The Hong Kong Polytechnic University

Department of Building and Real Estate

KNOWLEDGE ACQUISITION METHODS FACING REAL ESTATE ENTERPRISES

Li Min

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

September 2010

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written nor material which has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

(Signed)

Li Min (Name of student)

ACKNOWLEDGEMENTS

The research is supported by a postgraduate studentship provided by The Hong Kong Polytechnic University. I would like to express my gratitude to Professor Heng Li, my chief supervisor, for his enthusiasm, his kind advice, numerous constructive comments and the revisions of my writings. I am deeply grateful to his guidance, support and friendship, and I am deeply impressed by his rigorous scholarship and his scholarly research works. The research cannot be finished without his huge support and contribution.

ABSTRACT

Although the real estate industry is one of the largest and sturdiest pillars of every economy, it suffers from many deficiencies. For example, it lacks adequate tools to both effectively acquire the latest, most comprehensive and relevant knowledge, and to utilize that knowledge. This study explores the knowledge acquisition methods applicable to the real estate industry and uses this as the basis for the development of an effective real estate knowledge management system.

Firstly, in order to solve the problem of acquiring the necessary knowledge from the wide and diverse world of real estate, this dissertation proposes building a knowledge management “union” consisting of a number of real estate enterprises cooperating in the sharing of knowledge. An inconsistent, semantic interpretation of data is one of impediments to the effective exchange of knowledge between real estate enterprises. Focusing on this challenge, this dissertation develops an ontology-based agent cooperation search model. The framework of the model is described as well as the design of the process of the agent cooperation search. Through a study of ontology modeling theory, the steps of real estate domain and process ontology modeling are defined. Finally, it is demonstrated how to model the real estate ontology in order to realize the cooperative search by means of both a real estate project management ontology modeling example and a real estate project management knowledge cooperation search example.

Secondly, this dissertation proposes a form of genetic algorithm-based knowledge

discovery of the relevant implicit knowledge from the internal databases of real estate enterprises. It provides the framework for and analyzes the effectiveness of such a knowledge discovery methodology. In addition, it demonstrates how this method can be applied in real life through an example of a knowledge discovery exercise of this type.

In conclusion, this dissertation explores the design of a practical real estate knowledge management prototype system. It provides the design principles and the framework of the system as well as the structure of its major modules. The aim of the prototype system is to show how to apply the above model to the development of a real estate knowledge management system that will deliver commercial value to its users

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION 1

1.1 Research Background 1

1.2 Research Objectives 2

1.3 Research Methodology 3

1.4 Organization of the Dissertation 4

CHAPTER 2: LITERATURE REVIEW 7

2.1 Introduction 7

2.2 Ontology Modeling Theory 8

2.3 Agent Cooperation Search Technology 11

2.4 Methods of Knowledge Discover from Database 20

2.5 Patterns of Knowledge Management System 26

2.6 Knowledge Management Issue in Real Estate Enterprise 35

2.7 Conclusions 38

CHAPTER 3: REAL ESTATE ONTOLOGY MODELING 40

3.1 Introduction 40

3.2 Comparison of Common Knowledge Representing Methods 40

3.3 Ontology Modeling Basic Theory 43

3.4 Domain Ontology Modeling 54

3.4.1 Domain Ontology Concept Gain 54

3.4.2 Domain Ontology Modeling Method 61

3.5 Process Ontology Modeling 69

3.5.1 Description Logic Based Process Ontology Criterion 69

3.5.2 Process Ontology Based Knowledge Base 79

3.5.3 Process Ontology Modeling Method 87

3.6 Experiment 92

3.6.1 Background of Experiment 92

3.6.2 Real Estate Project Management Domain Ontology Modeling 95

3.6.3 Real Estate Project Management Process Ontology Modeling 97

3.7 Conclusions 101

CHAPTER 4: ONTOLOGY BASED AGENT COOPERATION SEARCH MODEL 102

4.1 Introduction 102

4.2 The Agent Cooperation Frame of FIFA 102

4.3 The Frame of Ontology Based Agent Cooperation Search Model 107

4.3.1 Storage Layer 108

4.3.2 Semantic Layer 109

4.3.3 Service Layer 110

4.3.4 Application Layer 114

4.3.5 Compared with the Frame of FIFA 116

4.4 Design of DLOP 117

4.5 Design of Cooperation Process..120
4.5.1 Initiate Cooperation..121
4.5.2 Seek Cooperation 121
4.5.3 Building Cooperation Plan by Negotiation 122
4.5.4 Execute Cooperation 122
4.5.5 Monitor Cooperation 123
4.6 Experiments...125
4.6.1 Experiment Scheme...125
4.6.2 Process of Real Estate Project Management Knowledge Cooperation Search.........128
4.7 Conclusion..136

CHAPTER 5: GENETIC ALGORITHM BASED KNOWLEDGE DISCOVERY METHOD.......................137
5.1 Introduction...137
5.2 Application Prospect of KDD in real estate industry...138
5.3 Principle of Genetic Algorithm 139
5.4 Genetic Algorithm Based Knowledge Discovery 143
5.5 Experiment 146
5.5.1 The Real Estate Project Sale Database 146
5.5.2 Process of Real Estate Project Sale Knowledge Discovery 148
5.5.3 Result of Real Estate Project Sale Knowledge Discovery 151
5.6 Conclusions 151

CHAPTER 6: DESIGN OF REAL ESTATE KNOWLEDGE MANAGEMENT PROTOTYPE SYSTEM......153
6.1 Introduction 153
6.2 Design Principle of Real Estate Knowledge Management Prototype System................153
6.3 Architecture of Real Estate Knowledge Management Prototype System.......................154
6.4 Design of Major Modules of Real Estate Knowledge Management Prototype System ..156
6.4.1 Ontology Knowledge Gain Module and Ontology Modeling Tool..........................156
6.4.2 Ontology Parser...158
6.4.3 Model-View Converter..160
6.4.4 Agent Cooperation Search Module 161
6.4.5 Knowledge Discovery Module..166
6.5 Development and Validity of Real Estate Knowledge Management Prototype System..167
6.5.1 Development of Real Estate Knowledge Management Prototype System...............167
6.5.2 Validity of Real Estate Knowledge Management Prototype System 167
6.6 Comparison with the Protégé Tool Developed by the Stanford University 168
6.6.1 Purpose...168
6.6.2 User 168
6.6.2 Running Pattern...168
6.6.2 Functions 169
6.7 Conclusions...169

CHAPTER 7: CONCLUSION..171
7.1 Introduction 171
7.2 Contributions at Theory Level..171
7.3 Contributions at Model Level...172

7.4 Limitations....173
7.5 Future Direction of Research....173

References....176

LIST OF FIGURES

Number *Page*

Figure 1.1 Methodology of the Dissertation ... 4

Figure 1.2 Structure of the Dissertation ... 4

Figure 2.1 Layer Model Based Knowledge Management System ... 27

Figure 2.2 General System Framework Based Knowledge Management System ... 27

Figure 2.3 Lifecycle Based Knowledge Management System ... 28

Figure 2.4 Knowledge Practice Framework Based Knowledge Management System ... 29

Figure 3.1 Knowledge Representing System Architecture Based On Description Logic ... 46

Figure 3.2 Tableau Satisfiability Algorithm ... 48

Figure 3.3 Ontology Gain Process Diagram ... 55

Figure 3.4 DLOP Core Conception-Relation Diagram ... 70

Figure 3.5 Whole Process of the Real Estate Project Management ... 93

Figure 3.6 Ontology Gain Window ... 96

Figure 3.7 Domain ontology schematic drawing of Real estate knowledge management system ... 97

Figure 3.8 PreStructureEstablishProcess ... 99

Figure 3.9 MediStructureEstablishProcess ... 100

Figure 3.10 PostStructureEstablishProcess ... 100

Figure 3.11 StructureEstablishProcess ... 101

Figure 4.1 Agent Cooperation Model ... 103

Figure 4.2 Agent Life Period Diagram ... 104

Figure 4.3 Frame of Ontology Based Agent Cooperation Search Model ... 107

Figure 4.4 Latent Agent Discovery Algorithm ... 114

Figure 4.5 Process Parser Algorithm ... 119

Figure 4.6 Cooperate Control Process of Cooperative Initiator ... 120

Figure 4.7 Cooperation Implementation ... 120

Figure 4.8 CPMA Monitoring Management Algorithm ... 125

Figure 4.9 Architecture of Real Estate Project Knowledge Cooperation Search System ... 128

