AGENT-ORIENTED ONTOLOGY PATTERN CLASSIFICATION FOR ONTOLOGY-BASED MULTI-AGENT SYSTEM DEVELOPMENT

Fu Swee Tee

Master of Advanced Information Technology
SOFTWARE ENGINEERING
2012
Abstract

Multi-agent system have received much attention nowadays in various domains to solve complex problems within the open or closed environment. Ontology plays a crucial role within the multi-agent system in which it represents the information domain of the agent system. Furthermore, ontology is needed to govern agent interaction and agent behaviour. Work has been done to integrate ontology engineering approach within the MaSE methodology to include the use of ontologies for information domain specification. However, efficient developments of ontology are still a challenge as it requires extensive knowledge and experience for the underlying domain of interest. Therefore, with the issues faced above, there are still insufficient works to support the developments of the ontology-based multi-agent system systematically.

This research employs a reusable approach to include the use of agent-oriented ontology patterns during the development of an ontology-based multi-agent system. An investigation into agent-oriented ontology patterns that can be used within the multi-agent system shall be explored in this thesis. Agent-oriented ontology pattern refers to ontology pattern that records the experience of implementing ontology that is to be used within the agent domain. These informal agent-oriented ontology patterns are able to reduce the need for extensive experience during ontology engineering within the multi-agent system development. Here, we are focused on capturing the experiences in the use of ontology to develop the agent domain, the agent interoperability and the agent reasoning.

Consequently, we present a pattern classification scheme to ease the adoption of agent-oriented ontology patterns collected during the development of the ontology-based multi-agent system. The pattern classification scheme supports the classification of the agent-oriented
ontology patterns into the appropriate dimensions which improves the comprehensiveness of the ontology pattern. Alternatively, it also supports the selection of the right ontology patterns to match to the problem at hand during multi-agent system development.

Finally, we shall demonstrate the adoption of the informal agent-oriented ontology patterns within the proposed pattern classification for a case study of an agent-based travel support system. Appropriate pattern selection guidelines to aid the reuse of agent-oriented ontology patterns within the ROADMAP/AOR methodology are presented. The result for the demonstration of the case study will show the incorporation of agent-oriented ontology patterns have introduced a systematic way to the development of an ontology-based multi-agent system.
Declaration

This is to certify that:

i. The thesis comprises only my original work towards the Master.

ii. Due acknowledgement has been made in the text to all other material used.

iii. This thesis is less than 100,000 words in length exclusive of tables, figures, bibliographies, and appendices.

Fu Swee Tee
I would like to express my sincere gratitude to my supervisor, Dr. Cheah Wai Shiang for his guidance and support throughout the course of my master research effort. I would also like to express my appreciation to my employer, International College of Advanced Technology Sarawak (ICATS) for their support in this endeavor.

Finally, I would like to thank my family for their love, understanding and motivation since I first enrolled in this Master course.
# Table of Contents

**Chapter 1: Introduction**
1.1 Research Problem 5
1.2 Motivation 5
1.3 Aims 6
1.4 Scope 7
1.5 Thesis Contributions 8
1.6 Research Methodology 10
1.7 Thesis Overview 13

**Chapter 2: Literature Review**
2.1 Ontology for Multi-Agent System 16
2.2 Patterns for Multi-Agent System 20
2.3 Ontology Patterns 24
2.4 Two-Ways Classification Scheme 27
2.5 Conclusion 31

**Chapter 3: Classification Scheme for Agent-Oriented Ontology Pattern**
3.1 Agent-Oriented Ontology Patterns 34
3.2 Dimensions for Classification Scheme 35
3.2.1 Software Process Dimension of the Classification Scheme 36
3.2.1.1 Conceptual Information Modelling Level 37
3.2.1.2 Platform Independent Computational Design Level 38
3.2.1.3 Platform Specific Design and Implementation Level 39
3.2.2 Aspect Dimension of the Classification Scheme 40
3.2.2.1 Conceptual Information Modelling Level 41
3.2.2.2 Platform Independent Computational Design Level 45
3.2.2.3 Platform Specific Design and Implementation Level 47
3.3 Attributes for Attribute-Based Analysis 49
3.3.1 Two-Ways Classification Scheme Level Attributes 51
3.3.2 Two-Ways Classification Scheme Category Attributes 55
3.4 Conclusion 62

