
$C^2_j \cap C^1_i = \bot$

Figure 7:14 Case S4
7.4.2.6 Super-sub relation

Super-super relation mapping occurs when the relational matrix intimates that $C_j^1$ is the super class of $SC_j^2$. All possible places where $SC_j^2$ and $C_j^1$ can fit have been shown in the figures 7.16-7.19. Conditions §6, §7 and §8 have similar logical explanations to conditions 6, 7 and 8 respectively.

Similar to condition 9, in condition §9 the exact position of $SC_j^2$ cannot be determined. It is compared with $sC_j^1$ and conditions 6,7 and 8 are checked with respect to $sC_j^1$ and $SC_j^2$. 

Figure 7:15 Case §5
Logical Consistency check:
\[ SC_j^2 \subseteq C_i \]
\[ SC_j^1 \subseteq SC_j^2 \]
\[ SC_j^1 \cap C_j^2 \equiv \bot \]

Figure 7:16 Case Š6

Logical Consistency check:
\[ SC_j^2 \subseteq C_i \]
\[ SC_j^1 \subseteq SC_j^2 \]
\[ SC_j^1 \cap C_j^2 \equiv \bot \]

Figure 7:17 Case Š7
If none of the conditions (Case 1 to Case 9 and Case Š1 to Case Š9) are satisfied in this process then it is clear that a new concept needs to be added in the ontology. For this, the process finds the super most concept, unrelated to the existing ontology, as a new concept and adds its sub-concepts accordingly.
7.5 IMPLEMENTATION METHOD

The proposed methodology for ontology merging and reconfiguration for both concrete and fuzzy domains can be created in an OWL API such as Protege (Protégé API). Merging and reconfiguration is carried out in Java. The overall implementation method is summarized in figure 7.20. Jena parser, a Java API is used as the Ontology API to get the concept names without the namespace. Wordnet (Wordnet API) is used to get the synonym, hyponym and hypernym of the concepts for carrying out the word similarity and finally calculating the Lexicon similarity matrix. A structural similarity matrix is calculated as previously described. Pellet-reasoner (Pellet: OWL 2 Reasoner) is
used to find the relationship between concepts (i.e. equivalence, super, sub etc.). As the two ontologies considered here are from the same domain, the assumption that they are built on same base ontology is valid. This assumption has been used for building the TBox and ABox for reasoners.

The semantic similarity matrix and structural similarity matrix are used to calculate the relational similarity matrix. The next step involves ontology merging with reconfiguration, restructure and consistency checks as illustrated in figure 7.21. Section 8.5 provides a detailed explanation of the implementation methods using example enterprise ontologies.
Figure 7:20 Merged ontology
// Start
{
    Step 1: Input two ontologies
    Step 2: Get Concept name
    Step 3: Get synonyms, hypernyms, hyponyms
    Step 4: Calculate Semantic, structural and relational matrix
    Step 5 for int j = 1 to J (j ∈ O2) s
        {
            k = argmax, ERM(Cj), SRM(Cj), sRM(Cj) }
            
            check case 1 to case 9 for (Cj, C^j) 
            if ! satisfied with all super-concept
            add as new concept
        }
    end //
}
CHAPTER 8 : HIGH PERFORMANCE VES: IMPLEMENTATION

8.1 INTRODUCTION

This chapter discusses implementation examples to demonstrate and test the research. The literature review in chapter 2, analysed the different phases of a VE lifecycle and their associated problems. Chapters 4-7 proposed the different mechanisms for enhancing the performance of individual enterprises and the overall VE with correct partner identification (Chapter 4), partner selection with risk analysis (Chapter 5), enterprise interoperability (Chapter 6) and enterprise knowledge enhancement (Chapter 7). This chapter shows how they can be applied and implemented in real life VEs.

VEs, as mentioned in the previous chapters, are common goal driven temporary alliances of independent enterprises situated locally or globally. The success of a VE not only depends on each of the individual enterprise’s skill, resource and competency but also on their ability to share these, plus appropriate information and knowledge with the other partners in the VE. Therefore, attainment of a VE’s goals depends on the successful execution of each of the VE lifecycle phases: Precreation, Creation, Operation and Termination. This chapter focuses on the implementation of efficient and effective execution of each of the VE lifecycle stages as proposed in chapters 4-7 and is organized as follows:

1. **Section 8.2** describes the process of identification of appropriate partners from the ECOS-DB ontology in accordance with the VE requirements. The ECOS-Query system, as explained in chapter 4, extracts the implicit and explicit information from the ECOS ontology to answers the queries. The process of implementation using an example and results obtained are also shown in this section.
2. **Section 8.3** describes the process of partner selection. Identified partners from section 8.2 are used for risk analysis and game theoretic revenue sharing analysis, as described in chapter 5, for optimal partner selection as explained in the chapter 5.

3. **Section 8.4** describes the process of achieving interoperability among the enterprises for effective and efficient transfer of information and knowledge. The proposed methodology uses a DL based approach, as described in chapter 6, to identify the interrelation between ontological entities. Two VE partners’ ontologies have been taken as an example for ontology mapping to achieve interoperability among the enterprises.

4. **Section 8.5** describes the knowledge enhancement process for the enterprises during the termination phase. Two ontologies have been taken as an example in the form of existing and new knowledge to demonstrate the ontology merging steps, as described in chapter 7. The process of reconfiguration and restructuring of an existing ontology to accommodate new knowledge has also been shown in the example.

For the implementation of the above demonstrations, a pool of 50 companies has been taken as a ECOS-DB from the Synergy project (Synergy, 2010) with slight modification such as competency description, data type role description etc. to suit the VE requirements. Protege (Protégé API), a GUI for OWL, has been used for modification and storage of the ECOS-DB.
8.2 ECOS-QUERY IMPLEMENTATION

This section presents the implementation part of the proposed ECOS-Query to extract explicit and implicit information from the ECOS ontology as described in chapter 4. Figure 8.1 depicts the ECOS ontology for companies’ database providing information regarding their competencies, skill, location, capacity etc. ECOS provides a platform for enterprises to publish their information on the web with a coherent structure and universally accepted terminologies. The ECOS ontology can be further extended to accommodate different functionality depending on the market requirements.

As explained earlier, the formation of a VE starts with the identification of a business opportunity and then the search for suitable enterprises, i.e. the partner identification stage. At this stage the ECOS ontology needs to be queried to extract explicit and implicit information regarding the enterprises. The ECOS ontology (Figure 8.1) presented in Protégé (Protégé API) with different competencies, skills etc. for 50 enterprises. Protégé can export the OWL ontology (ECOS ontology in this case) in RDF/XML format as an input file for JAVA programs. The ontology has been provided in RDF/XML format in the appendix. Consider the following scenario: A VE is to be created with requirements as shown in table 8.1 and this is translated into the following ECOS-Query:
\[ Q(a, b, c, d, e, f) = \]

\[
\text{Company}(a) \land \text{Broker}(a) \land \text{hasCompetencies}(a, \text{Project Management})\
\land \text{Company}(b) \land \text{hasCompetencies}(b, \text{Automation and Control}) \land
\neg \text{hasLocation}(b, \text{USA}) \land \text{Company}(c) \land \text{hasCompetencies}(c, \text{Manufacturing})\
\land \geq \text{hasCapacity}(c, 1700) \land \text{Company}(d) \land \text{hasCompetencies}(d, \text{Customer Care})\
\land \text{hasProduct}(d, \text{Logistics}) \land \text{hasLocation}(d, \text{London}) \land
\neg \text{hasExperiences}(d, \text{None}) \land \text{Company}(e) \land \text{hasCompetencies}(e, \text{Supplier})\
\land \{\text{hasSkill}(e, \text{Metal}) \lor \text{hasSkill}(e, \text{Steel})\} \land \text{Company}(f)\
\land \text{hasCompetencies}(f, \text{Software Development}) \land \neg \text{hasHumanResources}(f, \text{Low Skill})
\]

This has been implemented as shown in the figure 8.2.
Figure 8.1 ECOS Ontology in Protégé
<table>
<thead>
<tr>
<th>Company</th>
<th>Class Requirement</th>
<th>Role Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Class Name</td>
<td>Role</td>
</tr>
<tr>
<td>A</td>
<td>Company</td>
<td>hasCompetencies</td>
</tr>
<tr>
<td></td>
<td>Broker</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Company</td>
<td>hasCompetencies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hasLocation</td>
</tr>
<tr>
<td>C</td>
<td>Company</td>
<td>hasCompetencies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hasCapacity</td>
</tr>
<tr>
<td>D</td>
<td>Company</td>
<td>hasCompetencies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hasProduct</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hasLocation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hasExperiences</td>
</tr>
<tr>
<td>E</td>
<td>Company</td>
<td>hasCompetencies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hasSkill</td>
</tr>
<tr>
<td>F</td>
<td>Company</td>
<td>hasCompetencies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hasHumanResources</td>
</tr>
</tbody>
</table>

*Table 8.1 VE requirements*
According to the ECOS-Query, six enterprises are needed to form a VE with different competencies, skills, location etc. The ECOS ontology provides the information regarding the capacity, experience and human resources of an enterprise. However this query also requires a capacity, greater than some value to be found and the negation of the experience and resource information which are not explicitly mentioned in the ontology. To extract implicit information, such as this, requires the methods which were introduced and explained in chapter 4 and these have been implemented and tested as follows.

**Figure 8:2 Query written in JAVa**

```java
17  class ThesisTesting
18  {
19     public static void main (String[] args) throws OWLException, QueryParserException, QueryEngineException
20     {
21         String[] var = {"a", "b", "c", "d", "e", "f"}; //"a", "b";
22         String [] query = ("Company(f)", "hasCompetences(f, Software_Development)", "Company(e)", "hasCompetences(e, Supplier)", "Company(d)", "hasCompetences(d, Customer_Care)", "Company(e)", "hasCompetences(e, Metal)", "Company(d)", "hasCompetences(d, Steel)", "Company(e)", "hasCompetences(e, Customer_Care)"
23             "Company(c)", "hasCompetences(c, Manufacturing)", "Company(b)", "hasCompetences(b, Automation_and_Control)"
24             "Company(a)", "hasCompetences(a, Project_Management)"
25         );
26         String fileLocation = "C:\Users\mmkk\Desktop\ThesisCompDB.owl";
```
The ECOS-Query method has been coded in the JAVA and the SPARQL-DL (SPARQL-DL API) library for JAVA has been used for ABox and TBox queries. The ECOS-Query method starts with the query parsing and rearrangement. Query parsing
identifies the requirement of individual variables (a, b,.. in this case) and the query rearrangement process arranges the query according to the variable. Figure 8.2 shows the query in JAVA. OR, NOT, AND, GREATER, LESS and EQUAL prefixes are added before the queries to identify the type of queries as explained in chapter 4. The query rearrangement class in JAVA rearranges the queries in terms of class, properties and data according to the variable and has been shown in the figure 8.3.