Figure 4.10 Parameters Setting Interface of Coordinator CPMA 129
Figure 4.11 Registration Interface of RealEstateCompany Agent 130
Figure 4.12 Role Defining Interface of RealEstateCompany Agent 130
Figure 4.13 Agent List Online Interface of Coordinator APMA 131
Figure 4.14 Knowledge Request Interface of RealEstateCompany Agent 132
Figure 4.15 Responding Interface to Knowledge Request of HousingSaleAgency Agent 133
Figure 4.16 Agent Cooperation State Inspect Interface of CPMA 134
Figure 4.17 Agent Cooperation State Inspect Interface of RealEstateCokmpany 135
Figure 4.18 Operation Interface of the Protégé Tool 135
Figure 5.1 Genetic Algorithm Flow 142
Figure 5.2 CRISP-DM Model 143
Figure 5.3 Database Knowledge Discovery Model Based on Gene Algorithm 145
Figure 6.1 Architecture of Real Estate Knowledge Management System 156
Figure 6.2 Protégé Development Interface Chart 157
Figure 6.3 Jena Knowledge Ontology API Framework 159
Figure 6.4 Relation Between Pattern modules in JSP 160
Figure 6.5 Model-View Converter Interactive Sequence 161
Figure 6.6 Agent Structure And Multithread Concurrent Control Mechanism 162
Figure 6.7 Multithread Dataflow of Agent Structure 165

LIST OF TABLES

Number *Page*

Table 3 .1 ALC Syntax & Semantic 52

Table 3.2 Context Matrix 58

Table 3.3 DLOP Construction Semantic 73

Table 4.1 Table1 Basic Services Provided By AMS 112

Table 5.1 Visitor & Customer Information 147

Table5.2 Transaction Record 147

Table 5.3 Unit Information 147

Table 5.4 Building Information 147

Table 5.5 Transaction Record Tidied up 149

Table 5.6 Coding Rule of Transaction Record 149

Table 5.7 Some of Customer Information Records Coded 150

LIST OF ABBREVIATIONS

MAS	Multi-Agent System
FIPA	Foundation of Intelligent Physical Agents
AMS	Agent Management Service
DF	Directory Facilitator
MTS	Message Transport Service
CPMA	Cooperation Process Management Agent
OWL	Ontology Web Language
KDD	Knowledge Discovery from Database
DLOP	Description Logic based Ontology Process

Chapter 1

INTRODUCTION

1.1 Background

Although the real estate industry is the cornerstone of any economy, it suffers from many deficiencies. For example, the industry lacks adequate tools to effectively acquire and manage relevant knowledge of past activities. How to capture and disseminate knowledge that exits in the interior and exterior organization is becoming an important problem to real-estate developing enterprises (Lin Li 2006). The development of modern information technology provides significant opportunities to resolve this shortcoming. How to effectively employ IT in the development of the real estate industry has attracted significant attention from interested researchers and professionals in recent years. This dissertation aims to advance this area of study.

Every industry has its own unique characteristics. The real estate industry has the following distinctive traits:

1. Its activities extend over the entire lifecycle of a property. Therefore, it needs knowledge at every stage, from land acquisition, design, tendering, construction, sale, operations and maintenance, and ultimately demolition. Real Estate Groups are Knowledge-Intensive Organization (Jennifer 2001). Knowledge takes various forms over the course of this cycle, some examples of which are: professional books, policies, laws, standards, articles, studies, contracts, experience, and so on.

2. The relevant knowledge is rapidly changing in parallel with advances in science

and technology. It also must be kept aligned with government policies and standards which frequently change or are updated continually.

3. Much of this knowledge exists in implicit form, either in the heads of experts or in large, dispersed databases.

4. There is rapid employee turnover from one project or position to another and from one company to another. This effect is magnified by frequent recruitment and dismissal within the industry.

Given these peculiarities, real estate enterprises face the following challenges:

1. How to retain the latest, most comprehensive and relevant knowledge over the lifecycle of a property; and,

2. How to exploit and manage the implicit knowledge of the enterprise?

The development of modern information technology provides some promising tools for solving these problems. The two issues above can be expressed in another way: 1) how to efficiently acquire the latest relevant knowledge from the market in which the real estate enterprise operates; and, 2) how to harvest the implicit knowledge within the business using the enterprise's own internal resources. Existing technology and methods can be used for this knowledge acquisition, including: ontology modeling, multi-agent cooperation modeling, data mining methods, etc. The dissertation explores how to apply developments in IT to the real estate industry in order to accelerate growth in this field.

1.2 Objectives

This dissertation explores knowledge acquisition methods from outside of and within a real estate enterprise; most specifically, ontology based agent cooperation search modeling and genetic algorithm based knowledge discovery methods. Its final objective is to lay the foundation for the development of a kind of applied real estate knowledge management system which is suitable for most real estate enterprises, helps them to effectively manage relevant knowledge.

1.3 Methodology

In order to solve the problem of how to acquire the latest relevant knowledge from the external environment of a real estate enterprise, this dissertation proposes an ontology based agent cooperation search model. The task of acquiring knowledge from the Internet is highly complex. The dissertation aims to achieve this by means of cooperation between agents which represent the real estate enterprise or the subsidiary of a large real estate group. Ontology modeling is used to formalize the semantic of distributed agents because the unanimous semantic is the basis of cooperation between agents.

To achieve knowledge acquisition inside the enterprise, this dissertation proposes a genetic algorithm based knowledge discovery method, which is used to mine the knowledge hidden in a business's databases, such as the relationship between or among variables.

These methods used in this dissertation can be described by the figure 1.1.

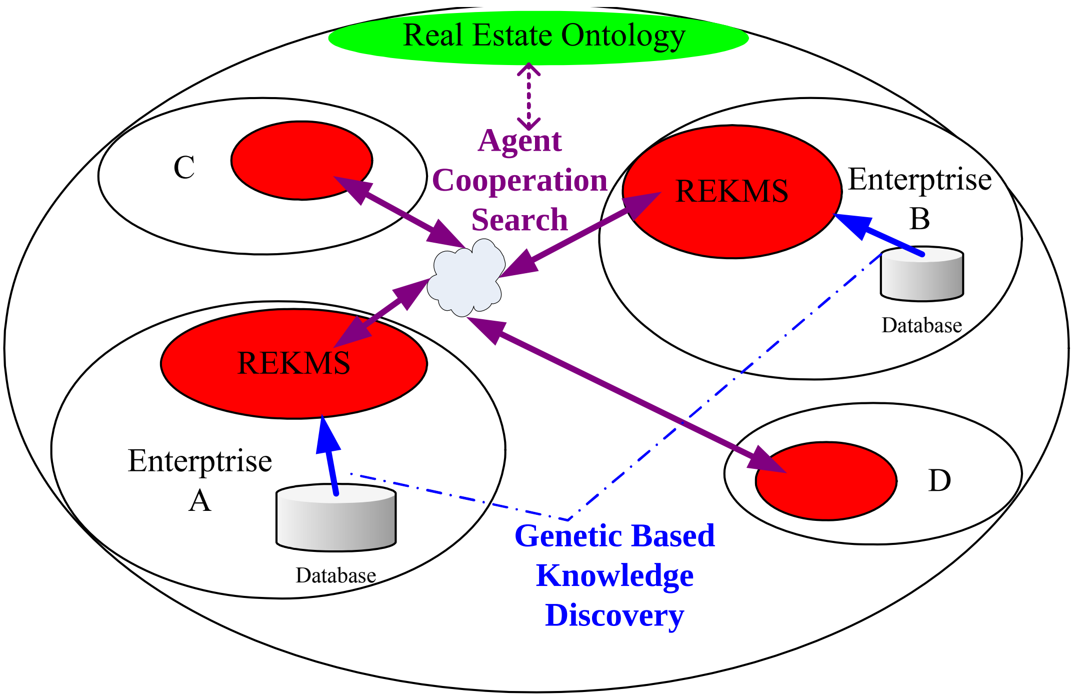


Figure 1.1 Methodology

1.4 Organization of the Dissertation

There are seven chapters in this dissertation. These chapters are organized according to their relationship to the objectives of the research. These relationships are shown in Figure 1.2.