**Chapter 4: Pattern by Pattern Analysis and Classification**
4.1 Attribute-Based Analysis: Concept and Tools 65
4.2 Attribute-Based Analysis: The Process 68
4.3 Agent-Oriented Ontology Patterns Analysis and Classification 69
4.3.1 Analysis and Classification at Conceptual Information Modelling Level 70
4.3.1.1 Generic Scope and Domain 71
4.3.1.2 Generic Use Cases – Interaction 72
4.3.1.3 Action 74
4.3.2 Analysis and Classification at Platform Independent Computational Design Level 75
4.3.2.1 Legal Domain 76
4.3.2.2 Argumentation Protocol 78
4.3.2.3 Explanation Strategy 80
4.3.3 Analysis and Classification at Platform Specific Design and Implementation Level 81
4.3.3.1 Upper-Level Travel Ontology 82
4.3.3.2 Message Content Ontology 85
4.3.3.3 Scheduling Task Ontology 87
4.4 Validation of Pattern Classification Scheme for Agent-Oriented Ontology Patterns 89
4.4.1 Collections of Agent-Oriented Ontology Patterns 89
4.4.2 Result of the Analysis of Agent-Oriented Ontology Patterns 92
4.5 Observation on Analysis and Classification of Agent-Oriented Ontology Patterns 104
Chapter 5: Integrating Agent-Oriented Ontology Patterns for Agent-Based Travel Support System Development

5.1 Introduction to Agent-Based Travel Support System
5.2 ROADMAP and AOR Methodology for Agent Development
5.3 Agent-Oriented Ontology Pattern Selection
5.4 Scenario 1: Finding the Restaurant Information through Agent-Based Travel Support System
5.5 Scenario 2: Adding the New Restaurant Details through Agent-Based Travel Support System
5.6 Lesson Learned
5.7 Conclusion

Chapter 6: Summary and Conclusion

6.1 Thesis Summary
6.2 Future Research Areas

Appendix A: Knowledge Source of Agent-Oriented Ontology Patterns

Appendix B: Mapping HOMER Questions to ROADMAP/AOR Modelling

Appendix C: Description on the PSM-level for Scenario 2 – Adding the New Restaurant Details through Agent-Based Travel Support System