The next step in the ECOS-Query involves query optimization. The query optimization process deletes the parts of the query which are redundant i.e. which do not affect the overall results of the query (Section 4.5). The result obtained in the query optimization part has been shown in the figure 8.4.

In this case, variable ‘a’ requires two classes: Company and Broker to be examined. These classes are associated with AND-Class and also Broker is a subclass of Company. Therefore, individuals satisfying the class Broker will automatically satisfy class Company. A query about the class Company is therefore redundant here and has been deleted by the Query optimization step in the ECOS-Query. After this step variable ‘a’ is queried only for class Broker and the rearranged query after Query optimization has been shown in the figure 8.5.
The next step in the ECOS-Query involves finding individuals satisfying the requirements of variables. For variable ‘b’ the ECOS-Query removes those individuals whose location is USA. Similarly for variable ‘c’ companies whose capacity is less than
1700 have been removed. For variable ‘d’ companies with experience ‘NONE’ have been removed. For variable ‘e’ companies with skill either steel or metal have been selected. For variable ‘f’ companies with low skill human resource have been removed. The final result obtained in the ECOS-Query has been shown in figure 8.6. Table 8.2 shows the outcome of the ECOS-Query and specifies the names of the companies fulfilling the requirements. This ECOS-Query has identified the possible set of enterprises capable of forming a VE according to the identified opportunity as specified in the scenario definition given at the start of this section. The next step in the VE lifecycle is to select the optimal partners. Inputs from the ECOS-Query are used to analyse the risk factors and game theoretic revenue sharing in the partner selection process. The next section describes the implementation of the partner selection process in this example VE.
-----Answer of a is:-----
<http://kmm.lboro.ac.uk/ecos/#Euro_Vision>
<http://kmm.lboro.ac.uk/ecos/#Europa_Group>
<http://kmm.lboro.ac.uk/ecos/#UK_VE_Ltd>
<http://kmm.lboro.ac.uk/ecos/#UK_Consortium>
-----Answer of b is:-----
<http://kmm.lboro.ac.uk/ecos/#Integral_Consulting>
<http://kmm.lboro.ac.uk/ecos/#Pertee_Associates>
<http://kmm.lboro.ac.uk/ecos/#Kiewit_Limited>
<http://kmm.lboro.ac.uk/ecos/#Berger_Firm>
-----Answer of c is:-----
<http://kmm.lboro.ac.uk/ecos/#Manson_Electronics>
<http://kmm.lboro.ac.uk/ecos/#DegenKolb_Inc>
<http://kmm.lboro.ac.uk/ecos/#KPFF_manufacturers>
<http://kmm.lboro.ac.uk/ecos/#Europol_Industries>
<http://kmm.lboro.ac.uk/ecos/#ABCJ_Limited>
<http://kmm.lboro.ac.uk/ecos/#Mark_and_Bon_Engg>
-----Answer of d is:-----
<http://kmm.lboro.ac.uk/ecos/#PLD_Groups>
<http://kmm.lboro.ac.uk/ecos/#Hatch-Mott_Engg>
<http://kmm.lboro.ac.uk/ecos/#PB_Groups>
<http://kmm.lboro.ac.uk/ecos/#Harriott_and_Smith_Ltd>
<http://kmm.lboro.ac.uk/ecos/#Paramatrix>
-----Answer of e is:-----
<http://kmm.lboro.ac.uk/ecos/#Hart_Crower_Networks>
<http://kmm.lboro.ac.uk/ecos/#J_and_H_Sales_Ltd>
<http://kmm.lboro.ac.uk/ecos/#ICI_Groups>
<http://kmm.lboro.ac.uk/ecos/#Parsons_Limited>
-----Answer of f is:-----
<http://kmm.lboro.ac.uk/ecos/#Pace_Groups>
<http://kmm.lboro.ac.uk/ecos/#UK_Computing_Services>
<http://kmm.lboro.ac.uk/ecos/#Midlands_IT_Networks>
<http://kmm.lboro.ac.uk/ecos/#Alan_and_Wyne>
<http://kmm.lboro.ac.uk/ecos/#Pulp_INC>

BUILD SUCCESSFUL (total time: 7 seconds)
<table>
<thead>
<tr>
<th>Company Variable</th>
<th>Capable Companies name</th>
</tr>
</thead>
</table>

Table 8:2 ECOS-Query results: capable companies for the formation of the VE
8.3 PARTNER SELECTION IN VE

Partner selection in a VE selects the most appropriate set of enterprises in terms of risk, cooperation, competencies etc. from the pool of identified sets of potential enterprises. The ECOS-Query, in the last section, identified the sets of potential partners capable of forming the VE. This section will illustrate the game theoretic approach for the selection of optimal partners as discussed in the chapter 5 for the development of a revenue sharing mechanism. This approach uses the collaborative performance risk, imposed collaborative risk and network risk to calculate the marginal contribution of enterprises based on value addition and risk sharing. Corollary 1, section 5.2.3, proves that the optimal partners possess the minimum risk associated cost. Theorem 1 along with corollary 1, section 5.2.3 splits the partner selection problem into two phases: first phase calculates the optimal partners based on the minimization of risk associated cost, whereas, second phase calculates the payoff (revenue) allocation based on marginal contribution and minimum payoff acceptance.

Theorem 2, section 5.4, further proves that for solutions in the first phase, the minimization of risk associated cost, requires $n$ relaxed quadratic problems to be solved sequentially, where $n$ is the number of sub-projects. The second phase, payoff allocation problem, uses the result from the first phase to solve a linear problem for optimal distribution of revenue. Section 5.5 discussed the sensitivity analysis of the VE which finds the degree of optimism, below which the VE will be infeasible from the interval value of risk due to the mismatch between minimum acceptance and marginal contribution. The quadratic and linear parts of the problem are solved using Matlab functions 'quadprog' and 'linprog' respectively.

Section 5.2 discussed the various risk quantification methods proposed in the literature. Enterprises can adopt any of the suitable methods fit for the VE requirements.
to calculate the different risks values. For example, stochastic analysis for risk preferences in different scenarios, fuzzy analysis with partial or incomplete information, questionnaire and ANP/AHP for network analysis. The application of such well-established methods is beyond the scope of this research and thesis. As mentioned in the chapter 5, this research assumes that the enterprises already have quantified the different risks values and therefore, for the testing of the proposed partner selection method risks values have been randomly generated here. This section first illustrates and discusses the results obtained from the various test problem sets and then discusses the solution for the current partner selection problem from section 8.2.

The test problem sets have been developed with different numbers of subprojects and bidding companies and varied from 4 to 9. The lower limit of the individual risk is generated randomly between 0 to 0.1. The individual values of each sub-project were varied between 10000-12000. The remaining values were also varied and the results obtained have been shown in table 8.3.

The experiment was carried out for the particular number of projects and companies until increasing any other parameter would make the problem infeasible. For each set of parameters, different optimistic values (1 to 0) were applied and the critical value of the solution was determined and as explained in the section 5.4, the solution became infeasible if the optimistic value became less than the critical value. If the critical value does not lies between 0 and 1 then a critical value does not exist (NE) as mentioned in the critical value column. Infeasible in the solution type (ST) column indicates that the marginal contribution is less than the expected profit (case 1 as explained in the section 5.3). Multiple solutions (MS) exist when the individual marginal contribution is greater than the expected profit and the collective contribution is less than the overall profit (case 2). A unique solution set is obtained when the individual marginal contribution is greater
<table>
<thead>
<tr>
<th>No.</th>
<th>Pro</th>
<th>Comp</th>
<th>Upper risk</th>
<th>Minimum acceptance</th>
<th>Network Risk</th>
<th>Cost</th>
<th>Optimism</th>
<th>Critical</th>
<th>Solution type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0-0.1</td>
<td>3000-4000</td>
<td>1</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>4</td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0.1-0.2</td>
<td>3000-4000</td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td>0.25-0.3</td>
<td>2500-3000</td>
<td>0.1-0.2</td>
<td>3000-4000</td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td>0.25-0.3</td>
<td>2500-3000</td>
<td>0.1-0.2</td>
<td>4000-5000</td>
<td>1</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>0.06</td>
<td>MS</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td>0.25-0.3</td>
<td>3000-4000</td>
<td>0.2-0.3</td>
<td>4000-5000</td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0-0.1</td>
<td>3000-4000</td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>41</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>43</td>
<td></td>
<td></td>
<td>0.3-0.4</td>
<td>2000-2500</td>
<td>0.1-0.2</td>
<td>3000-4000</td>
<td>1</td>
<td>0.15</td>
<td>MS</td>
</tr>
<tr>
<td>44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.17</td>
<td>MS</td>
</tr>
<tr>
<td>45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>0.17</td>
<td>MS</td>
</tr>
<tr>
<td>46</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>0.15</td>
<td>MS</td>
</tr>
<tr>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>0.029</td>
<td>MS</td>
</tr>
<tr>
<td>48</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>0.029</td>
<td>MS</td>
</tr>
<tr>
<td>49</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0.066</td>
<td>Infeasible</td>
</tr>
<tr>
<td>50</td>
<td></td>
<td></td>
<td>0.3-0.4</td>
<td>2000-2500</td>
<td>0.1-0.2</td>
<td>4000-5000</td>
<td>1</td>
<td>0.52</td>
<td>MS</td>
</tr>
<tr>
<td>51</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.15</td>
<td>MS</td>
</tr>
<tr>
<td></td>
<td>0.2-0.4</td>
<td>3000-4000</td>
<td>0.2-0.3</td>
<td>5000-6000</td>
<td>0.6</td>
<td>0.13</td>
<td>MS</td>
<td>0.8</td>
<td>0.13</td>
</tr>
<tr>
<td>---</td>
<td>---------</td>
<td>-----------</td>
<td>---------</td>
<td>----------</td>
<td>------</td>
<td>-------</td>
<td>-------</td>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>52</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>0.3-0.4</td>
<td>3000-4000</td>
<td>0.2-0.3</td>
<td>5000-6000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>58</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>61</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>62</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>63</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>0.2-0.25</td>
<td>1500-2200</td>
<td>0.0-0.1</td>
<td>3000-4000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>65</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>66</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>67</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>68</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>69</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>70</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0.0-0.1</td>
<td>3000-4000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>72</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>73</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>74</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>75</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>76</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>77</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>0.3-0.4</td>
<td>2000-2500</td>
<td>0.0-0.1</td>
<td>3000-4000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>79</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>81</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>82</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>83</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>84</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>0.3-0.4</td>
<td>2000-2500</td>
<td>0.1-0.2</td>
<td>4000-5000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>86</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>87</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>88</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>89</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>91</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>92</td>
<td>0.25-0.3</td>
<td>3000-4000</td>
<td>0.2-0.3</td>
<td>4000-5000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>93</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>94</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>95</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>97</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>98</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>99</td>
<td>0.2-0.25</td>
<td>1500-2000</td>
<td>0.0-0.1</td>
<td>3000-4000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>102</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>103</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