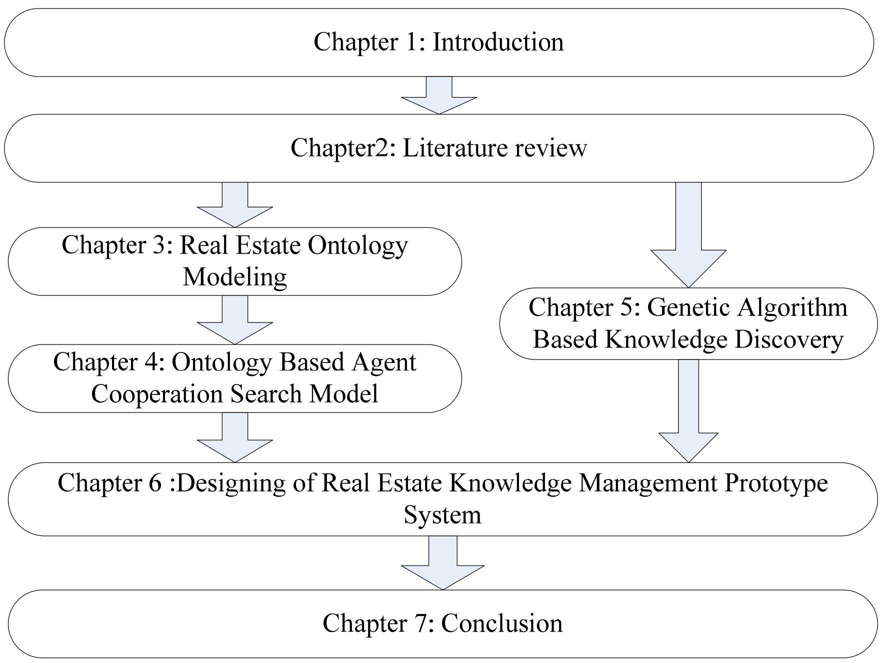


Figure 1.2 Structure of Dissertation

Chapter 1: Introduction

This chapter introduces the background, aims, research methodology and organizational structure of the dissertation.

Chapter 2: Literature Review

This chapter summarizes and analyzes the current state of the theory and the technology relevant to the research content of the dissertation. The theory and the technology involved mainly reflect ontology modeling theory, agent cooperation technology, methods of knowledge discovery, patterns of knowledge management systems and knowledge management issue in real estate enterprises

Chapter 3: Ontology Modeling

Compared with other common knowledge representing way, the ontology method is selected as the knowledge representing means for sharing knowledge between real estate enterprises. Ontology modeling is one method for resolving the inconsistent semantic understanding during the exchange of knowledge among different real estate enterprises. Through the study of ontology modeling theory, this chapter develops the steps for real estate domain ontology and process ontology modeling. Furthermore, it demonstrates how to model real estate ontology by means of a real estate project management ontology modeling example.

Chapter 4: Ontology Based Agent Cooperation Search Model

In order to solve the problem of acquiring knowledge from outside of the enterprise, this chapter proposes building a knowledge management union consisting of real estate enterprises to realize the sharing of relevant knowledge. Inconsistent semantic understanding is one of the difficulties in effectively exchanging knowledge among real estate firms. Focusing on this difficulty, the chapter brings forward and builds the ontology based agent cooperation search model. It gives the framework of the model and designs the process of the agent cooperation search. It then demonstrates how to realize a cooperation search by means of a real estate project management knowledge cooperation search example.

Chapter 5: Genetic Algorithm Based Knowledge Discovery Method

This chapter suggests a genetic algorithm based knowledge discovery method for acquiring the implicit knowledge from the internal databases of a real estate business. It analyzes the application prospect of the knowledge discovery method in the real estate industry and provides the framework for genetic algorithm based knowledge discovery methods. In addition, it demonstrates how to apply the above method through an example of knowledge discovery from a real estate sales database.

Chapter 6: Designing of Real Estate Knowledge Management Prototype System

This chapter explores the design of a real estate knowledge management prototype system. It lays out the design principles and the framework of the system, as well as the design plans of its major modules. The aim of the prototype system is to show how to apply the above model and method to the development of a real estate knowledge management system.

Chapter 7: Conclusion

This chapter sums up the contributions and the limitations of this dissertation and gives some recommendations for further research.

C h a p t e r   2

LITERATURE REVIEW

2.1 Introduction

The goal of this dissertation is to explore the knowledge acquisition methods facing the real estate industry and using this as the basis for building a real estate enterprise knowledge management system. Knowledge management activities include: knowledge acquisition, knowledge coding and storage, knowledge delivery, knowledge use, knowledge appraisal, knowledge innovation, knowledge updating and maintenance, etc. Knowledge acquisition is the first stage of the knowledge management lifecycle. It is a very important initial step. Knowledge acquisition refers to adding meta-information to information. It is a transformation process from structured, semi-structured and unstructured information to the structured information. According to the source of information, there are four categories: (1) Background information of individuals and groups.(2)Various homogeneous or heterogeneous database.(3)Various documentations.(4)Knowledge over the web including information(Tiedong Chen, Ziyu Liu and Lei Huang 2007).This dissertation mainly study on how to acquire the information from the second and the forth sources, i.e. from the internal database and the external website.

With the development of IT and the increasing importance of knowledge for commercial success, knowledge management has risen in prominence in recent years. There are a number of researchers who have made valuable contributions in this field. This chapter will review the literature that closely relates to the context of this dissertation.

According to the research methodology described in Chapter one, relevant literature can be classified in the following ways:

- Ontology modeling theory
- Agent cooperation search technology
- Methods of knowledge discovery from databases
- Patterns of knowledge management systems
- Knowledge management issue in real estate enterprises

2.2 Ontology Modeling Theory

The ontology concept derives from the philosophical domain and can be traced back to the ancient Greek philosopher, Aristotle (384-322 BC). In philosophy it is defined as "the description of the systematic in the world, namely the theory of existence." It is the systematic explanation of the objective existence, and it concerns itself with the abstract essence of objective reality.

Ontology in computer science can be traced back to the 1980s. McCarthy used this term in 1980, and Hayes and Sowa used it in 1984. But, the article published by Alexander in 1986 is regarded as new research of ontology in the computer domain which is different from philosophical domain. Subsequently, the concept of ontology was further developed in the field of artificial intelligence, and was gradually endowed with a new meaning (Christopher Welty 2003). Neches and other people are considered to be the persons who first defined ontology in the field of artificial intelligence. They maintained: "this ontology defined the basic terminology and relations composing the glossary table in the subject domain, and the rules combining these terminology and relations to define the extension of the glossary table" (Robert Neches, etc. 1991).

The most famous and widespread quotable definition is the one proposed by Thomas R. Gruber (1993): “the ontology is the conceptual, explicit and standard explanation”. The ontology definition given by researchers represented by Gruber is established on the base of conceptualization in the traditional artificial intelligence teaching material “Logical Foundations of Artificial Intelligence” (M. R. Genesereth and N. J. N. 1987). The ontology definition was synthesized and further analyzed in literature (Christopher Welty. 2003). It now has the following four meanings:

(1) Conceptualization: it is the abstract simple observation (Thomas R. Gruber 1993) of the described world starting from the special goal. It can be obtained through abstracting some phenomenon in the objective world. So, its meaning is independent from the concrete environmental state;

(2) Formal: it is the accurate math description, and can be read by computers;

(3) Explicit: the concepts, relations and restriction are all defined explicitly; and,

(4) Shared: what the ontology embodies is the common knowledge, and what it reflects is the common concept set in the related domain. It refers to the team but not the individual.

Different researchers have attained common cognition about the ontology: the ontology is the semantic base of the communication (dialog, interoperability, sharing and so on) between the different entities (e.g. person, machine or software system and so on) in the domain (maybe specific domain, or a broader scope). The ontology draws the elementary knowledge system in the special domain through the standardized description about the concepts, terminology and relations. So, its core meaning lies in providing one kind of the definite, clear and formal cognition to enable the cooperation between machines. The machine cannot understand the semantic implication in the natural language as man does. Therefore, the discussion of the ontology in the computer domain should be about how to express the common cognition, namely, how to represent the formal, definite concept.