Bibliography
List of Figures

Figure 1.1  Thesis Layout 15
Figure 2.1  Extended MaSE Phases 19
Figure 4.1  Attribute-based Analysis Process 68
Figure 4.2  Generic Scope and Domain for Ontology Requirement Specification 72
Figure 4.3  Competency Questions for Generic Use Cases - Interaction Ontology 73
Figure 4.4  Competency Questions for Action Ontology Pattern 75
Figure 4.5  Concept Taxonomy of Legal Entity Domain 77
Figure 4.6  Strategy used by the Explainer 80
Figure 4.7  Strategy used by the Initiator 81
Figure 4.8  Code snippet of travel ontology 83
Figure 4.9  An instance of hotel 84
Figure 4.10  Content Reference Model 86
Figure 4.11  Java Object Classes of Concept Person showing members and set and get methods 86
Figure 4.12  Goal expressions for scheduling task ontology 87
Figure 4.13  The two axioms for preferences and cost function 88
Figure 5.1  The Combined Modelling Process for ROADMAP and AOR 111
Figure 5.2  Overall Framework for the Incorporation of Agent-Oriented Ontology Pattern into the Development of Multi-Agent System 113
Figure 5.3  Overall Goal Model for Scenario 1 of the Agent-Based Travel Support System 121
Figure 5.4  Organizational Model for Scenario 1 of the Agent-Based Travel Support System 122
Figure 5.5  Domain Model for Scenario 1 of the Agent-Based Travel Support System 123
Figure 5.6  Knowledge Model for Scenario 1 of the Agent-Based Travel Support System 125
Figure 5.7  P83 General Structure of Hotel Ontology 126
Figure 5.8  P84 Restaurant Domain Ontology for representing User Profile 126
Figure 5.9  P75 CollectionEntity Ontology Design Pattern 127
Figure 5.10  Restaurant Ontology for Scenario 1 of the Agent-Based Travel Support System 128
Figure 5.11  Interpretation and Clarification in the Ontology Negotiation Protocol 129
Figure 5.12  Evaluation and Justification of Query Result in the Ontology Negotiation Protocol 129
Figure 5.13  Interaction Model for Successful Execution of Scenario 1: Finding the Restaurant Information 131
Figure 5.14  Interaction Model for Scenario 1: Finding the Restaurant Information with the Incorporation of PIM-Interaction Agent-Ontology Patterns 132
Figure 5.15  Scenario Model for Scenario 1 of the Agent-Based Travel Support System 135
Figure 5.16  Behaviour Model for Scenario 1 of the Agent-Based Travel Support System 137
Figure 5.17  JADE Specific Information Model for Information Type of Restaurant 139
Figure 5.18  JADE Specific Information Model for Action Type of FindRestaurant 140
Figure 5.19  JADE Specific Interaction Model for RestaurantFinder 141
Figure 5.20  JADE Specific Behaviour Model for the goal Find Potential Restaurant 144
Figure 5.21  Screenshot for the Main Page of Scenario 1 145
Figure 5.22  Screenshot for the Search-By-Keyword Page of Scenario 1 146
Figure 5.23  Screenshot for the Potential Restaurant Lists returned using the Search-By-Keyword 146
Figure 5.24  Screenshot for the Advance Search Page of Scenario 1 147
Figure 5.25  Screenshot for the Restaurant Listing returned using the Advance Search Function 147
Figure 5.26  Overall Goal Model for Scenario 2 of the Agent-Based Travel Support System 151
Figure 5.27  Organizational Model for Scenario 2 of the Agent-Based Travel Support System 152
Figure 5.28  Domain Model for Scenario 2 of the Agent-Based Travel Support System 152
Figure 5.29  Knowledge Model for Scenario 2 of our Agent-Based Travel Support System 154
Figure 5.30  Restaurant Ontology for the Agent-Based Travel Support System 155
Figure 5.31  P76 FIPA Request Interaction Protocol 156
Figure 5.32  Interaction Model for Successful Adding of New Restaurant in the Agent-Based Travel Support System 158
Figure 5.33  Interaction Model for Unsuccessful Adding of New Restaurant in the Agent-Based Travel Support System  159
Figure 5.34  Scenario Model for Scenario 2 of the Agent-Based Travel Support System  160
Figure 5.35  Behaviour Model for Scenario 2 of the Agent-Based Travel Support System  161
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 2.1</td>
<td>Validation Results of Improved Two-Ways Classification Scheme</td>