151
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>104</td>
<td>7</td>
<td>7</td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0.0-0.2</td>
</tr>
<tr>
<td>105</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ne Unique</td>
</tr>
<tr>
<td>106</td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>107</td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>108</td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>109</td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>110</td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>111</td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>112</td>
<td></td>
<td></td>
<td>0.3-0.4</td>
<td>2000-2500</td>
<td>0.0-0.2</td>
</tr>
<tr>
<td>113</td>
<td></td>
<td></td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>114</td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>115</td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>116</td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>117</td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>118</td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>119</td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>120</td>
<td></td>
<td></td>
<td>0.3-0.4</td>
<td>3000-4000</td>
<td>0.2-0.3</td>
</tr>
<tr>
<td>121</td>
<td></td>
<td></td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>122</td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>123</td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>124</td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>125</td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>126</td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>127</td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>128</td>
<td></td>
<td></td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0.0-0.1</td>
</tr>
<tr>
<td>129</td>
<td></td>
<td></td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>130</td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>131</td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>132</td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>133</td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>134</td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>135</td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>136</td>
<td></td>
<td></td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0.1-0.2</td>
</tr>
<tr>
<td>137</td>
<td></td>
<td></td>
<td>1</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>138</td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>139</td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>140</td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>141</td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>142</td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>143</td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>144</td>
<td></td>
<td></td>
<td>0.3-0.4</td>
<td>2500-3000</td>
<td>0.1-0.2</td>
</tr>
<tr>
<td>145</td>
<td></td>
<td></td>
<td>1</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>146</td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>147</td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>148</td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>149</td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>150</td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>151</td>
<td></td>
<td></td>
<td>0</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>152</td>
<td></td>
<td></td>
<td>0.3-0.4</td>
<td>3000-4000</td>
<td>0.2-0.3</td>
</tr>
<tr>
<td>153</td>
<td></td>
<td></td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>154</td>
<td></td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>155</td>
<td></td>
<td></td>
<td>0.6</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>156</td>
<td></td>
<td></td>
<td>0.5</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>157</td>
<td></td>
<td></td>
<td>0.4</td>
<td>NE</td>
<td>Unique</td>
</tr>
<tr>
<td>158</td>
<td></td>
<td></td>
<td>0.2</td>
<td>NE</td>
<td>Unique</td>
</tr>
</tbody>
</table>
Table 8.3 Experiment results

<table>
<thead>
<tr>
<th>Case</th>
<th>Value Range</th>
<th>Profit Range</th>
<th>Critical Value</th>
<th>Solution Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>156</td>
<td>0.2-0.25</td>
<td>2000-2500</td>
<td>0.1-0.1</td>
<td>3000-4000</td>
</tr>
<tr>
<td>157</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>159</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>166</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>167</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>168</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>169</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>170</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>171</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>172</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>174</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>176</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>178</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>182</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>184</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>185</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>190</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>193</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>194</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>195</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>196</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>197</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>198</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>199</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>203</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>204</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

than the expected profit but the collective contribution is greater than the overall profit (case 3).

Table 8.1 depicts the different types of solutions obtained during the experiment with varying parameters. For example in problem set 43, at optimistic value 1, a unique VE is formed with multiple solutions (MS) for the revenue sharing. This solution set has the critical value 0.15, i.e. in the future if the optimistic value goes below 0.15 due to the external or internal influences, the solution will become infeasible. A decrease in the
optimistic value increases the overall risks and decreases the marginal contribution of the companies driving the solution set from Unique to MS.

Problem set 49 indicates that the critical value is 0.066 and with zero optimism the solution becomes infeasible. In this case identify the enterprise responsible for infeasibility and then repeat the overall process. Similarly in the problem set 89, 90 and 91 optimistic values are in the range of the critical value making the solution infeasible. In problem set 161 and 162, changing the optimistic value from 1 to 0.8 changes the critical value from 0.05 to NE. This is due to the different set of solutions i.e. the different set of enterprises which are selected in both cases. It can be seen that on one hand the solution for problem 161 is optimal but when the optimistic value goes below 0.05 it becomes infeasible on the other hand the solution for problem 162 is robust for all values of optimism. This analysis provides an insight for the enterprises to analyze risks and profit in the VE with different partners when the risk evaluation is based on the incomplete information. This section next discusses the implementation part for the results obtained in section 8.2.

The ECOS-Query result identified the possible candidate partners for each subproject (Figure 8.6). The partner selection approach with risk minimization and revenue sharing mechanism is then applied with different values or parameters (risk, costs etc.) to find the optimal candidates for each subproject and thereby form a robust VE. The different values of parameters and results have been shown in Table 8.4
Table 8.4 Partner selection problem results

Problem sets 1-8, in table 8.4, show that the revenue sharing will be more than the marginal contribution so will have multiple solutions. Table 8.5 shows the actual results for the problem set 1. Column X is the solution set i.e. in problem set 1, for project 1, the first enterprise is selected, for subproject 3, the fourth enterprise is selected and so on. Columns MC and LB correspond to the marginal contribution and the lower bound of the revenue based on the minimum acceptance and costs. Column Rev corresponds to the actual shared revenue and the column Value corresponds to the actual value of the subproject. As the sum of the MCs is less than the sum of the values, the problem set 1 has multiple solutions.

<table>
<thead>
<tr>
<th>No.</th>
<th>LR</th>
<th>UR</th>
<th>NR</th>
<th>MA</th>
<th>Cost</th>
<th>Value</th>
<th>Opt</th>
<th>Crit.</th>
<th>ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0.2-0.25</td>
<td>0-0.1</td>
<td>1500-2000</td>
<td>3000-4000</td>
<td>10000-12000</td>
<td>1</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>0.8</td>
<td>NE</td>
<td>MS</td>
<td>0.6</td>
<td>0.5</td>
<td>0.4</td>
<td>NE</td>
<td>MS</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.1</td>
<td>2000-2500</td>
<td>3000-4000</td>
<td>10000-12000</td>
<td>1</td>
<td>NE</td>
<td>MS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>0.5</td>
<td>0.4</td>
<td>NE</td>
<td>UN</td>
</tr>
<tr>
<td>6</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.1</td>
<td>0.2-0.25</td>
<td>0-0.1</td>
<td>2000-2500</td>
<td>3000-4000</td>
<td>10000-12000</td>
<td>1</td>
<td>0.42</td>
<td>MS</td>
</tr>
<tr>
<td>9</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.34</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0.1-0.2</td>
<td></td>
<td></td>
<td>0.1-0.2</td>
<td>2000-2500</td>
<td>3000-4000</td>
<td>10000-12000</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.34</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>0.69</td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>0.69</td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>2500-3000</td>
<td>4000-5000</td>
<td>10000-12000</td>
<td>1</td>
<td>A</td>
<td>INF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.69</td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>0</td>
<td>0.1-0.2</td>
<td></td>
<td></td>
<td>0.1-0.2</td>
<td>2500-3000</td>
<td>4000-5000</td>
<td>10000-12000</td>
<td>1</td>
</tr>
<tr>
<td>24</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.69</td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>0</td>
<td>0.1-0.2</td>
<td></td>
<td></td>
<td>0.1-0.2</td>
<td>2500-3000</td>
<td>4000-5000</td>
<td>10000-12000</td>
<td>1</td>
</tr>
<tr>
<td>29</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.69</td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>0</td>
<td>0.1-0.2</td>
<td></td>
<td></td>
<td>0.1-0.2</td>
<td>2500-3000</td>
<td>4000-5000</td>
<td>10000-12000</td>
<td>1</td>
</tr>
<tr>
<td>34</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td>0.8</td>
<td>0.69</td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>0</td>
<td>0.1-0.2</td>
<td></td>
<td></td>
<td>0.1-0.2</td>
<td>2500-3000</td>
<td>4000-5000</td>
<td>10000-12000</td>
<td>1</td>
</tr>
<tr>
<td>No.</td>
<td>Sub-project.</td>
<td>X</td>
<td>MC</td>
<td>LB</td>
<td>Rev</td>
<td>Value</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>--------------</td>
<td>----</td>
<td>------</td>
<td>------</td>
<td>-----</td>
<td>-------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10016</td>
<td>5583</td>
<td>&gt;MC</td>
<td>10448</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>8581</td>
<td>5105</td>
<td>&gt;MC</td>
<td>11336</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4</td>
<td>8846</td>
<td>5657</td>
<td>&gt;MC</td>
<td>11689</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>8186</td>
<td>5591</td>
<td>&gt;MC</td>
<td>10689</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1</td>
<td>9126</td>
<td>4895</td>
<td>&gt;MC</td>
<td>11561</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>3</td>
<td>11107</td>
<td>5126</td>
<td>&gt;MC</td>
<td>11351</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>1</td>
<td>11118</td>
<td>5921</td>
<td>10492</td>
<td>10448</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>11694</td>
<td>5407</td>
<td>10717</td>
<td>11336</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>6</td>
<td>12139</td>
<td>5722</td>
<td>11121</td>
<td>11689</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2</td>
<td>11458</td>
<td>5576</td>
<td>10611</td>
<td>10689</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1</td>
<td>12334</td>
<td>5178</td>
<td>11081</td>
<td>11561</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>4</td>
<td>15180</td>
<td>5224</td>
<td>13052</td>
<td>11351</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td>1</td>
<td>5337</td>
<td>6610</td>
<td>INF</td>
<td>10448</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>7154</td>
<td>7358</td>
<td>INF</td>
<td>11336</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>5</td>
<td>7986</td>
<td>6807</td>
<td>INF</td>
<td>11689</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3</td>
<td>7165</td>
<td>7225</td>
<td>INF</td>
<td>10689</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4</td>
<td>7899</td>
<td>6821</td>
<td>INF</td>
<td>11561</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>3</td>
<td>9548</td>
<td>6831</td>
<td>INF</td>
<td>11351</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 8.5 Results of problem set 1, 9 and 22

Problem sets 9-14, in table 8.4, show unique solutions and the corresponding results of problem set 9 have been shown in table 8.5. The sum of MC is greater than the sum of values leading to the unique solution for the revenue sharing problem. Problem set 19-28, in table 8.4, shows the infeasibility of the solution. In problem 19-21, critical values are in the range of the optimistic value making the solution infeasible. Problem set 22-28, for all values of optimism are infeasible. The corresponding solution of problem 22 has been shown in table 8.5. The MC of subprojects 1, 2 and 4 are less than their corresponding LBs making the problem infeasible.