The ontology representation language should generally have rich and direct-viewing power of expression, and be conveniently comprehended, processed and applied by computers. The ontology representation languages can be divided into two kinds at present: the ontology representation language based on the predicate logic and the ontology representation language based on the graph.

Now, most of the ontology representation languages based on the predicate logic use XML syntax. For example, XOL (XML-based Ontology exchange Language); SHOE(Simple HFML Ontology Extension); OML (Ontology Markup Language); the establishment on RDF(s)- OIL (Ontology Interchange Language) and DAML (DARPA Agent Markup Language)+OIL; KIF (Knowledge Interchange Format), which is a formal language based on the predicate calculation; the ontology framework language ONTOLINGUA, which uses the KIF as its core; the ontology based knowledge representation language CYCL; the first-predicate logic based knowledge ontology representation language LOOM; F-Logic (Frame logic) integrating with the frame and first-predicate language, and so on. At present, OWL (Web Ontology Language) is also one popular kind of the ontology description language (Sean Bechhofer, etc. 2004) proposed by W3C, and comes from DAML+OIL, keeps the DAML+OIL frame and the majority syntax, semantic characteristic, and simultaneously extends and restricts itself aiming at different application scopes (different representation and computing capability). It has three sub-languages: OWL Full，OWL DL and OWL Lite；As the restriction of three sub-languages increases, their representation capability decreases, but their calculability (the conclusion can be drawn through computers) strengthens in turn.

The graph based representation method is also an important one, and its most remarkable characteristic is of intuition. The representative ontology languages based on graphs are: WordNet; CG (Conceptual Graph); Conceptual Representation used by Roux, et. al. DAG (Directed Acyclic Graph) used by Faure and Poibeau; LcG (Lexical Conceptual Graph) used by Borgo, Lexieal.

At present, there have been many engineering projects referred to the ontology development in different domains. There are also several ontology development processes and methods in the base of these engineering practices. They are summed up as follows: the enterprise modeling method (Michael Gruninger and Mark S. Fox 1995),the framework method (Mike Uschold and Michael Gruninger 1996) and the ontology development 101 method (Natalya F. Noy and Deborah L. McGuinness 2001).

2.3 Agent Cooperation Search Technology

The traditional information retrieval is to find the related texts satisfying the user’s request from numerous texts, according to the user's inquiry condition (namely key words). Its crucial problem is to differentiate the related texts from the irrelevant texts. The traditional information retrieval model may roughly be divided into three kinds: Boolean model (Salton G. and M. M. J. 1983; C. Menzel and M. G. 2001), vector space model and probabilistic model (Robertson S. and S. J. K. 1979).

Boolean model is the most typical set model, and is the basic component provided by the information retrieval system. It is widely applied in the traditional information retrieval. It represents the text by Boolean expression, and then through comparing with the user's inquiry expression logically to retrieve the related texts. So, the standard Boolean logistic model is the binary logic. In the Boolean model, a series of binary feature variables should be first defined aiming at the texts, and are usually the text index keywords extracted from the text, and sometimes they also include some more complex characteristic variables, such as the data, the phrases, the personal signature, the descriptive words added manually and so on. Next, these characteristic variables are used to represent the DI= (di_1, di_2,…, di_n), therein: n is a characteristic amount; di_k is True or False, if characteristic item k appears in the DI, endow it with True value, otherwise, endow it with False. In Boolean model, the user may connect many keywords into a logical expression to submit the inquiry with the logical operators such as: "$\wedge$" (AND), "$\vee$" (OR), "$\neg$" (NOT) and so on, according to the Boolean relation of retrieval keywords in the content. The match function is determined by the basic principle of the Boolean logic and retrieved through comparing the text expression with the user's inquiry expression logically, so the text is related or irrelevant to the inquiry.

The vector space model (VSM) has overcome the shortcomings of the binary weight in the Boolean model. It uses the non-binary weight to express the weight of the characteristic item in the text and users' inquiry, and proposes the model structure with the partial match. In the vector space model, the text is represented with the weighted vectors composed of characteristic items: the text vector DI= (t_1, wi_1; t_2, wi_2; …; t_n, wi_n), therein: n is a characteristic number; the characteristic item t_k is similar to the one in the Boolean model; wi_k is the weight of the characteristic item t_k in the text i. There are generally two methods to ascertain the weight wi_k. One is that the expert or the users endow the weight manually according to their own experience or the domain knowledge. However, this is inefficient and difficult to use for processing large-scale text collection. Another method is to utilize the statistics knowledge, the text statistical information (such as term frequency, the same present frequency etc.) to calculate the weight. Most statistical methods are based on the theory of Shannon information science: (1) the higher frequency of the characteristic item in all the texts, the less information entropy it contains; (2) if the frequency of the characteristic item is high only in the few texts, this characteristic item has high information entropy.

The probabilistic model (Probabilistic Model) (Robertson S and S. J. K. 1979) is used to overcome some uncertainty in the retrieval, and a kind of retrieval model taking the probability theory in mathematical theory as its principle. In this model, the representation of the text and user's inquiry is the same as the Boolean model. At the same time, according to the users' feedback, the texts are divided into two kinds (related and irrelevant), and then according to the distributed state of each characteristic variable (word) in the related and irrelevant text sets, their related probability is calculated and represented as the probability: $O(R) = P(R) / (1 - P(R))$ (R expresses that "the text is related", $\neg R$ expresses that "the text is irrelevant"). The superiority of the probabilistic model lies in its many forms, and its use of the strict mathematical theory as its base, and its ability to sort the retrieval results according to the degree of correlation probability. Its retrieval efficiency therefore obviously surpasses the Boolean model .

The search method is divided into three kinds in literature (C. Menzel and M. G. 2001): scan, search, and choose. Other search strategies, such as filtering, search and so on, can be synthesized by the above information search strategies. Three basic information search methods are described as follows:

- Scan: recognize whether to include some characteristics in the objects according to the condition;
- Search: find the location of the objects or the objects themselves fit for the special condition according to certain rules;
- Choose: choose some of the objects according to illegible or accurate condition. Herein, the object refers to various types of documents.

Others may be considered to be the exceptional case or combination of the three kinds of basic information search strategy. For example, the information search may be regarded as scanning after the search, then choosing according to the condition. The inquiry may be regarded as choosing according to the condition after the search.

Peer-to-Peer (P2P) has very good prospects regarding sharing the distributive files, but the present P2P system still lacks the effective information search mechanism. Literature (James Pitkow, etc. 2002) proposes one kind of P2P information search mechanism driven by the subject, and obtains the global subject through clustering the documents in the node, and then organizes the nodes with similar subject together to compose the subject overlay network. When P2P network searches the information, it searches according to the route related to the subject to improve the search efficiency. Moreover, it indicates through the simulation experiment in Chord: the P2P information search engineer driven by the subject can decrease the average network bandwidth and search path length to improve the success ratio of the search.

Literature (Feng Chen and Yunsong Zou 2004) describes how to finish the task of searching information in the main computer by the agent. Because of its presenting search characteristic under the condition of the computer network, the search process is regarded as a three-dimensional space, which is composed of the condition, the method, the information source. It offers one kind of the information search strategy based on the agent, and finally uses AgentSpeak(L) to state the strategy and the planning process.

At present, the development direction of the information search is to combine with Web in the research of the information search algorithm. One tendency of the development of Web information search is to provide the individualized Web search through matching the content of homepages with the user's interest model. The construction idea of the user's interest model used in OBIWAN system in literature (S. Gauch and A. P. 1999) is: the nodes are divided according to the grade, and each node with weight represents the interest degree to this theme, and the node is stored as a keyword weight vector(the weight based on TF-IDF measure method), and the interest model is stored in the client end ,and uses the interest model to filter and arrange the search results. The system developed in Literature (Fang Liu, etc. 2004) tries to eliminate different meanings of the searched terminology before submission to the search engineer. Its method is to combine each terminology and the little sort set, and extends the user's search with the terminology of other special sorts before submitting to the search engineer. Outride system in Literature (James Pitkow, etc. 2002) acts as the mediate role between the user and the search engineer, its inquiry revision is completed based on the user's interest model when inputting the search, and its returned results are filtered and realigned before displaying to the users.