<td>24</td>
</tr>
<tr>
<td>Table 2.2</td>
<td>Two-Way Classification Scheme for Pattern Classification</td>
<td>27</td>
</tr>
<tr>
<td>Table 2.3</td>
<td>Attributes for Improved Two-Way Classification Scheme</td>
<td>30</td>
</tr>
<tr>
<td>Table 3.1</td>
<td>Sample Collections of Agent-Oriented Ontology Patterns</td>
<td>35</td>
</tr>
<tr>
<td>Table 3.2</td>
<td>Pattern Classification for Agent-Oriented Ontology Patterns</td>
<td>35</td>
</tr>
<tr>
<td>Table 3.3</td>
<td>Software Process Dimension Level</td>
<td>37</td>
</tr>
<tr>
<td>Table 3.4</td>
<td>Aspect Dimension at Conceptual Domain Modelling Level</td>
<td>41</td>
</tr>
<tr>
<td>Table 3.5</td>
<td>Aspect Dimension at Platform Independent Computational Design Level</td>
<td>45</td>
</tr>
<tr>
<td>Table 3.6</td>
<td>Aspect Dimension at Platform Specific Design and Implementation Level</td>
<td>47</td>
</tr>
<tr>
<td>Table 3.7</td>
<td>Attributes for Pattern Classification</td>
<td>51</td>
</tr>
<tr>
<td>Table 4.1</td>
<td>Identification of software process dimension attributes for agent-oriented ontology patterns</td>
<td>67</td>
</tr>
<tr>
<td>Table 4.2</td>
<td>Identification of category attributes for conceptual domain modelling level</td>
<td>67</td>
</tr>
<tr>
<td>Table 4.3</td>
<td>Identification of category attributes for platform independent computational design level</td>
<td>67</td>
</tr>
<tr>
<td>Table 4.4</td>
<td>Identification of category attributes for platform specific design and implementation level</td>
<td>67</td>
</tr>
<tr>
<td>Table 4.5</td>
<td>Identification of category attributes for conceptual domain modelling level</td>
<td>79</td>
</tr>
<tr>
<td>Table 4.6</td>
<td>Complete Listing of Agent-Oriented Ontology Patterns</td>
<td>92</td>
</tr>
<tr>
<td>Table 4.7</td>
<td>Identification of software process dimension attributes for agent-oriented ontology patterns</td>
<td>93</td>
</tr>
<tr>
<td>Table 4.8</td>
<td>Identification of category attributes for conceptual domain modelling level</td>
<td>97</td>
</tr>
<tr>
<td>Table 4.9</td>
<td>Identification of category attributes of agent-oriented ontology patterns for platform independent computational design level</td>
<td>98</td>
</tr>
<tr>
<td>Table 4.10</td>
<td>Identification of category attributes of agent-oriented ontology patterns for specific design and implementation level</td>
<td>101</td>
</tr>
<tr>
<td>Table 4.11</td>
<td>A Comprehensive View of Agent-Oriented Ontology Pattern</td>
<td>104</td>
</tr>
<tr>
<td>Table 5.1</td>
<td>CIM-Level Agent-Oriented Ontology Patterns incorporated in Scenario 1 of our case study</td>
<td>116</td>
</tr>
<tr>
<td>Table 5.2</td>
<td>Elicitation Questions of HOMER</td>
<td>117</td>
</tr>
<tr>
<td>Table 5.3</td>
<td>Incorporating CIM-level Agent-Oriented Ontology Patterns into HOMER</td>
<td>118</td>
</tr>
<tr>
<td>Table 5.4</td>
<td>Answers to the Extended HOMER Elicitation Questions for First Scenario of our Agent-Based Travel Support System</td>
<td>121</td>
</tr>
<tr>
<td>Table 5.5</td>
<td>Role Model for Traveller</td>
<td>122</td>
</tr>
<tr>
<td>Table 5.6</td>
<td>Role Model for Restaurant Finder</td>
<td>122</td>
</tr>
<tr>
<td>Table 5.7</td>
<td>PIM-Level Agent-Oriented Ontology Patterns incorporated in Scenario 1 of our case study</td>
<td>123</td>
</tr>
<tr>
<td>Table 5.8</td>
<td>CIM-Level Agent-Oriented Ontology Patterns incorporated in Scenario 2 of our case study</td>
<td>148</td>
</tr>
<tr>
<td>Table 5.9</td>
<td>Answers to the Extended HOMER Elicitation Questions for Scenario 2 of our Agent Based Travel Support System</td>
<td>150</td>
</tr>
<tr>
<td>Table 5.10</td>
<td>Role Model for Restaurant Adder</td>
<td>151</td>
</tr>
<tr>
<td>Table 5.11</td>
<td>Role Model for Restaurant Seller</td>
<td>151</td>
</tr>
<tr>
<td>Table 5.12</td>
<td>PIM-Level Agent-Oriented Ontology Patterns incorporated in Scenario 2 of our case study</td>
<td>153</td>
</tr>
<tr>
<td>Table 5.13</td>
<td>Evaluation result upon completion of Scenario 1 of the case study by the evaluator</td>
<td>164</td>
</tr>
</tbody>
</table>
Chapter 1: Introduction