The selection of optimal partners depends on the different parameters i.e. risks values, costs, network optimism value etc. Assuming the parameters are equal to the parameters described in the problem set 9, a unique solution is obtained as shown in table 8.6.
### 8.4 Interoperability in VE

As discussed in the chapters 2 and 6, semantic interoperability remains a major issue during the operational phase of a VE. Semantic interoperability plays an important role in the success of a VE by facilitating the effective and efficient communication, collaboration and cooperation. Chapter 6 described the methodology for achieving logically consistent semantic interoperability based on description logic. This section provides the implementation part of the proposed methodology.

The methodology has been described in section 6.5. To illustrate the whole process, two basic ontologies have been developed, one for a manufacturing enterprise (Enterprise ontology A, Mark and Bonn Engg.) and the other for a customer care enterprise dealing with marketing (Enterprise ontology B, Hatch Mott Engg.)(figure 8.7). These ontologies have been mapped to assist collaboration between enterprises. The ontologies were developed using Protégé and exported as owl files to enable access by java APIs and by reasoners (Pellet and SPARQLDL Java API) to find the relationships between the entities.
The first step of the procedure starts with input of atomic level relationships among the entities of the two ontologies in the java program using the TboxSimilarity class as a starting point in the methodology. Figure 8.8 depicts the insertion of similarity concepts in the ontology mapping method. For example, the first three lines assert that the \( \text{EntA:Product} \equiv \text{EntB:Product} \), similarly the next three lines assert that the \( \text{EntA:Repair} \equiv \text{EntB:Rectify} \) and so forth. The next steps involved finding the lexical similarity between the entities using wordnet. Lexically similar entities are further logically tested (as described in section 6.4) to obtain the relationship (equivalent, super, sub) as described next.
Figure 8:7 Two different ontologies
8.4.1 Concepts vs. Concepts

The processes for deriving concept level relationships (or bridging axioms) have been explained in section 6.4.1 and these have been implemented to determine the lexically similar concepts. The methodology finds that the concepts \( \text{EntA:Service} \) and \( \text{Ent:B:Service} \) are similar and their relationship has been derived as follows:
From the definition of concept $EntA: Service$ in the enterprise ontology A:

$EntA: Service \equiv EntA: Repair \cup EntA: Replacement \hspace{1cm} (8.1)$

From the definition of concept $EntB:Service$ in the enterprise ontology B:

$EntB: Service \equiv EntB: Rectify \cup EntB: Renewal \hspace{1cm} (8.2)$

From the atomic level similarity input (Figure 8.8):

$EntA: Repair \equiv EntB: Rectify \hspace{1cm} (8.3)$

$EntA: Replacement \equiv EntB: Renewal \hspace{1cm} (8.4)$

From equations 8.1-8.4 the inference $EntA: Service \equiv EntB: Service$ can be established and the assertion in the program is shown in the figure 8.9.

<table>
<thead>
<tr>
<th>Equivalent Class Assertion</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{EntA: Service}$</td>
</tr>
<tr>
<td>$\text{EntB: Service}$</td>
</tr>
</tbody>
</table>

Figure 8.9 Equivalent class assertion

In many cases, as described in the chapters 2 and 6, heterogeneity occurs in the ontology mapping due to sub-class conflicts i.e. there is a different level of sub-concepts divisions. As shown in the figure (8.10), the lexically similar concepts Dimension in ontology A and B have different levels of sub-concepts division. $EntA:Dimension$ has six sub-concepts, whereas $EntB:Dimension$ has no sub concept. Using the TBox definition and atomic level similarity input as:
Using the equations 8.5-8.6 the following can be inferred (Figure 8.11)

\[
\text{EntB : Dimension} \subseteq \text{EntA : Bore} \cup \text{EntA : Breadth} \cup \text{EntA : Depth} \cup \\
\text{EntA : Height} \cup \text{EntA : Length} \cup \text{EntA : Radius}
\]
8.4.2 Role vs. role

This section considers the implementation of the proposed methods, as described in the section 6.4.2, to find the correspondence between properties or roles and checks the correspondence between the range and domain of roles. Figure 8.12 shows the definition of lexically similar role \textit{hasDimension} in both ontologies in terms of their domain and range concepts and can be written as:

\begin{equation}
\text{EntA : hasDimension(EntA : Product, EntA : Dimension) .........................(8.7)}
\end{equation}

\begin{equation}
\text{EntB : hasDimension(EntB : Product, EntB : Dimension) .........................(8.8)}
\end{equation}

Here the first part in the brackets states the domain and second part states the range of the role. Using the result (8.6) and atomic similarity input, the following is established:

\begin{equation}
\text{EntA : Product} \equiv \text{EntB : Product} ...........................................(8.9)
\end{equation}

\begin{equation}
\text{EntA : Dimension} \supseteq \text{EntB : Dimension} ...........................................(8.10)
\end{equation}

Using the equation 8-9, 8.10 and from section 6.4.2.1.b following can be inferred

\begin{equation}
\text{EntA : hasDimension} \supseteq \text{EntB : hasDimension}
\end{equation}
This has been asserted in the methodology with the bridging axiom as shown in the figure 8.13.

8.4.3 Concepts vs. Property

As mentioned in the chapter 6, a concept in one ontology can be described as role in another ontology. In order to determine the correspondence between these two, analysis of concepts and range and domain of role must be examined. Figure 8.14 shows the lexically similar concept EntA:Troubleshooting in the ontology A and role EntB:hasTroubleshooting in terms of its domain and role concepts in ontology B. Their correspondence is inferred as follows:

--- Sub Property Assertion ---
<http://www.owl-ontologies.com/EntB.owl#hasDimension> is sub propert of <http://www.owl-ontologies.com/EntA.owl#hasDimension>

Figure 8:13 Property assertion.

Figure 8:14 Class vs. Property comparison
$EntB: \text{hasTroubleshooting}(EntB: \text{Product}, EntB: \text{Malfunctioning}) \quad \ldots .(8.11)$

$EntA: \text{Product} \equiv EntB: \text{Product} \quad \ldots .(8.12)$

$EntA: \text{TroubleShooting} \equiv EntA: \text{Malfunctioning} \quad \ldots .(8.13)$

Using the equations 8.11-8.13 following can be inferred:

$EntA: \text{TroubleShooting} \equiv EntB: \text{Product} \cap$

$\forall EntB: \text{hasTroubleShooting}, EntB: \text{Malfunctioning}$

The corresponding bridging axiom added in the methodology as shown in the figure 8.15.

<table>
<thead>
<tr>
<th>Class Equivalent to Property</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.owl-ontologies.com/EntA.owl#TroubleShooting">http://www.owl-ontologies.com/EntA.owl#TroubleShooting</a> is equivalent to <a href="http://www.owl-ontologies.com/EntB.owl#hasTroubleShooting">http://www.owl-ontologies.com/EntB.owl#hasTroubleShooting</a></td>
</tr>
</tbody>
</table>

**Figure 8:15 Class equivalent property assertion**

With similar logic, the correspondence between lexically similar $EntA: \text{Strength}$ and $EntB: \text{hasStrength}$ is inferred and asserted as shown in the figure 8.16:

$EntA: \text{Strength} \equiv EntB: \text{Product} \cap \forall EntB: \text{hasStrength}, EntB: \text{Strength}$

<table>
<thead>
<tr>
<th>Class Equivalent to Property</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.owl-ontologies.com/EntA.owl#Strength">http://www.owl-ontologies.com/EntA.owl#Strength</a> is equivalent to <a href="http://www.owl-ontologies.com/EntB.owl#hasStrength">http://www.owl-ontologies.com/EntB.owl#hasStrength</a></td>
</tr>
</tbody>
</table>

**Figure 8:16 Class equivalent to property assertion**

Final mapping results are added in the merged ontology in the form of axioms. Figure 8.17 shows the mapped ontology achieved with this process.
Figure 8:17 Mapped global ontology
8.5 KNOWLEDGE INTEGRATION IN VE

Chapter 2 and chapter 7 discussed the importance of knowledge acquisition and knowledge integration for enterprises. Member enterprises involved in a VE gather valuable knowledge during the operational phase and it is important to incorporate discovered knowledge into existing knowledge. Chapter 7 discussed the DL based ontology merging technique to accommodate discovered knowledge with logical verification. This section explains the implementation procedure for discovered knowledge by ontology reconfiguration and ontology restructuring.

In order to illustrate the overall procedure, two ontologies in the manufacturing domain of car manufacturing have been developed in Protégé, as shown in figure 8.18. The Car ontology describes the current knowledge of the field, whereas the New Car ontology represents new knowledge in the field of car manufacturing. In order to merge the two ontologies, reconfiguration or restructure of an existing ontology (Car Ontology) is carried out to incorporate the new knowledge (New Car ontology) using the process described in the chapter 7.