Another tendency of information searches facing Web is to provide the search service by means of the Web Service, and there are some gate websites providing the corresponding Web search service on Internet. For example, Google is an important

Web site, which provides Web service to the public, and allows the application program to use functions such as the search and spelling check. More importantly it can use the gigantic Google resource library. Google provides several interfaces of Web Services to the public at large in the form of Web APIs service., That means that the software developer can use different developmental tools (such as: Java, Perl or Visual Studio.NET) to call for Web APIs service so as to realize its own application services in different developmental environments.

The research of Agent cooperation based search model begins with the research of the cooperation theory. Malone and Crowson discover that the cooperation between the autonomous entities is a universal problem. In many research areas there are some relevant works in progress, such as computer science, sociology, politics, management science, system science, economics, linguistics, psychics and so on. Now, the special research frame, i.e. the cooperation theory to research the cooperation problem between the autonomous entities has been established. It has been almost accepted that cooperation is to manage the dependent relations between each kind of activities. Thus, it is also to perform cooperation that the autonomous entity manages the dependent relations between each kind of activities.

The general property of the Agent cooperation in the multi-Agent system is discussed, including the MAS structure and organization, the Agent coordination, coherent behavior, coordination, consultation, multi-Agent planning, interaction, and so on.

In recent years, the semantic Web technology is introduced into the Agent cooperation process, in which the famous research projects includes OAA series (T. R. Payne, R.Singh. and K. Sycara 2002) of SRI（Stanford Research Institute）in the AI center; the

RETSINA project and the related work (Baisong Liu 2005; Katia Sycara etc. 2003) developed by the Software Agents Lab led by Sycara of Robotics Institute in Carnegie Mellon University; <I-N-OVA> model and O-P3 project and related application(Austin Tate, J. M., Bradshaw, Andrzej Uszok 2004; Austin Tate,J. D.,J.L., Alex Nixon 2002)proposed by Artificial Intelligence Applications Institute(AIAI) in Edinburgh University; SWF project organized by Digital Enterprise Research Institute(DERI)(Michael Stollberg, U.K., Peter Zugmann, Reinhold Herzog 2003; S, M.S.T. 2005). These projects have unique characteristics and representation.

OAA (The Open Agent Architecture) is the framework developed by AI center in SRI, which is used to construct multi-Agent systems. Its latest edition is OAA2.3.1 published on 11.4, 2005. OAA provides the technology and method to develop new Agent, package heterogeneous system and reuse the existent Agents, and can create dynamic Agent team agilely to cooperate. There are several different kinds of Agent in the OAA system, where Facilitator is the core of the OAA structure, and is responsible to coordinate the communication and cooperative problem with other Agents in the system. The most remarkable characteristic of OAA is that it introduces the semantic technology into Agent cooperation. Each Agent registers its capability in Facilitators; the Agent requesting service constructs the goal and transfers it to Facilitators. Facilitators manage the cooperation according to the semantic illustration about its goal and capability, to meet the requester.

RETSINA (Reusable Environment for Task-Structured Intelligent Networked Agents) is an open multi-Agent system which supports the communication between heterogeneous Agents. Like OAA, there are several different kinds of Agent in RESTINA system, in which the middle Agent (Middle Agents) provides semantic matching services between the service requester and the provider. RESTINA system has made the forward-looking exploration in the analysis of the basic establishment needed in the communication between open heterogeneous Agents, and the very valuable achievement for its future research.RCAL (RETSINA Calendar Agent) (T. R. Payne, R. Singh and K. Sycara 2002; Terry R, P. R. S., Katia Sycara 2002) uses the RETSINA technology to develop the calendar arrangement Agents running on the semantic Web. It uses the knowledge of semantic Web to provide services for the users directly, and the semantic capability has already become a part of this Agent. This technology has also validated that the present Web ontology language and is able to be used by Agents directly.

SWF is a semantic Web Agent platform developed by the DERI organization. The complex and fine technical framework defined in SWF includes: define a suitable proxy model to reflect the real world entity of the cooperation; confirm the cooperation mechanism (D. Garlan and D. Le Métaye eds. 1997); designs Agent for its cooperators and the resource mechanism (Mingkai Dong, Zhongchi Shi 2004) applied in the automatic cooperation execution; take the ontology as the bottom data model of semantic interoperability on Web. The conceptual model and technical framework of Agent independent cooperation on semantic Web proposed in SWF is a significant exploration about the Agent cooperation technology on semantic Web.

In the literature (Integrating Agents, Ontologies, and Semantic Web Services for Collaboration on the Semantic Web) (Terry R., P. R. S., Katia Sycara 2002), the foundational establishment supported by the automatic cooperation of the semantic Web is: Agent will become an electronic proxy of the real world entity as one kind of cooperation of interactive way to attain their owner's goal; Agent will be able to visit and exchange labeled information based on the semantic on Web, and use Web Services as one kind of computing tools.

In the research process of Agent cooperation, there are many process ontologies used to represent the cooperation process between various kinds of Agent united with the ontology technology. PSL (Process Specification Language) (C. Menzel, and M. G. 2001; C. Schlenoff, M. Gruninger and M. Ciocoiu, 1999), as an international standard for the process exchange information between the manufacture systems, describes the relations between the concepts, such as the activity, the activity instance, the time and the object and so on. PSL, as the basic standard of the process, has been expanded or mended by other many process ontologies. For example, the process ontology SPAR(A. Tate, 2003) (Sharable Plan and Activity Representation) describes the agent plan problem as the basic concepts, such as the space, the time, the agent, the activity, the inference, the planning and so on. The semantic Web service technology extends the influence of the research of the process ontology. Web service is a new functional level of the current Web, and its goal is to achieve the seamless integration of distributional modules. The semantic Web service labels its various aspects with clear semantics understandable to machines to realize automatic search, combination and the use of services. It has already exerted huge influence on the application in the electronic commerce and the enterprise integration. Because it can realize dynamic cooperation between different systems and organizations, it also promotes the academic and the industrial world to research the dynamic knowledge of the activities. At present, there have been two important research ways: One is OWL-S (Ontology Web Service Language), which mainly provides precise semantic description of services to realize the automatic discovery, call, composition, interoperability and monitoring of Web services. Another is WSMO (Web Service Modeling Ontology), which mainly includes four modules: the ontology, the Web service, the goal and the mediator. Herein, the ontology provides the concepts and relations of describing the domain, and the goal represents the benefit of the requester party. The Web service provides the semantics description of the functional and non-functional attributes, and the mediator is taken as the connectors of the Web service and the goal. The common thing between OWL-S and the WSMO is to provide the semantic description of the service, and the difference is that in OWL-S, the service is loose, and the service profile describes the functional and non-functional attributes. It provides much more descriptive service modes, and the relation among the services. The service profile is one-to-many, so, the service may simultaneously have many service profiles. But in WSMO, the service is described from the view of the call and composition, and the service must have a certain goal, the service is organized and searched through the goal. So WSMO is more suitable to the combination and call of services, but is not as rich as OWL-S in the descriptive service mode.

Recently, the new research on the combination of the semantic technology and Agent emerges in China. For example, Professor Shi Zhongzhi and others of the Chinese Academy of Science propose one kind of Agent model based on dynamic description logic (DDL) (Jiewen Luo, Zhongzhi Shi, Maoguang Wang and He Huang 2005), and use the unified formal framework of the static and dynamic knowledge representation and inference provided by the dynamic knowledge logic to depict the Agent mental state. Based on this, they propose a multi-Agent cooperation model (Wenjie Luo, Zhongchi Shi 2006; Lei Chen 2004) with DDL.

2.4 Methods of Knowledge Discovery from Databases

Knowledge discovery is defined as “the non-trivial extraction of implicit, unknown, and potentially useful information from data” (U. M. Fayyad, etc. 1996). In (W. J. Frawley

1991), a clear distinction between data mining and knowledge discovery is drawn. Under their conventions, the knowledge discovery process takes the raw results from data mining (the process of extracting trends or patterns from data) and carefully and accurately transforms them into useful and understandable information. This information is not typically retrievable by standard techniques but is uncovered through the use of AI techniques.