The emergences of Semantic Web, Linked Data and the introduction of semantic technologies in different areas of application have attracted various attentions to look into ontologies nowadays. Ontologies had been adopted in a range of areas such as tourism (Vukmirovic, M. et al. 2006), manufacturing (Mingwei, W. et al. 2010) (Georgoudakis, M. et al. 2012), e-commerce (Ruixue, F. et al. 2008), biological (Jiann-Jyh, L. et al. 2003), knowledge management (Choiński, D. et al. 2011) (Béhé, F. et al. 2012) and agricultural (Rajakaruna, G. M. et al. 2012). The ontology has established to be a powerful tool to enable knowledge sharing and also in achieving the semantic interoperability among heterogeneous distributed systems (Dolia, P. M. 2010).

Alternatively, ontology forms an integral part of a multi-agent system which it specifies the information domain of the system and the individual agents (DiLeo, J. et al. 2002). A multi-agent system, is a system made up of multiple intelligent agents working together to find solution to problems that are beyond the individual capabilities or knowledge of each entity (Dolia, P. M. 2010). Multi-agent systems have received much attention nowadays, as it provides a framework for developing complex application in a distributed environment. In addition, multi-agent system is used to cater for situation which will change dynamically and require highly heterogeneous agents to cope up with the changes over time. Within the multi-agent system environment, ontology plays the role of supporting agent communication and agent reasoning. The agents operate and cooperate in the manner where messages are passed/exchanged between the agent themselves and such exchanges must conform to some form of ontologies in order for the message to be understood by the agents.
There are various multi-agent systems which incorporate ontology to facilitate the sharing of heterogeneous, autonomous knowledge sources in a capable, adaptable and extensible manner. They are known as ontology-based multi-agent system. Ontology-based multi-agent system utilized ontology to provide a formal model that reflects the agent domain; allowing a safe communication between various agents within the open/closed environment; and enabling a process of inference concerning the different events that might occur within the agent domain (De Azevedo, R. R. et al. 2010). Greer K. et al. (2004) developed application which facilitates ontology-based access to web services through agents, whereby ontology is being developed to store information describing the Web Service which will be used as the parameters structure to be transferred when making an operation call. Hunyadi et al show the integration of domain ontology and application ontology in an E-Learning Multi-Agent Architecture (Hunyadi, D. et al. 2008). The domain ontology defines the concepts/properties within the training domain whereas the application ontology denoting the organization of academic concepts that are studied during the training session. Prashant M. D. presents the integration of ontologies into Multi-Agent Systems Engineering (MaSE) for University Teaching Environment (Dolia, P. M. 2010). Domain ontology is being used here to define the concepts and relationships that exist within the teaching domain in governing the data transferred between agents. Choinski, D. et al. (2012) have employed ontology mechanisms in handling knowledge management and learning mechanisms needed within the Multi-Agent System (MAS). Ontology is used as the hierarchical representation of the specific domain knowledge in the multi-agent system to facilitate ontology based knowledge sharing.

To date, ad-hoc method is used for developing ontology among the agent community (Breitman, K. K. 2007). The ad-hoc approach is built based on their own perspectives, without
adherence to any ontology standard, for example, most of the ontologies are developed by people who have only a brief knowledge of the underlying logical formalisms of ontology language, instead of carefully developed by a group of expert engineers (Blomqvist, E. 2010). It has been reported that the use of ad-hoc approach brings to the absence of common and structured guidelines that slowed the development of ontologies (Asunciión Gómez-Pérez. et al. 2004) and causes incompatibilities issue when software developers trying to link them together (Brinkley, J. F. et al. 2006). DiLeo, J. et al. had extended the Multi-Agent System Engineering (MaSE) methodology by incorporating ontology engineering for information domain specification (DiLeo, J. et al. 2002) in the development of a multi-agent system. In the work, a step to engineer the System Ontology is added into the MaSE to allow the construction of ontologies during the MaSE analysis phase. Once the system ontology is being developed, the methodology allows the analyst to specify the information flow by using object from the ontology as parameters in the conversations between the agents.

Based on a research carried out by Hammar K. et al. (2009), efficient ontology development continues to be a challenge as it still requires both a lot of domain knowledge/experience and knowledge of the underlying logical theory. Therefore, ontology patterns are introduced to ease the knowledge reuse and aid the tasks of engineering ontologies (Hammar, K. 2009).

A pattern in general is a template that describes solution for a recurring problem. Patterns facilitate communication and sharing of knowledge among software practitioners (Cicortas, A. et al. 2008) (Bushman, F. et al. 2007). There are a wide variety of ontology patterns available nowadays and these patterns can be categorized into different levels. For example, ontology design patterns (ODP) that intends to describe a common recurring construct in ontologies
development (Blomqvist, E. et al. 2005). Content ODP named Action is an example of ontology design patterns that represent the relations between different types of actions, the state of the actions, and defines the relation between a plan and a set of proposed actions (Blomqvist, E. 2010).