The first step in this methodology starts with the concept names identified with the help of the Jena parser. Initial TBox similarity among the concepts of two ontologies has been provided, similar to figure 8.8, as shown in the figure 8.19. The next step involves calculating the similarity matrix. This step calculates the Lexicon similarity matrix and Structural similarity. For example, consider the two concepts: Water from the Car ontology (C: Water) and Oil from the New Car ontology (NC: Oil). As there is no similarity between the two concepts in terms of synonyms, hypernyms and hyponyms, their lexicon similarity:
Figure 8.18 Input Ontologies
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Car")),
    fac.getOWLClass(IRI.create(ont2NS+"Car")))

//2
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Brake")),
    fac.getOWLClass(IRI.create(ont2NS+"Brake")))

//3
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Drum_Brake")),
    fac.getOWLClass(IRI.create(ont2NS+"Drum_Brake")))

//4
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Power_Brake")),
    fac.getOWLClass(IRI.create(ont2NS+"Power_Brake")))

//5
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Engine")),
    fac.getOWLClass(IRI.create(ont2NS+"Engine")))

//6
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Valve")),
    fac.getOWLClass(IRI.create(ont2NS+"Valve")))

//7
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Fuel Injector")),
    fac.getOWLClass(IRI.create(ont2NS+"Fuel Injector")))

//8
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Cooling")),
    fac.getOWLClass(IRI.create(ont2NS+"Cooling")))

//9
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Safety")),
    fac.getOWLClass(IRI.create(ont2NS+"Safety")))

//10
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Steering")),
    fac.getOWLClass(IRI.create(ont2NS+"Steering")))

//11
new TboxSimilarity().equivalentClassAddition(go, goNS,
    fac.getOWLClass(IRI.create(ont1NS+"Steering System")),
    fac.getOWLClass(IRI.create(ont2NS+"Steering System")))

Figure 8:19 Tbox input
\( \psi(C:Water, NC:Oil) = 0. \)

For calculating the structural similarity, super and sub concepts need to be identified. Super concept of \( C:Water \) are:

\( S(C:Water) = \{ \text{Cooling, Engine, Car} \}. \)

Sub concepts of \( C:Water \) are:

\( s(C:Water) = \emptyset. \)

Similarly, super and sub concepts of concept \( NC:Oil \) are:

\( S(NC:Oil) = \{ \text{Cooling, Engine, Car} \}. \)

\( s(NC:Oil) = \emptyset. \)

The next step involves the calculation of structural similarity values: Equivalent relation similarity (\( ER \)), Super relation similarity (\( SupR \)) and Sub relation similarity (\( subR \)) as described in section 7.2 as follows:

\[
\begin{align*}
ER(C:Water, NC:Oil) &= 0.5 \left( \frac{\text{Sim}(SC_i^1, SC_j^2)}{|SC_i^1 \cup SC_j^2|} \right)^2 + 0.5 \left( \frac{\text{Sim}(sC_i^1, sC_j^2)}{|sC_i^1 \cup sC_j^2|} \right)^2 \\
\end{align*}
\]

As sub-concepts of both the concepts are empty,

\( ER(C:Water, NC:Oil) = \left( \frac{3}{3} \right)^2 = 1 \)

Similarly, as both sub-concepts sets are empty it follows that:

\( SupR(C:Water, NC:Oil) = 0 \) and

\( subR(C:Water, NC:Oil) = 0. \)

The next step involves the calculation of Ontology similarity index (matrix): Equivalent similarity matrix (\( ER \)), Super similarity matrix (\( SR \)) and sub similarity matrix (\( sR \)) as discussed in the section 7.3 as follows:

Considering equal weightage for semantic and structural similarity
\[ \mathcal{ER} (C: Water, NC: Oil) = 0.5 + 0.5 = 0.5 \]

Similarly,
\[ \mathcal{SR} (C: Water, NC: Oil) = 0 \quad \text{and} \quad \mathcal{sR} (C: Water, NC: Oil) = 0. \]

In a similar manner the similarity matrix is calculated between ‘NC: Oil’ and all the concepts of Car Ontology to get \( \mathcal{ERM} \), \( \mathcal{SRM} \) and \( \mathcal{sRM} \) (section 7.3). The non-zero values obtained are:
\[ \mathcal{ER} (C: Water, NC: Oil) = 0.5 \quad \text{and} \quad \mathcal{ER} (C: air, NC: Oil) = 0.5. \]

The final step involves the merging of the new concept into the existing concept. As both the similarity matrix indexes have the same value i.e. 0.5, the algorithm arbitrarily selects one of them and tries to merge it logically into the existing concept as explained in section 5. Taking ‘C: water’ the logical relations obtained are:
\[
\begin{align*}
C: Cooling \equiv NC: Cooling \quad \text{(From TBox similarity)} \quad \text{……………….. (a)} \\
NC: Oil \subseteq \{NC: Cooling \equiv C: Cooling\} \quad \text{………………….. (b)} \\
NC: Oil \cap C: Water \equiv \bot \quad \text{………………………………… (c)}
\end{align*}
\]

Clearly, conditions (a), (b) and (c) lead to the case (2) (from chapter 7) and ‘NC: Oil \subseteq C: Cooling’ is established as shown in the figure 8.20. In the case of the same similarity index occurring between more than one concept, a random selection process has been adopted as \( C: Water \) is selected instead of \( C: air \) here, although a similar relation can be derived using \( C: air \) and \( NC: Oil \). Table 8.7 depicts the maximum similarity between the concepts of new car ontology with car ontology, types of similarity, logical case obtained and logical relation asserted in this process. The final merged ontology has been shown in the figure 8.20 which represents the logical consistent merging of the car ontology (existing knowledge) with the new car ontology (discovered knowledge.)
<table>
<thead>
<tr>
<th>New Concept</th>
<th>Max similarity with Existing Concept</th>
<th>Type of similarity</th>
<th>Logical Case</th>
<th>Final relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>NC: Car</td>
<td>C: Car</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Brake</td>
<td>C: Brake</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Drum_Brake</td>
<td>C: Drum_Brake</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Double_Edge</td>
<td>C: Single_Leading_edge</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Drum_Brake</td>
</tr>
<tr>
<td>NC: Power_Brake</td>
<td>C: Power_Brake</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Electro_Hydraulic</td>
<td>C: Air_Suspended</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Power_Brake</td>
</tr>
<tr>
<td>NC: Hydraulic</td>
<td>C: Vaccum_Suspended</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Power_Brake</td>
</tr>
<tr>
<td>NC: Engine</td>
<td>C: Engine</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Cooling</td>
<td>C: Cooling</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Oil</td>
<td>C: air</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Cooling</td>
</tr>
<tr>
<td>NC: Fuel Injector</td>
<td>C: Fuel Injector</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Multi_PointInjector</td>
<td>C: Direct_Injection</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Fuel Injector</td>
</tr>
<tr>
<td>NC: Valve</td>
<td>C: Valve</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Tapped_Valve</td>
<td>C: Spring_valve</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Valve</td>
</tr>
<tr>
<td>NC: Safety</td>
<td>C: Safety</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Anti_Skid_Brake</td>
<td>C: Seat_belt</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Safety</td>
</tr>
<tr>
<td>NC: Back_Camera</td>
<td>C: Air_bags</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Safety</td>
</tr>
<tr>
<td>NC: Fog_Light</td>
<td>C: Seat_belt</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Safety</td>
</tr>
<tr>
<td>NC: Reverse_Backup_Camera</td>
<td>C: Automated_Braking</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Safety</td>
</tr>
<tr>
<td>NC: Steering_Wheel_Control</td>
<td>C: Air_bags</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Safety</td>
</tr>
<tr>
<td>NC: Steering</td>
<td>C: Steering</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Power_Steering</td>
<td>C: steering</td>
<td>Sub-class</td>
<td>Case $$1</td>
<td>C: Steering</td>
</tr>
<tr>
<td>NC: Pump</td>
<td>C: steering</td>
<td>Sub-class</td>
<td>Case $$1</td>
<td>C: Power_Steering</td>
</tr>
<tr>
<td>NC: Reservoir</td>
<td>C: steering</td>
<td>Sub-class</td>
<td>Case $$1</td>
<td>C: Power_Steering</td>
</tr>
<tr>
<td>NC: Rotary_Valve</td>
<td>C: steering</td>
<td>Sub-class</td>
<td>Case $$1</td>
<td>C: Power_Steering</td>
</tr>
<tr>
<td>NC: Steering_System</td>
<td>C: Steering_System</td>
<td>TBox</td>
<td>-</td>
<td>Equivalent</td>
</tr>
<tr>
<td>NC: Cam_and_Lever</td>
<td>C: Rack_and_pinion</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Steering_System</td>
</tr>
<tr>
<td>NC: Worm_and_Roller</td>
<td>C: Worm_and_Nut</td>
<td>Equivalent</td>
<td>Case 2</td>
<td>C: Steering_System</td>
</tr>
</tbody>
</table>

Table 8.7 Merging process outcome
Figure 8.20. Merged / Reconfigured Ontology
CHAPTER 9 : CONCLUSION AND FUTURE SCOPE

9.1 RESEARCH OBJECTIVES AND CONTRIBUTIONS

As VEs are market driven short term collaborations of independent and autonomous enterprises, their success depends upon the efficient and effective execution of their lifecycle. Chapter 2 analyzed the lifecycle of a VE: ‘Precreation, Creation, Operation and Dissolution’ and identified the research objectives for the formation of high performance VE and enhancement of enterprises competency. The detailed discussion on research objectives and contributions made for each objective are as follows:

Research Objective 1:

- To develop the DL-based query system, which can extract explicit and implicit information from an ECOS-ontology to identify the potential partners in the pre-creation stage.

A VE lifecycle starts with the identification of market opportunity. In the precreation stage a potential set of partners, fulfilling the VE requirements, needs to be identified for the formation of the VE. Chapter 2, section 2.3.1 described the benefits of the ECOS ontology for publishing the enterprise competency with structured and organized semantics. ECOS uses the OWL, a DL based ontology, to develop, maintain and publish information in machine readable RDF/XML format (Khilwani et al., 2011). In the pre-creation stage, the ECOS ontology must be queried to satisfy the VE requirements to identify suitable partners. As explained in the section 3.2.1, the DL-based query language (SPARQL-DL) does not support the data type complex queries and negation of concepts (roles) where negation is not defined explicitly in the ontology. The first objective in this research therefore required the development of a DL-based query
method (ECOS-DL query) which can extract explicit and implicit information from an ECOS-ontology.

**Contribution 1:**

The development of a DL-based ECOS-Query system is discussed in the chapter 4. This query system can extract both explicit and implicit information from ontology based enterprise information systems such as ECOS and it is referred to as the ECOS-Query system in this thesis. Current DL-based ontology query systems cannot extract information if it is not explicitly mentioned in the ontology and do not support customized data-type queries (Pan, 2004). The ECOS-Query system, developed in this thesis, overcomes these limitations and can extract implicit information and customized data type information. The overall procedure is divided into three steps: the first step starts with the query rearrangement where queries are arranged according to the variables. NOT and DATA prefixes are used to identify the implicit nature of the queries. In the second step, query optimization exploits the DL-based interrelation between the concepts and roles and removes the redundant parts of the queries which do not affect the final result. In the final step the query is processed using the SPARQL-DL API to get explicit and implicit information from the ECOS-ontology. An implementation of the ECOS-Query system was developed to test and demonstrate this contribution and this has been shown in detail in section 8.2.