Although there are many approaches to KDD, six common and essential elements qualify each as a knowledge discovery technique. The following are basic features that all KDD techniques share:

- All approaches deal with large amounts of data
- Efficiency is required due to the large volume of data
- Accuracy is an essential element
- All require the use of a high-level language
- All approaches use some form of automated learning
- All produce some interesting results

Large amounts of data are required to provide sufficient information to derive additional knowledge. Since large amounts of data are required, processing efficiency is essential. Accuracy is required to assure that discovered knowledge is valid. The results should be presented in a manner that is understandable by humans. One of the major premises of KDD is that the knowledge is discovered using intelligent learning techniques that sift through the data in an automated process. For this technique to be considered useful in terms of knowledge discovery, the discovered knowledge must be interesting; that is, it must have potential value to the user.

Learning algorithms are an integral part of KDD. Learning techniques may be supervised or unsupervised. Supervised learning techniques typically enjoy a better success rate as defined in terms of usefulness of discovered knowledge. Generally speaking, learning algorithms are complex and considered the hardest part of any KDD technique (R. J. Brachman and T. Anand 1996).

Machine discovery is one of the earliest fields that has contributed to KDD (W. Buntine 1996). While machine discovery relies solely on an autonomous approach to information discovery, KDD typically combines automated approaches with human interaction to assure accurate, useful and understandable results.

There are many different approaches that are classified as KDD techniques. There are quantitative approaches, such as the probabilistic and statistical approaches. There are approaches that utilize visualization techniques. There are classification approaches such as Bayesian classification, inductive logic, data cleaning/pattern discovery, and decision tree analysis. Other approaches include deviation and trend analysis, genetic algorithms, neural networks, and hybrid approaches that combine two or more techniques.

Because of the ways that these techniques can be used and combined, there is a lack of agreement on how they techniques should be categorized. For example, the Bayesian approach may be logically grouped with probabilistic approaches, classification approaches, or visualization approaches. For the sake of organization, each approach described here is included in the group that it seemed to fit best. However, this selection is not intended to imply a strict categorization.

- Probabilistic Approach

This family of KDD techniques utilizes graphical representation models to compare different knowledge representations. These models are based on probabilities and data independencies. They are useful for applications involving uncertainty and applications structured such that a probability may be assigned to each “outcome” or bit of discovered knowledge. Probabilistic techniques may be used in diagnostic systems and in planning and control systems (W. Buntine 1996). Automated probabilistic tools are available both commercially and in the public domain.

- Statistical Approach

The statistical approach uses rule discovery and is based on data relationships. An ``inductive learning algorithm can automatically select useful attributes to construct rules from a database with many relations" (C. N. Hsu and C. A. Knoblock 1996). This type of induction is used to generalize patterns in the data and to construct rules from the noted patterns. Online analytical processing (OLAP) is an example of a statistically-oriented approach. Automated statistical tools are available both commercially and in the public domain.

An example of a statistical application is determining what ratio of all transactions in a sales database are cash sales. The system would note that of all the transactions in the database only 60% are cash sales. Therefore, the system may accurately conclude that 40% are collectibles.

- Classification Approach

Classification is probably the oldest and most widely-used of all the KDD approaches (J. R. Quinlan 1993). This approach groups data according to similarities or classes. There are many types of classification techniques and numerous automated tools available.

The Bayesian Approach to KDD "is a graphical model that uses directed arcs exclusively to form a directed acyclic graph" (W. Buntine 1996). Although the Bayesian approach uses probabilities and a graphical means of representation, it is also considered a type of classification.

Bayesian networks are typically used when the uncertainty associated with an outcome can be expressed in terms of a probability. This approach relies on encoded domain knowledge and has been used for diagnostic systems. Other pattern recognition applications, including the Hidden Markov Model, can be modeled using a Bayesian approach (W. Buntine 1996). Automated tools are available both commercially and in the public domain.

Pattern Discovery and Data Cleaning is another type of classification that systematically reduces a large database to a few pertinent and informative records (U. M. Fayyad and P. Smyth 1996). If redundant and uninteresting data is eliminated, the task of discovering patterns in the data is simplified. This approach works on the premise of the old adage, "less is more". The pattern discovery and data cleaning techniques are useful for reducing enormous volumes of application data, such as those encountered when analyzing automated sensor recordings. Once the sensor readings are reduced to a manageable size using a data cleaning technique, the patterns in the data may be more easily recognized. Automated tools using these techniques are available both commercially and in the public domain.

The Decision Tree Approach uses production rules, builds a directed cyclical graph based on data premises, and classifies data according to its attributes. This method requires that data classes are discrete and predefined (J. R. Quinlan 1993). According to (W. J. Frawley, etc. 1991), the primary use of this approach is for predictive models that may be appropriate for either classification or regression techniques. Tools for decision tree analysis are available commercially and in the public domain.

The genetic algorithm is not only one kind of computation model of the natural evolution system, but also one adaptability search method of the universal solution of the optimization problem. From an overall view, the genetic algorithm is the earliest research direction and area with the strongest influence, widespread application in the evolution algorithm. It has several unique performances:

- When solving the problem, the genetic algorithm must first choose the encoding mode. Its direct processed object is the parameter code set, but not the problem parameter itself. And the search process isn't restricted by the continuity of the optimization functions, or the existent requirement of the differential coefficient of the optimization functions. Through the reorganization of the fine chromosome gene, the genetic algorithm can process effectively the very complex solution of the optimization function in the tradition. The genetic algorithm takes the goal function value directly as the search information.

- If the genetic algorithm handles the individuals of the colony with n scale in each generation, it has very high parallelism. Thus, it has the remarkable search efficiency, and may reduce the expensive cost.

- The genetic algorithm has very strong robustness. It constringes to the optimal or satisfactory solution with much probability when the problems to be solved are non-continuous, and multi-peak with the yawp. So, it has the capability of gaining good global optimal solution.

Since the basic idea of the genetic algorithm is simple, and the running way and the steps for realization are standardized, it is convenient for concrete use.

2.5 Patterns of Knowledge Management Systems

With the development of the network information technology and knowledge economy, more and more enterprises in their service production process realize that knowledge is the main resource, besides manpower and capital. The enterprise needs to recognize, capture, develop, decompose, use, store and share its knowledge effectively to improve its competition and knowledge innovation ability. Through the effort of many researchers, there are a lot of knowledge management systems that have been developed.

The layer model based knowledge management system is the three-layer pattern (R. H. Sprague 1980) proposed by Sprague in 1980, like Figure 2.1. The first layer consists of knowledge management tools including the expert system language (such as: Oracle PIPSQL) or programming language (such as: C++) composed of its basic components. The second layer is the knowledge management system builder (such as: Lotus Notes), which is used to establish various special knowledge management systems. The third and final layer is the established special knowledge management system. This kind of model points out the sequent phases of its development, and has important significance to manage its developmental process and construct its developmental tools quickly.

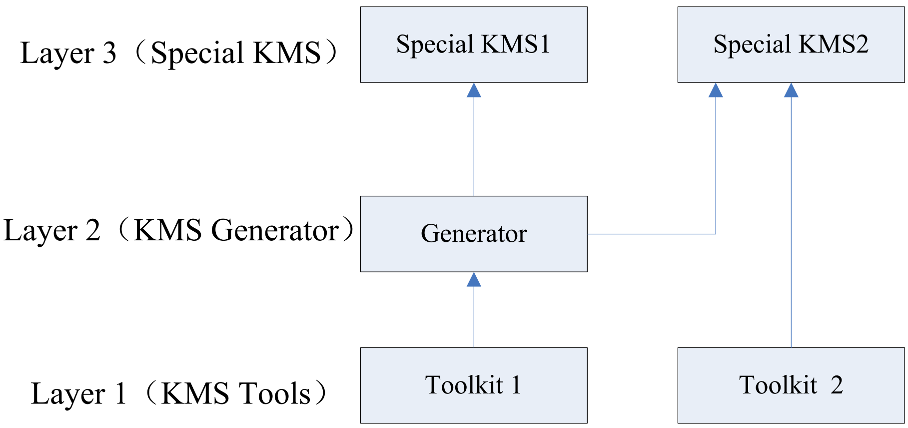


Figure 2.1 Layer Model Based Knowledge Management System

The knowledge management system based on the general system frame regards itself as a kind of information system, and can be researched through its input, its process and its output, shown as figure 2.2. Its advantages are: simple structure, wide coverage, and the inclusion of all main components. Is principal disadvantage is that it cannot give prominence to the importance of the knowledge base or the essential features of peoples' knowledge transition.