To ease the adoption of patterns among the wider software community, pattern classification had been introduced. Pattern classification is an arrangement of patterns into groups with similar set of properties. The main goal of pattern classification is to make it easier for software developers to select the right pattern to apply (Hammouda, I. et al. 2004). It is believed that with a well established pattern classification framework, it can further improve the selection and comprehension of ontology pattern among the software practitioners. Hence, it enables a proper organization of patterns into its appropriate dimensions and categories which is useful for the user to find and identify patterns that can match to their problem at hand.

An investigation on the experience of developing ontology-based multi-agent system had been conducted. It has been discovered that various experiences in adopting the ontology for multi-agent system have been introduced and practiced in agent interaction and agent reasoning but these experiences has not been recorded into patterns for reuse purpose (Fuzitaki C. 2011). In fact, WaiShiang (WaiShiang, C. 2010) had further claimed for the needs to have ontology patterns for ease of multi-agent system development.

This thesis is based on the premises that the adoption of agent-oriented ontology patterns is able to help software practitioner to start the ontology engineering pragmatically in the development of a multi-agent system. Therefore, to ease the adoption of agent-oriented ontology patterns in the development of multi-agent system, we introduce 90 agent-oriented ontology patterns that can be employed within the agent domain and a pattern classification scheme to
facilitate the selection of agent-oriented ontology patterns for reuse purpose among the agent community.

1.1 Research Problem

Although ontology patterns have been introduced, we report the insufficiency of the adoption of ontology patterns for ontology-based multi-agent system development as following:

- Most of the ontology patterns lack context. It is hard to reuse and use in domain specific development such as agent system. For example, various categories of ontology patterns have been introduced. How do those categories like ontology application patterns, ontology architecture patterns, ontology design patterns, ontology semantic patterns and ontology syntactic patterns are able to relate to ontology-based multi-agent system development?

- Lack of ontology pattern for multi-agent system. Ontology has been used on agent communication, agent reasoning, and as agent knowledge. However, the experience in the use of ontology within multi-agent system development does not much explore.

1.2 Motivation

This research is motivated by the following:

- The need of agent-oriented ontology patterns for multi-agent system development. One way of incorporating the ontology for agent development is through ontology engineering. Alternatively, we seen the value of ontology patterns in improving the adoption of
ontology for agent development. As a result, exploring the ontology patterns that can be employed in the development of the multi-agent system is worth researching.

- The need to organize the agent-oriented ontology patterns based upon an appropriate classification scheme so as to promote reusability among the agent community. With the increase of ontology patterns, it is necessary to appropriately group the patterns to make it easier for software practitioners to select the right pattern to apply to a particular problem domain.

- The need to have an authentic case study to demonstrate on the adoption of an agent-oriented ontology pattern for multi-agent system development. The usage of the agent-oriented ontology patterns that have been classified through the classification scheme is worth to explore. This case study shall reveal the potential of the agent-oriented ontology patterns and classification for the development of agent systems.

1.3 Aims

The aim of this thesis is to explore on agent-oriented ontology patterns in the development of ontology-based multi-agent system. It is believed that reusing the agent-oriented ontology patterns is able to facilitate sharing and reuse of experiences among wider software community. Moreover, pattern classification is needed to improve the adoption of agent-oriented ontology patterns in multi-agent system development so as to enable sharing of knowledge in a capable, adaptable and extensible manner.

This thesis shall addresses to the following issues:-
• How to rapidly prototype an ontology-based multi-agent system in a pragmatic and comprehensive manner?

• How pattern classification can promote the adoption of agent-oriented ontology patterns for ontology-based multi-agent system development?

This thesis addresses to the issues listed by exploring on ontology patterns that can be used in developing ontology-based multi-agent systems. A comprehensive pattern classification that is based upon the Two-Ways classification scheme to analyze and classify agent-oriented ontology patterns is thoroughly investigated. In addition, the potential usage of the agent-oriented ontology pattern is explored through the case study.

1.4 Scope

We adopt the current practice in organizing pattern through patterns and pattern classification.

The ontology patterns that we are going to observe in this research are the patterns which can be commonly utilized for the development of agent-oriented application. These patterns are collected based on the existing works related to ontology pattern which can be used in the development of the multi-agent system.