**Research Objective 2:**

- To identify a method for the analysis of different risks which may occur during the creation phase of a VE.

The success of a VE and its achievement of objectives depends on the individual partners’ capabilities and their cooperative relationships, in other words, on the optimal
selection of partners. Section 2.3.2 described the importance of risk analysis in VE formation. Enterprises collaborating to form a VE generally reduce the performance risk but increase the relational risk without any controlling mechanism. The second objective set in this thesis therefore required the analysis of different risks present in the formation of a VE and the study of the different risks in terms of cost increment and profit decrement with minimization of the overall risks.

**Contribution 2:**

Chapter 5, section 5.2 describes the risk analysis in partner selection. This approach extends the risk analysis in partner selection from the augmented enterprise paradigm (Huang and Chen, 2005; Li and Liao, 2007) to the VE paradigm by analyzing risks not only from the VE perspective but also an individual enterprise perspective. Assuming that the different risks are already quantified, this risk analysis process divides the identified risks in partner selection in a VE into two major categories: Performance risks and Network risks. A performance risk is a measure of an individual enterprise’s capacity and capability to perform a task whereas a network risk is the measure of interdependency among enterprises in a VE in which lack of trust, inaccurate information sharing etc. hinders the effective collaboration. Performance risk is further subdivided into individual, collaborative and imposed collaborative performance as shown in the figure 5.1. Each individual risk is associated with quality and capacity constraints of individual enterprises. Collaborating performance risk is the measure of collective quality and capacity constraints of a VE. Imposed collaborative risk measures the risk induction among the member enterprises with the selection of an enterprise. Quantification of different risks and their effect on costs and profits have been further explained in the section 5.2. Performance risk and imposed collaborative risk are measures of the
respective direct and indirect risk contributions of individual enterprises in the VE whereas, network risk and collaborative performance risk are measures of direct and indirect risk respectively of VE. Direct and indirect risk analysis is further used to calculate the enterprises input in the VE.

**Research Objective 3:**

- *To develop a revenue sharing mechanism for optimal partner selection in the creation phase.*

The general structure of a VE is very different from traditional enterprises with a centralized control system. Therefore, it is difficult to determine the level of effort needed from member enterprises and a fixed share distribution without analyzing the effort invites dissatisfaction among the enterprises. Therefore, as discussed in section 2.3.2, an appropriate mechanism is needed to address the following issues:

1. Determining the effort put forward by enterprises in terms of risks and value addition.
2. Division of revenue based on the effort provided by the enterprises.

**Contribution 3:**

Chapter 5, section 5.3 explains the game theoretic aspects and develops the revenue sharing mechanism. The characteristics of VEs are similar to cooperative game theory analysis as enterprises pool their resources for larger global benefit. Although cooperative game theory has been widely used in the collaborative network analysis (Nagarajan and Sosic, 2008) to optimize supply chain, price setting, risk sharing, inventory pooling etc. (Garnot and Sosic, 2003; Cachon and Netessine; 2004; Reinhardt and Dada, 2005) it has not been used in the partner selection process. The proposed
method first calculates the marginal contribution of each enterprise based on cost, profit and direct and indirect risk contribution. Cooperative game theory in collaborative network analysis uses the direct contribution, risk and other factors to optimize the overall network, but indirect contribution in terms of contribution, risk, cost etc. is also a very important factor in VEs due to determine the contribution of individual enterprises. The mechanism developed here considers both direct and indirect contribution of enterprises. Direct risk contribution is the calculation of increased operational cost whereas indirect risk contribution is the calculation of 1. Risks enterprises are bringing in the VE and 2. Risk enterprises are undertaking in the VE Theorem 1 along with the corollary 1 proves that the optimal partners have the minimum risk associated cost. Theorem 2 further breaks the Non-Linear integer problems into n-sub problems. Section 5.5 presents the methodology for sensitivity analysis for measuring the robustness of the VE. Section 8.2 describes the implementation for testing and demonstrating this methodology on a generated case study data set.

**Research Objective 4:**

-To develop a DL-based ontology mapping technique to achieve interoperability in the operation phase.

Chapter 2, section 2.3.3 explained the importance of interoperability among enterprises during the operation phase. Semantic interoperability ensures that the information transferred between the enterprises should be understood with the correct intention. Simple correspondence between the two ontologies produces logical inconsistencies and different heterogeneity as explained in that section. The fourth
objective set in this research is to provide ontology mapping in the DL paradigm for achieving interoperability in the VE during its operation phase.

**Contribution 4:**

Chapter 6 describes the DL-based methodology for ontology mapping using bridging axioms between the entities of the different ontologies. The process of logical derivation of bridging axioms has been explained in section 6.4. The contribution made in this thesis is to extend the DL-based consistent mapping technique of Dou and McDermott, (2006) in an efficient manner and to carry out the mapping at the concept level, role level and role-concept level with logical consistency. Section 8.4 shows the process of mapping with two different ontologies and thereby tests and demonstrates the contribution.

**Research Objective 5:**

- *To develop a DL-based ontology merging technique for knowledge enhancement of enterprises during the termination phases.*

Chapter 2, section 2.3.4 explains the importance of knowledge management for enhancing the core-competency of an enterprise. Therefore, knowledge discovered during the life of the VE needs to be disseminated among enterprises during the dissolution phase of the VE. As further explained in that section, an ontology merging technique is required to merge a new ontology semantically and structurally with the existing ontologies with the following characteristics:

4. Deduce similar or new concepts
5. Deduce the possibility of merging concepts, i.e. by restructuring an ontology.
6. Achieve logically consistent mappings.
Contribution 5:

Chapter 7 describes the proposed methodology for ontology merging. Various approaches have been reported in the literature for ontology merging to enhance the enterprises knowledge by constructing the new ontology (Raunich and Rahm, 2011; Chen et. al., 2011). Consistency of the merged ontology based on logical structure needs to be addressed in the ontology merging (Chen et. al., 2011). The ontology merging approach developed in this thesis merges the new knowledge into the existing knowledge by restructuring or reconfiguring the existing ontology rather than building a new one. This technique first measures the semantic and structural similarity in terms of equivalent, super and sub concepts. In the next step, it tries to find the position for the new concept in the existing ontology with the help of a DL-based consistency check mechanism. Section 8.5 shows the implementation part of this with the help of two ontologies representing existing and discovered knowledge. DL-based logical consistency check ensures that the merged ontology is valid. Section 8.5 shows the process of merging with two different ontologies and thereby tests and demonstrates the contribution.

9.2 Threats to validity

This research, on supporting SMEs in collaboration, especially VEs, to enhance the performance of individual enterprises and VEs, is based on various available results and ontology representation. The validity of this research is subject to the following assumptions:

1. The risk analysis section (chapter 5) assumes that the different risks in the VE formation have already been identified and quantified correctly. Any
deficiency in risk identification will affect the quality of results obtained from the proposed method for risk analysis.

2. This research further assumes that the ontology representation is carried out using a two variable system, for example OWL. The proposed methodologies (ECOS-Query, ontology mapping and ontology merging) will not be effective if the ontology representation system or software uses more than two variables to define concepts and roles such as SWRL in OWL.

3. The ontology mapping (Chapter 6) and ontology merging (Chapter 7) process assumes that the atomic level similarity is available or can simply be derived using the domain knowledge. The relationships between complex concepts and roles are derived using atomic level similarity and hence if atomic level similarity is unavailable, theses methods will not be able to produce the desired result.

9.3 FUTURE EXTENSION

The potential areas for the extension of this work could be:

1. Development of the DL-query, ontology mapping and merging with complex definition of concepts and roles. In many areas of ontology application complex definition of concepts and roles are necessary to express the information such as inverse and transitive. ECOS-Query, ontology mapping and ontology merging can be further explored to accommodate these complex definitions.

2 Development of the DL-query, ontology mapping and merging with more expressive DLs. More expressive DLs such as constructors with cardinality
restriction, qualified cardinality restriction, transitive closure etc. can also be further investigated for extracting implicit and explicit information in line with ECOS-Query.

3 **Inclusion of Semantic web rule language (SWRL) in ontology query, mapping and merging.** SWRL provides ontology to express complex interrelation between three and more variables as compare to two variables (binary) in OWL. ECOS-Query, ontology mapping and matching can further be extended to accommodate SWRL.

4 **Complexity of ontology query, mapping and merging in the presence of different DL-constructor.** Complexity of the ontology query, mapping and merging depends on the ontology constructor. One of the key area of future extension of this research is to understand, explore and resolve the complexity issues related to inclusion of different ontology constructors.

5 **Game theoretic based cost and profit distribution with uncertainty in cost and profit.** Game theoretic profit distribution approach proposed in this research can also be extended in the uncertain environment i.e. with uncertain cost and profit scenario.
REFERENCES


185


Massmann, S., Engmann, D., Rahm, E., 2006. COMA++: Results for the Ontology Alignment Contest OAEI 2006, in: International Workshop on Ontology Matching, Collocated with the 5th ISWC-2006. Athens, Georgia, USA.


Pan, J.Z., 2004. DESCRIPTION LOGICS: REASONING SUPPORT FOR THE SEMANTIC WEB.