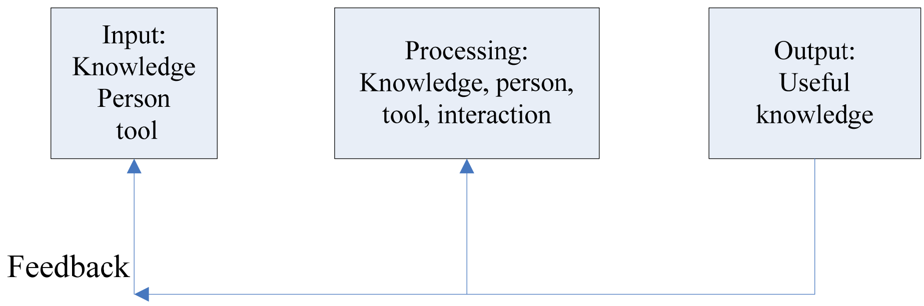


Figure 2.2 General System Framework Based Knowledge Management System

In 1997, Ruggles built a kind of knowledge management system (R. Ruggles 1997) according to the knowledge life cycle: knowledge acquirement →knowledge coding and storage →knowledge transmission →knowledge utilization→ the creation of new knowledge, shown as Figure 2.3. Through building and using this kind of knowledge management system, people can investigate its influence at each stage. Furthermore Ruggles, Lee, Zhu Xiao Feng, and Liu Hao have successfully applied the knowledge management system model based on the knowledge life cycle to the design and development of the knowledge management system. The knowledge management system set up by Lee (Peter Meso, Robert Smith 2000) is made up of four parts, including knowledge capture →knowledge development →knowledge sharing →knowledge utilization; The knowledge management system built up by Zhu Xiao Feng (Thomas H. Davenport and Laurence Prusak 1998) comprises four modules, including knowledge production, knowledge mining, knowledge reorganization and knowledge utilization; The knowledge management system constructed by Liu Hao (Hao Liu, Cuohu Qin 2001) is made up of four parts, including the knowledge production system, knowledge capture and structure system, knowledge distribution system and knowledge application system. This kind of knowledge management system based on the knowledge lifecycle has the advantage of simplifying the concepts, but cannot provide ample internality or indicate which aspects of the research are lacking for the system.

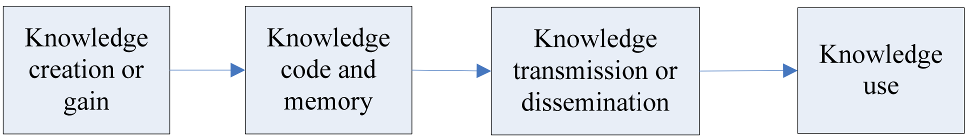


Figure 2.3 Lifecycle Based Knowledge Management System

Brent Gallupe (2001) proposed the knowledge management system based on the knowledge practice frame. It takes the knowledge management practice of the organization as its center, and points out the type and way supporting the practices and activities. It investigates the knowledge management practice in two parts, including the process of identifying problems and the classification of the problems to be solved, which produces four kinds of practice, shown as Figure 2.4. The first practical activity is to confirm new problems through inspiring the creative power and producing new knowledge. The second practice is to solve new problems through creating and storing the knowledge related to these problems. The third practice is to use the knowledge concerned to handle the solved problems (such as the resource library). The fourth practice involves delivering the new information to the individual to support the cognition of the original problems.

		Question Classification	
		NewOnly	Past Solved
Question Process	Question Understanding	Creation opportunity（1）	Monitoring and Training（4）
	Question Answer	Question Answer（2）	Knowledge Gain (3)

Figure 2.4　Knowledge Practice Framework Based Knowledge Management System

The Distributed Knowledge Management Systems proposed by Joseph M. Firestone in Literature (R.H. Sprague 1980) can apply the data exchange system (DES), the online transaction processing (OLTP), the batch processing and so on to the enterprise management systems.

G. DeSanctis and R. B. Gallupe (1985) proposed the resource based knowledge management system (Haiqun Ma 2000) composed of the knowledge library sub system, the user interface sub system, the group support or knowledge transition sub system and the user/knowledge builder sub system; and the resource knowledge management system developed by Peter Meso and Robert Smith is composed of knowledge, the technology base, the organization base, human resource and culture. The resource knowledge management system constructed by Brent J, Bowman in Literature (Ke Li, Qinghua Bai 2001) has eleven modules, such as the outer information, inter information, the resource library creator and management tools, the knowledge library, the metadata, the knowledge retrieval tool, the network gateway interface, the Worldwide Web and the Internet, format automatic applications, data warehouse/data mining and extraction tools, and enterprise applications.

Li Ke proposed the XML based knowledge management system model which is also based on the analysis of the enterprise knowledge management demand and flow. It has eight parts, such as the agent, multi-document transition interface, the content management, the knowledge publishing and sharing, the workflow coordination, the decision-making support, XML and database interface and the knowledge management database. It has the following advantages compared with other knowledge management systems: uniform and good document structure; easy to storage uniformly, convenient for classification management; Web browser; the knowledge publishing and sharing on Web through XML; the quick search based on meta data, high retrieval efficiency; transmission between numerous systems; technical merits and the direction of future development (Xiaofeng Zhu and Songzhen Lang 1999).

The traditional knowledge management can be realized on a single computer, or within a small scope of the close network. Along with the network development and Internet popularization, all traditional applications move to the platform of Web computation pattern, and the traditional two layers i.e. C/S computation also change into three layers. Some organizations or enterprises establish the information systems related to various kinds of services in the internal operation flow, and may expand very easily to the external Internet environment (Hao Liu and Cuohu Qin 2001). In the network information society, the construction of knowledge management needs the load bearing of the concrete network, but the knowledge management system on the enterprise information platform uses Web as its base, and may combine well with the service process. The support of the exchange between information and knowledge is a long-term fixed goal of the information technology. As the development of Web itself, the on-line information exponential growth on Intranet and Web makes it hard to realize the exchange between information and knowledge or attain the anticipatory target. The Web technology and the knowledge management may promote each other's development. On the one hand, the Web management of various resources needs the knowledge management, specifically the support of the knowledge engineering technology to overcome difficulties which Web faces now, to have Web release its potential fully. For example, the management of various kinds of documents on Web requires one of the important knowledge management technologies i.e. the document management technology. On the other hand, the knowledge management on Web platform faces new opportunity and challenge. The knowledge management based on the existing Web computation environment has gradually become mature, but with the development of the semantic Web there are some problems of how to share the terminology concepts in a distributed organization, and to reduce the knowledge management load of organizations. With the further development of the semantic Web technology, it is the inevitable tendency to use the semantic Web technology in the knowledge management system. The knowledge management system based on the semantic Web computing environment can fully use the scientific, structural and semantic characteristics provided by the semantic Web, use the KE technology and method to provide accurate support to the discovery and location of the Web resources, the data mining based on Web and knowledge discovery in database (Thomas R. Gruber 1993). As the semantic Web technology gradually becomes mature, more and more knowledge management systems will use this new technology.

At present, the ontology technology is used one after another in the knowledge management system to solve the inconsistent semantic problem. For example, the literature (Bin Shen 2004) has constructed an ontology based knowledge management system and realized the share and reuse of the enterprise knowledge with the ontology. The literature (Baisong Liu 2005) proposes one kind of ontology-based open knowledge management method (OOKMM), and constructs a knowledge management system with three layers. It takes opening as its core, and researches into the opening realization of the knowledge management system in the three-dimension and understanding sharing establishment, the knowledge open integration, the automatic construction of the ontology knowledge, and many-layer systematization. The purpose of the understanding and sharing establishment is to provide the modeling method, language and tool of the information content, and help the enterprise design and maintain the share ontology which is agreed upon by knowledge engineers and comprehended by computers for its open model, which offers a common comprehension base of the current information to the knowledge engineers and computers. The open integration of the knowledge relies on the assistant tools provided by the ontology integration mechanism, which forms the additional meta knowledge conveniently when helping knowledge engineers establish the information body according to the assigned form. The ontology based knowledge management system proposed in the literature (Xin Huang, Xiaojuan Xu and Guoliang Xu 2005) applies the semantic description of the corresponding resource to realize the sharing of the information and resource with the domain knowledge. At the same time, the ontology is a high share conception model and represented in the formal way. So, it can be processed directly by the computer system, and has the widespread application in the coordination of the many application systems.