In addition, the pattern classification work in this thesis extends to the classification scheme that is introduced by WaiShiang, C. (2010). Patterns are classified based upon two dimensions namely: - software process dimension and aspect dimension. Detailed analysis will be performed on the patterns collected and these patterns will be categorized accordingly based on the two dimensions in the classification scheme.
To demonstrate the feasibility of our approach, a case study is used to demonstrate the adoption of agent-oriented ontology pattern in ROADMAP/AOR methodology for the development of ontology-based multi-agent system related to the travel industry. Pattern selection principals are identified to support the incorporation of agent-oriented ontology pattern within the ROADMAP/AOR methodology. The development of the multi-agent system will be based on the Java Agent Development (JADE) platform.

1.5 Thesis Contributions

This thesis makes a main contribution to the field of agent-oriented software engineering in the area of agent-oriented ontology especially during the development of ontology-based multi-agent system. This main contribution encompasses the following:

- A novel pattern classification scheme for analyzing and classifying agent-oriented ontology pattern is introduced in ontology-based multi-agent system development. This approach is based on the improved Two-Way classification scheme which broadly categorizes patterns by means of two dimensions: the software process dimension and the aspect dimension. Each of the dimensions consists of sets of attribute which is used for analyzing and classifying the agent-oriented ontology patterns collected. These attributes are redefined based upon the ontology concept/properties needed during the development of multi-agent system. The pattern classification scheme leverage the know-how in developing agent knowledge; know-how of agent interaction in open system; and know-how of developing agent intelligent through ontology during the development of the multi-agent system. In other words, developing ontology for multi-agent system requires the experience on how to develop the domain knowledge for agent system; how the
ontology can be structured on message exchange (e.g. agent communication and resolving conflict); how the ontology can be infer for agent reasoning.

• The agent-oriented ontology patterns are introduced towards the adoption of ontology in the development of multi-agent system. This is done by exploring 90 ontology patterns that can be commonly and specifically used during multi-agent system development. Based on the classification result of the 90 agent-oriented ontology patterns collected, we observed that most of the agent-oriented ontology patterns distributed at the platform-independent computational design level whereas few works have been performed at the conceptual domain modeling level and platform-specific design and implementation level. Therefore, it can be seen that the current agent-oriented ontology pattern can provide ample solution for recurring modeling and design problem in the development of an ontology-based multi-agent system. On the other hand, software practitioners can carry out more research works, related to the sharing of ontology engineering experiences at analysis level and implementation level.

• The pattern selection guidelines are proposed in this thesis. These guidelines are used to determine the type of agent-oriented ontology patterns to be incorporated within each stage in the ROADMAP/AOR methodology during the development of a multi-agent system.

• A case study to demonstrate the adoption of agent-oriented ontology patterns in JADE multi-agent system development is introduced. The main purpose of this case study is to demonstrate the feasibility of the pattern and pattern classification in the development of the multi-agent system. The case study that is presented in this research is the
development on the reusing of agent-oriented ontology patterns in a multi-agent system related to the travel industry. It is the first case study showing the incorporation of agent-oriented ontology patterns into ROADMAP/AOR methodologies in developing an ontology-based multi-agent system. Apart from that, we have showed the evaluation of a systematic development of an ontology-based multi-agent system from ontology engineering to ontology pattern within agent-oriented methodology. With this approach, we are able to reduce the need for extensive ontology engineering experience during the development of an ontology-based multi-agent system.

1.6 Research Methodology

This section presents an overview of the methods to be used in the study.

Phase 1: Data Collection

In this phase, ontology patterns that are used in the development of multi-agent system will be collected from various sources such as patterns that have been published through conference paper and related web sites. Most of the patterns will be collected from websites such as below:-

http://ontologydesignpatterns.org

http://www.w3.org/2001/sw/BestPractices/OEP

http://www.gong.manchester.ac.uk/odp/html/index.html

Phase 2: Study the Classification Scheme and Working on the Attributes of the Classification Scheme
The Two-Way classification scheme will be studied based on the two dimensions: software process dimension and aspect dimension. Definition for each of the level in the software process dimension will be defined. Then, the aspect dimension at each of the level will be further defined. The definition will be derived based on the study of ontology concepts that are described in various ontology-engineering methodologies and relate them to the concept that is describe within the viewpoint framework.

After each of the level and aspect has been defined with respect to the ontology domain, attribute-based analysis is being used to classify the patterns into its corresponding level and aspect within the classification scheme. Refinement on the attributes in the Two-Way classification scheme is being done to flawlessly adopt the ontology patterns within the classification scheme. This refinement is being carried out by identifying attributes in each of the software process and aspect dimension through studying on various ontology engineering methodologies and agent-oriented software development projects.