W3C [WWW Document], n.d. . URL http://www.w3.org/


Wordnet [WWW Document], n.d. URL http://wordnet.princeton.edu/


APPENDIX (ECOS DB)

```xml
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://kmm.lboro.ac.uk/ecos/#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:swrl="http://www.w3.org/2003/11/swrl#"
  xmlns:swrlb="http://www.w3.org/2003/11/swrlb#
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://kmm.lboro.ac.uk/ecos/#">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Customers"/>
  <owl:Class rdf:ID="Location"/>
  <owl:Class rdf:ID="Human_Resource">
    <rdfs:subClassOf rdf:ID="Capacity"></rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Preferences"/>
  <owl:Class rdf:ID="Broker">
    <rdfs:subClassOf rdf:ID="Company"></rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Experiences"/>
  <owl:Class rdf:ID="Comptencies"/>
  <owl:Class rdf:ID="Finance"/>
  <owl:Class rdf:ID="Skill"/>
  <owl:Class rdf:ID="Production_Capacity">
    <rdfs:subClassOf rdf:resource="#Capacity"/>
  </owl:Class>
  <owl:Class rdf:ID="Competetors"/>
  <owl:Class rdf:ID="Product"/>
  <owl:ObjectProperty rdf:ID="hasExperiences">
    <rdfs:domain rdf:resource="#Company"/>
    <rdfs:range rdf:resource="#Experiences"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasCompetetors">
    <rdfs:domain rdf:resource="#Company"/>
    <rdfs:range rdf:resource="#Competetors"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasHumanResources">
    <rdfs:domain rdf:resource="#Company"/>
    <rdfs:range rdf:resource="#Human_Resource"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasCustomers">
    <rdfs:domain rdf:resource="#Company"/>
    <rdfs:range rdf:resource="#Customers"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasSkill">
    <rdfs:domain rdf:resource="#Company"/>
    <rdfs:range rdf:resource="#Skill"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasProduct">
    <rdfs:domain rdf:resource="#Company"/>
    <rdfs:range rdf:resource="#Product"/>
  </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:ID="hasCapacity">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
    <rdfs:comment xml:lang="en"/>
    <rdfs:domain rdf:resource="#Company"/>
  </owl:DatatypeProperty>
  <Company rdf:ID="BCRA_Company">
    <hasCapacity rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1000</hasCapacity>
    <hasCustomers rdf:ID="UK"></hasCustomers>
    <hasLocation rdf:ID="Midlands"></hasLocation>
    <hasFinance rdf:ID="Medium"></hasFinance>
    <hasExperiences rdf:ID="UK_Partnership"></hasExperiences>
    <hasCompetencies rdf:ID="Moulding"></hasCompetencies>
    <hasSkill rdf:ID="Plastic"></hasSkill>
    <hasHumanResources rdf:ID="Medium_Skill"></hasHumanResources>
    <hasProduct rdf:ID="Rubber"></hasProduct>
    <Company rdf:ID="Pace_Groups">
      <hasFinance rdf:resource="#Medium"/>
    </Company>
  </Company>
</rdf:RDF>
```
<hasProduct rdf:resource="#CNC_Machining_Products"/>
<hasCompetencies rdf:resource="#Engineering_and_Manufacturing"/>
<hasProduct rdf:resource="#Car_Parts"/>
</Company>
<Company rdf:ID="Hatch-Mott_Engg">
<hasProduct rdf:resource="#Logistics"/>
<hasExperiences rdf:resource="#UK_Partnership"/>
<hasCustomers rdf:resource="#Europe"/>
<hasLocation rdf:resource="#London"/>
<hasFinance rdf:resource="#Low"/>
<hasHumanResources rdf:resource="#Low_Skill"/>
<hasCompetencies rdf:resource="#Customer_Care"/>
</Company>
</Broker>
<Company rdf:ID="UK_Consortium">
<hasCompetencies rdf:resource="#Project_Management"/>
<hasLocation rdf:resource="#Scotland"/>
<hasFinance rdf:resource="#Medium"/>
<hasExperiences rdf:resource="#EU_Partnership"/>
<hasCustomers rdf:resource="#Europe"/>
<hasHumanResources rdf:resource="#Medium_Skill"/>
</Broker>
</Company>
<Company rdf:ID="Midlands_IT_Networks">
<hasProduct rdf:resource="#IT_Security"/>
<hasSkill rdf:resource="#Wireless_Solutions"/>
<hasHumanResources rdf:resource="#High_Skill"/>
<hasCompetencies rdf:resource="#ICT_Support"/>
<hasFinance rdf:resource="#Excellent"/>
<hasCustomers rdf:resource="#Europe"/>
<hasSkill rdf:resource="#Mobile"/>
<hasLocation rdf:resource="#London"/>
<hasCompetencies rdf:resource="#Software_Development"/>
<hasExperiences rdf:resource="#EU_Partnership"/>
<hasSkill rdf:resource="#Software"/>
</Company>
</Broker>
<Company rdf:ID="Europa_Group">
<hasFinance rdf:resource="#High"/>
<hasHumanResources rdf:resource="#Medium_Skill"/>
<hasExperiences rdf:resource="#EU_Partnership"/>
<hasProduct rdf:resource="#Terminals"/>
<hasCompetencies rdf:resource="#Customer_Care"/>
<hasHumanResources rdf:resource="#Low_Skill"/>
<hasLocation rdf:resource="#London"/>
</Company>
<Company rdf:ID="Western_Corp.">
<hasCompetencies rdf:resource="#Marketing"/>
<hasFinance rdf:resource="#High"/>
<hasHumanResources rdf:resource="#High"/>
<hasExperiences rdf:resource="#EU_Partnership"/>
<hasProduct rdf:resource="#Outlet"/>
<hasCustomers rdf:resource="#UK"/>
</Company>
<Company rdf:ID="Hart_Crowser_Networks">
<hasSkill rdf:resource="#Metal"/>
<hasExperiences rdf:resource="#UK_Partnership"/>
<hasProduct rdf:resource="#Terminals"/>
<hasCustomers rdf:resource="#Europe"/>
<hasCompetencies rdf:resource="#Supplier"/>
<hasFinance rdf:resource="#Low"/>
</Company>
<Company rdf:ID="Munson_Forging">
<hasProduct rdf:resource="#Car_Parts"/>
<hasCompetencies rdf:resource="#Manufacturing"/>
<hasProduct rdf:resource="#UK_Partnership"/>
<hasCapacity rdf:resource="http://www.w3.org/2001/XMLSchema#int">750</hasCapacity>
<hasFinance rdf:resource="#Low"/>
<hasHumanResources rdf:resource="#Low_Skill"/>
<hasCustomers rdf:resource="#UK"/>
<hasCompetencies rdf:resource="#Supplier"/>
<hasProduct rdf:resource="#Logistics"/>
</Company>
<Company rdf:ID="Triad_Services">
<hasLocation rdf:resource="#London"/>
<hasHumanResources rdf:resource="#Medium_Skill"/>
<hasFinance rdf:resource="#High"/>
<hasCustomers rdf:resource="#Europe"/>
<hasCompetencies rdf:resource="#Customer_Care"/>
<hasExperiences rdf:resource="#None"/>
<hasProduct rdf:resource="#Logistics"/>
<hasFinance rdf:resource="#Low"/>
</Company>
<Company rdf:ID="Pierce_County">
<hasCompetencies rdf:resource="#Customer_Care"/>
<hasHumanResources rdf:resource="#Low_Skill"/>
<hasLocation rdf:resource="#London"/>
<hasCustomers rdf:resource="#UK"/>
<hasExperiences rdf:resource="#None"/>
</Company>
<Company rdf:ID="Pulp_Inc">
<hasLocation rdf:resource="#London"/>
</Company>
<hasExperiences rdf:resource="#UK_Partnership"/> <hasHumanResources rdf:resource="#Medium_Skill"/> <hasLocation rdf:resource="#London"/> </Company> <Company rdf:ID="#ICI_Groups"> <hasFinance rdf:resource="#Medium"/> <hasCustomers rdf:resource="#Europe"/> <hasExperiences rdf:resource="#UK_Partnership"/> <hasLocation rdf:resource="#Midlands"/> <hasCompetencies rdf:resource="#Supplier"/> <hasProduct rdf:resource="#Terminals"/> <hasHumanResources rdf:resource="#Medium_Skill"/> <hasSkill rdf:resource="#Steel"/> </Company> <Product rdf:ID="#Warehousing"/> <Company rdf:ID="#TransWorks_Inc"> <hasCapacity rdf:datatype="http://www.w3.org/2001/XMLSchema#int">200</hasCapacity> <hasHumanResources rdf:resource="#Medium_Skill"/> <hasFinance rdf:resource="#Medium"/> <hasLocation rdf:resource="#Midlands"/> <hasProduct rdf:resource="#Car_Parts"/> <hasCustomers rdf:resource="#UK"/> <hasSkill rdf:resource="#Metal"/> <hasCompetencies rdf:resource="#Manufacturing"/> </Company> </Company> <Company rdf:ID="#YTech_UK"> <hasHumanResources rdf:resource="#Low_Skill"/> <hasProduct rdf:resource="#Testing"/> <hasCompetencies rdf:resource="#Software_Development"/> <hasSkill rdf:resource="#Software"/> <hasLocation rdf:resource="#London"/> <hasFinance rdf:resource="#Medium"/> <hasExperiences rdf:resource="#None"/> <hasCustomers rdf:resource="#UK"/> </Company> <Company rdf:ID="#Wallis_Heat_Treatment"> <hasHumanResources rdf:resource="#Low_Skill"/> <hasProduct rdf:resource="#Testing"/> <hasCompetencies rdf:resource="#Software_Development"/> <hasSkill rdf:resource="#Software"/> <hasLocation rdf:resource="#London"/> <hasFinance rdf:resource="#Medium"/> <hasExperiences rdf:resource="#None"/> <hasCustomers rdf:resource="#UK"/> </Company> <Company rdf:ID="#Kappa_SSK_Ltd"> <hasLocation rdf:resource="#EU"/> <hasProduct rdf:resource="#Logistics"/> <hasFinance rdf:resource="#Medium"/> <hasCustomers rdf:resource="#Europe"/> <hasExperiences rdf:resource="#EU_Partnership"/> <hasHumanResources rdf:resource="#Medium_Skill"/> <hasCompetencies rdf:resource="#Customer_Care"/> <hasCustomers rdf:resource="#Europe"/> <hasCapacity rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1000</hasCapacity> </Company> <Company rdf:ID="#Cary_Cam_Engg"> <hasCustomers rdf:resource="#Europe"/> <hasCapacity rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1000</hasCapacity> </Company> <Company rdf:ID="#ENSR_Plastics"> <hasExperiences rdf:resource="#None"/> <hasSkill rdf:resource="#Plastic"/> <hasProduct rdf:resource="#Cables"/> <hasHumanResources rdf:resource="#Low_Skill"/> <hasLocation rdf:resource="#Scotland"/> <hasCompetencies rdf:resource="#Moulding"/> <hasCapacity rdf:datatype="http://www.w3.org/2001/XMLSchema#int">250</hasCapacity> </Company> <Company rdf:ID="#WSDOT"> <hasLocation rdf:resource="#Scotland"/> <hasCompetencies rdf:resource="#Supplier"/> <hasSkill rdf:resource="#Electronics"/> <hasCustomers rdf:resource="#UK"/> <hasProduct rdf:resource="#Terminals"/> <hasCompetencies rdf:resource="#Supplier"/> <hasSkill rdf:resource="#Electrical"/> </Company> <Company rdf:ID="#Anchor_Ltd"> <hasCustomers rdf:resource="#Europe"/> <hasFinance rdf:resource="#Excellent"/> </Company> <Company rdf:ID="#Kappa_SSK_Ltd"> <hasLocation rdf:resource="#EU"/> <hasProduct rdf:resource="#Logistics"/> <hasCompetencies rdf:resource="#Supplier"/> <hasSkill rdf:resource="#Electrical"/> </Company> <Company rdf:ID="#Anchor_Ltd"> <hasCustomers rdf:resource="#UK"/> <hasProduct rdf:resource="#Terminals"/> <hasCompetencies rdf:resource="#Supplier"/> <hasSkill rdf:resource="#Electrical"/> </Company> <Company rdf:ID="#Anchor_Ltd"> <hasCustomers rdf:resource="#UK"/> <hasProduct rdf:resource="#Terminals"/> <hasCompetencies rdf:resource="#Supplier"/> <hasSkill rdf:resource="#Electrical"/> </Company> <Company rdf:ID="#Anchor_Ltd"> <hasCustomers rdf:resource="#UK"/> <hasProduct rdf:resource="#Terminals"/> <hasCompetencies rdf:resource="#Supplier"/> <hasSkill rdf:resource="#Electrical"/> </Company>
<hasCompetencies rdf:resource="#Manufacturing"/>
<hasExperiences rdf:resource="#EU_Partnership"/>
<hasLocation rdf:resource="#Wales"/>
<hasProduct rdf:ID="Engine"/>
</hasProduct>
<hasHumanResources rdf:resource="#High_Skill"/>
<hasSkill rdf:resource="#Automobile"/>
<hasSkill rdf:ID="Aerospace"/>
</Company>
<Company rdf:ID="KPFF_manufacturers">
<hasCapacity rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1750</hasCapacity>
<hasHumanResources rdf:resource="#Medium_Skill"/>
<hasCompetencies rdf:resource="#Manufacturing"/>
<hasExperiences rdf:resource="#UK_Partnership"/>
<hasProduct rdf:resource="#Electronic_Components"/>
<hasLocation rdf:resource="#Midlands"/>
<hasFinance rdf:resource="#Medium"/>
<hasSkill rdf:resource="#Electrical"/>
<hasCustomers rdf:resource="#UK"/>
</Company>
<Company rdf:ID="GoodFellows_Inc">
<hasCompetencies rdf:resource="#Assembly"/>
<hasHumanResources rdf:resource="#Medium_Skill"/>
<hasLocation rdf:resource="#Scotland"/>
<hasCustomers rdf:resource="#UK"/>
<hasProduct rdf:resource="#Terminals"/>
<hasFinance rdf:resource="#Low"/>
<hasCapacity rdf:datatype="http://www.w3.org/2001/XMLSchema#int">800</hasCapacity>
<hasSkill rdf:resource="#Hardware"/>
<hasExperiences rdf:resource="#UK_Partnership"/>
</Company>
<Company rdf:ID="J_and_H_Sales_Ltd">
<hasSkill rdf:resource="#Metal"/>
<hasFinance rdf:resource="#Medium"/>
<hasLocation rdf:resource="#London"/>
<hasCompetencies rdf:resource="#Supplier"/>
<hasCustomers rdf:resource="#Europe"/>
<hasProduct rdf:resource="#Terminals"/>
<hasHumanResources rdf:resource="#Medium_Skill"/>
<hasExperiences rdf:resource="#EU_Partnership"/>
<hasSkill rdf:resource="#Steel"/>
</Company>
<Company rdf:ID="PLD_Groups">
<hasHumanResources rdf:resource="#Low_Skill"/>
</Company>
<Company rdf:ID="Seatle_National_Groups">
<hasExperiences rdf:resource="#International_Partnership"/>
<hasCompetencies rdf:resource="#Automation_and_Control"/>
<hasHumanResources rdf:resource="#High_Skill"/>
<hasProduct rdf:resource="#Testing"/>
<hasSkill rdf:resource="#Electronics"/>
<hasLocation rdf:resource="#USA"/>
<hasLocation rdf:resource="#Wales"/>
<hasFinance rdf:resource="#Medium"/>
<hasCustomers rdf:resource="#US"/>
</Company>
<Company rdf:ID="Perteet_Associates">
<hasCompetencies rdf:resource="#Automation_and_Control"/>
<hasCustomers rdf:resource="#Europe"/>
<hasSkill rdf:resource="#Electronics"/>
<hasLocation rdf:resource="#Wales"/>
<hasFinance rdf:resource="#Medium"/>
<hasExperiences rdf:resource="#EU_Partnership"/>
<hasHumanResources rdf:resource="#High_Skill"/>
<hasProduct rdf:resource="#Terminals"/>
</Company>
The following papers have been published or communicated for publication related to this research.