Follow by that, pattern by pattern analysis and classification is being done. Patterns are observed and reviewed to identify elements that might be shareable from its description, maps the pattern to its appropriate levels and categories on the level attributes table analysis and category attributes table analysis. The determination of the level and category to which a pattern belongs to is based upon the frequency of occurrence of attributes within the classification scheme.

**Phase 3: Validate the Classification Scheme**

Validation on the classification scheme is being carried out by analyzing and classifying the ontology patterns collected previously and presents the result in a comprehensive manner. A sample collection of ontology patterns will be listed and the result of the analysis will be
presented in the form of analysis tables. Two type of analysis table will be developed which are software process dimension analysis table and aspect dimension analysis table. Each of the patterns will be mapped into both of these analysis tables in order to determine which level and aspect they belong to. Consequently, observation will be performed on the result of classification.

**Phase 4: Demonstrate the adoption of patterns for MAS development**

(a) **Proposing Guidelines**

Guidelines are being proposed to ease the selection of ontology patterns for the development of ontology-based multi-agent system. These guidelines will be used to assist agent developer during pattern’s selection.

(b) **Developing prototype**

A case study on developing an ontology-based multi-agent system related to the travel industry through reusing the ontology pattern will be presented. This prototype is developed within the combined modelling process of ROADMAP and AOR methodologies. A demonstration on the reusability of the ontology patterns is presented in the development stage by following the guidelines proposed. JADE (Java Agent Development Environment) is used as the development platform due to its ability of sharing implementation code which is greatly suitable with the current agent development practice.

(c) **Evaluation**

An evaluation process will be performed to assess on the practicability of our agent-oriented ontology patterns and pattern classification scheme. This evaluation is performed by
requesting an evaluator to develop the case study proposed through the adoption of agent-oriented ontology patterns within the pattern classification scheme introduced. Finally, a survey is conducted to collect information on his/her experiences and challenges faced throughout the development of the case study.

1.7 Thesis Overview

This thesis comprises of six chapters. An overview of the thesis layout is shown in Figure 1.1. The remainder of this thesis is organized as follows:

Chapter Two is a literature review, providing background materials and survey on related works. It covers the importance of ontology development in multi-agent system; the needs of patterns in supporting the agent-application development and its adoption. This is followed by a review on the current practice of developing ontologies in multi-agent system development. Subsequently, issues related to the adoption of ontology patterns for multi-agent system is discussed. Lastly, a discussion on the classification scheme and processes which shall be adopted throughout this research is presented.

Chapter Three presents an overview of agent-oriented ontology pattern and an agent-oriented ontology pattern classification scheme that is based upon the improved Two-Ways classification scheme. A description on agent-oriented ontology patterns and its need, as well as a sample collection of these patterns is presented. Subsequently, the dimensions within the pattern classification scheme are defined based on the study of ontology concepts that are described in various ontology-engineering methodologies. Follow by that, attributes for the dimensions within the classification scheme are defined based on the ontology concepts/properties needed during the multi-agent system development.
Chapter Four describes the attribute-based analysis that is used for the classification of agent-oriented ontology pattern through pattern classification scheme. It involves analyzing and classifying the agent-oriented ontology patterns using level and category attributes, through sets of attribute tables. Several examples on classifying agent-oriented ontology patterns through the classification scheme are presented in this chapter. Subsequently, the chapter presents the validation of the classification scheme by classifying 90 agent-oriented ontology patterns. The sources of the agent-oriented ontology patterns are presented initially. This is followed by the result of the classification of 90 agent-oriented ontology patterns. Lastly, the observations of our result are presented at the end of the chapter.

Chapter Five presents the evaluation of the proposed pattern classification through a case study of an agent-based travel support system. This case study comprises of two scenarios: - (1) finding relevant restaurants; and (2) adding a new restaurant details. It includes a demonstration on the adoption of agent-oriented ontology patterns supported by the classification scheme in developing both scenarios of the agent-based travel-support system. The results and experiences from the process of developing this case study provide an evaluation of our approach.

This thesis concludes with Chapter Six which presents the conclusions reached by this research and possible future works.