1. **Ontology mapping using description logic and bridging axioms**
   
   Sri Krishna Kumar, J. A. Harding
   
   (Published in Computers in Industry, Volume 64, Issue 1, 2013, Pages 19–28)
   
   
   **Abstract:** In the last decade various proposals have been made to promote fruitful and efficient collaboration among small and medium sized enterprises (SMEs) in the form of virtual enterprises (VEs). The success of VEs depends on seamless interoperability of knowledge and data sharing. Ontology implementation is becoming an essential and successful tool for VE operation but commonly ontology mapping is also required to achieve interoperability. The current state of the art in ontology mapping indicates that mapping systems require a great deal of human intervention as mapping brings various types of conflicts and inconsistencies. The ontology mapping method proposed in this paper uses description logic (DL) based bridging axioms between the ontologies. Atomic concept level similarity has been taken as input to establish the complex concepts and roles level mapping. Manufacturing and marketing enterprise ontologies are considered and their mapping has been demonstrated as an example of the proposed mapping process.

2. **Risk Assessment in the Formation of Virtual Enterprises**
   
   Sri Krishna Kumar, J. A. Harding
   
   (Published in Adaptation and Value Creating Collaborative Networks IFIP Advances in Information and Communication Technology Volume 362, 2011, pp 450-455)
A Virtual Enterprise (VE) is considered as a temporary consortium of member enterprises formed to pool their core competencies and exploit the market opportunities. Although a VE has many phases, such as business opportunity identification, formation and partner selection, operation and dissolution. The partner selection phase is considered to be of the utmost importance and care should be taken to assess all the risk factors. This paper examines the partner selection problem by considering three types of risks, individual performance risk, collaborative performance risk and network risk. Based on the information provided by the potentially collaborating enterprises, a mathematical model has been developed for calculation of all three types of risks.


Rahul Swarnkar, Jennifer A. Harding, Bishnu P. Das, Robert I. Young, S. Krishna Kumar


Abstract:- Knowledge intensive industries benefit immensely from collaborative projects in virtual organisations. However the complexities of the business processes increase with the interdependencies. Successful operations of virtual organisations depends heavily on knowledge sharing among the partners as this is essential for improving the quality of decisions taken by the participating members. Collaboration moderators are specialist applications designed to address issues related to knowledge sharing and to provide functionality to raise awareness. This paper describes such a collaboration moderator service to aid collaborative drug discovery in a pharmaceutical virtual organization.
4. Querying ECOS Ontology: A Description Logic based approach

Sri Krishna Kumar, J. A. Harding

(Communicated in Journal of Web Semantics)

Abstract:- The ECOS ontology (Khilwani et. al. 2011) was developed for enterprises for publishing their information with both human and machine readable (through web) properties. Enterprises seeking for collaboration in the form of virtual enterprises (VE) can use this standard ontology for information exchange. However, formation of VEs requires various functionalities from information support including the ability to search for enterprises with required characteristics such as competency, capacity, resources etc. Most often searching an ECOS database, according to the VE requirements, involves extracting implicit information and knowledge. This paper addresses these issues and proposes a method for Description Logic based queries for extracting implicit information and knowledge which has not explicitly been defined in the concepts and roles in the ontology and data type. The proposed method starts with an ECOS-query rearrangement according to the variables and then moves to DL based optimization of the query i.e. deleting the unnecessary parts of a query without affecting the overall results. Finally the query processing delivers all the sets of possible results.

5. Partner Selection in VE based on Risk Analysis and Revenue Sharing: a Game Theoretic Approach

Sri Krishna Kumar, J. A. Harding

(Communicated in European Journal of Operations Research)

Abstract:- Partner selection is an important component in the formation of virtual enterprises (VEs). Finding the right partners in the formation of VE is a difficult task due to dynamic environment, risk factors and decentralized control. Traditional methods are insufficient in analysing direct and indirect contribution of enterprises in the VE. This paper develops a revenue sharing mechanism in the partner selection process. Different risk factors are considered
in evaluating direct and indirect contribution of the potential partners. Cooperative game theory is used to decide optimal revenue shares of individual enterprises based on their risks and contributions. The problem has been formulated as NLIP (non-linear integer programming) and using its duality divided in two stages. First stage finds the optimal partners by solving NLIP using series of relaxation and second stage divides the revenue using the linear programming. It has been shown that at stage problems can have infeasible, multiple or unique solutions depending on the expected profit and their contribution. Furthermore, sensitivity analysis is carried out to find the robustness of formed VE with the change in risk environment.

6. **Description Logic based Knowledge Merging for Concrete and Fuzzy Domain Ontology**

   Sri Krishna Kumar, J. A. Harding

   (Communicated in IMechE: Part B: Journal of Engineering Manufacture)

   **Abstract:** Enterprises, especially virtual enterprises (VEs) are nowadays getting more knowledge intensive and adopting efficient Knowledge management (KM) systems to boost their competitiveness. The major challenge for KM for VEs is to acquire, extract and integrate new knowledge with the existing source. Ontologies have been proved to be one of the best tools for representing knowledge with class, role and other characteristics. It is imperative to accommodate the new knowledge in the current ontologies with logical consistencies as it is tedious and costly to construct new ontologies every time. This paper introduces a mechanism and a process to integrate new knowledge in to the current system (ontology). Separate methods have been adopted for fuzzy and concrete domain ontologies. The process starts by finding the semantic and structural similarities between the concepts using Wordnet and Description logic (DL). DL-based reasoning is used next to determine the position and relationships between the incoming and existing knowledge. The experimental results provided show the efficacy of the proposed Method.