Now, the ontology based knowledge management system in the semantic Web environment has greatly developed in research. However, semantic web technology is a new technology after all, and many aspects are not perfect. Although it can solve some traditional problems by applying the semantic web technology in the knowledge management system, the introduction of new technology will inevitably cause some new problems, one of which is that there isn't a united developmental framework of the ontology knowledge management system in semantic environment. So, its framework research plays a very important role in its whole lifecycle (including information acquisition, information search, system maintenance and so on), which is also very important to its development efficiency and integration in the standard semantic Web (B.Motik, A. Maedche and R.Volz 2002).

Simply speaking, no matter what kind of pattern prevails, it should generally realize the following functions: knowledge representation and management; knowledge accumulation and transmission; knowledge excavation and regeneration; knowledge utilization and appraisal. A successful knowledge management system should also be an integrated multi-functions system which can support all main knowledge management and treatment, including knowledge gain, knowledge organization, knowledge classification and understanding, debugging and edition, search and retrieval, knowledge transmission, dissemination and sharing and so on. At the same time, in view of the enterprise demand for continual increasing development, the knowledge management system with a strong expansibility and good versatility should also have the following functions: the expandable data gain and import; knowledge classification mechanism; strong and flexible distributional search; classification jurisdiction management; several kinds of issue ways; comprehensive knowledge exhibition;

real-time knowledge accommodation and coordination work ability.

To some extent, the knowledge management system is also a kind of information management system. For any information management system, the information search performance is an essential technical specification. So, the research of the improvement of the engines efficiency is the basic task in the information management system. With the development of the semantic Web technology, the traditional search engine is facing a new opportunity and challenge. At present, the information on Web increases every day, and the Internet is facing the information explosion. So, how to design the search method with high performance is most important to improve the knowledge management system performance. As the information on Internet becomes more and more structural, the search engine may locate the information accurately. At the same time, the search engine needs some new methods to mine the concealed knowledge and thus improve the information search ratio, the search efficiency and semantic accuracy.
The ontology is the formal standard explanation about the share concept model, which aims to capture the knowledge in the related domain, provide the common comprehension about this domain, determine the glossary which this domain universally approves, and give the clear definition of these glossaries and the relations from the formal patterns at different levels.

The knowledge management is the multi-disciplinary application domain. It involves the management science, sociology, the document management, the human engineering, CSCW, knowledge engineering and so on. The information technology is the base of the knowledge management technology, and the knowledge management is the synthetic utilization of information technology. These technologies include many aspects, such as distributional store management, aggregated system, Internet/Intranet, database, word processing, electronic forms, group decisions and so on. The most important of them will contain the document management, the group, the text excavation and retrieval and the enterprise gateway. At present the main activities of KM have the document management system, capability management system, the experience knowledge library system and so on.

2.6 Knowledge management issue in real estate enterprises

The real estate enterprises are typical knowledge-intensive enterprises. With it increasingly being recognized that the knowledge management issue play a very important role during the process of enhancing enterprises' core competence, the knowledge management issue in real estate enterprises has started to draw the attentions from some researchers and scholars interested in the real estate field. Although not so many as other fields, there are some literatures in the aspect during recently years. These literatures mainly focused on the real estate body of knowledge, how to organize the real estate knowledge, the knowledge transfer methods in the real estate enterprises, the model and key technology of knowledge management system in real estate enterprises, the establishment and cultivation of the corporate culture of carrying out knowledge management in real estate enterprises, etc.

The real estate body of knowledge is what to be managed, be created, be coded, be stored and be learned during the process of knowledge management in real estate enterprise. Defining a body of knowledge in real estate is extremely difficult, since there appears to be no clear cut consensus on the boundary lines of the discipline, either academic or professional(Roy T. Black and Joseph S. Rabianski, 2003).Because of its importance, some researcher attempt to establish the real estate body of knowledge which can be widely accepted. From the global prospective, through analyzing the survey sent to real estate academics and professionals around the world, Roy T. Black an Joseph S. Rabianski (2003) defined a body of knowledge for real estate. Thirty six topics relevant to real estate production and sale are ranked overall and compared for differences by the geographic location of the respondents and the professional area of the respondents. Among the topics, risk and return analysis ranks no 1, and following it is discounted cash flow analysis, and then property market modeling-demand and supply analysis-housing, retail, office, etc is listed in the third position.

How to organize a variety of real estate knowledge is the second issue to be solved during carrying out knowledge management in real estate enterprises. MM Kwan and P Balasubramanian(2003) present the process-oriented knowledge management strategy. In the process-oriented knowledge management, the knowledge repository is organized by the organizational process in the form of process designs, case histories and lessons learned from past experiences. Shen,Y. (2007) discusses the concepts and classifications of knowledge map, and analyzes the characteristics of knowledge map in the real estate enterprise. From the actual needs of real estate enterprises, he brings forward the knowledge management model framework structure that based on knowledge map set. He also designs a prototype system of knowledge map in real estate business.

The creating process of knowledge in the real estate enterprises is of their own characteristics. According to Nonaka and Takeuchi (1995), Knowledge is created through a repeated four-phase process – explicit and tacit knowledge interchange through the so-called SECI process:(1) Socialisation.(2) Externalisation.(3) Combination.(4) Internalisation. Based on the above process, Patrick S.W. Fong (2009) proposed a model suggesting a process of knowledge development with an entrepreneurial environment for real estate enterprises. His research work found that the education level of staff play a role, with a staff –related "ambience" of knowledge development where the "culture" depend on the education level of staff and their interactive capabilities. Therefore, the three extracted factors of innovativeness, risk taking and proactiveness are managed differently depending on the educational level. Wu, J (2008) reveals the mechanism of knowledge transfer in R&D team of real estate enterprises, and models team knowledge transfer from the perspective of the factors that influenced it.

Knowledge management system is the tool facilitating implement knowledge management. How to establish the knowledge management system in real estate enterprises very interested not only the academics but also the professionals. Razali, MN and Manaf, Z (2005) developed a knowledge management systems model used as a basis guideline to build successful knowledge based system, that support all activities in real estate businesses. This model consists of five key elements: contents, technology, people, process and context that are related to each other and needed for the development of knowledge management systems. Xu, L.(2006) gave some thought about knowledge management system in real estate enterprises. He thought that knowledge management system in real estate enterprises is a system that focusing on the creation, collection, codification and diffusion of the knowledge within real estate enterprises. The KMS should include a language layer, a conceptual layer and an interface layer. The key technologies of establishing KMS include the network technology, object-oriented technology and the data base technology, the language of software development, the collaborative engineering technology, the knowledge push technology, the knowledge base and knowledge mining technology. Lin, L (2006) discussed the knowledge resources of real-estate developing enterprises, and advanced a management platform and an evaluating index system based on fuzzy theory.

The course of real estate marketing is a complex and profound knowledge activity. Xu, D (2006) divided the course of real estate marketing into these key factors: market orientation, product orientation, price strategy, the scheme development, the market promotion and management of sales force, etc. and used theories and methods of knowledge management to make idiographic analysis and study on each factor.

Li, JZ (2006) thinks that whether knowledge can become the knowledge advantage of the real estate enterprise, and hence be transformed as its core competence, is predominantly determined by the establishment of its corporate culture based on knowledge management. The corporate culture is the real estate enterprise's main inner-driving power to obtain its core competence. That is to improve the performance of real estate enterprises on the basis of the capability promotion of their staff by learning and the satisfaction of customers by respecting them. The enterprises should regard trust as the base, the human-centered management as the means, motivating system as tools, organizational learning as carriers.

2.7 Conclusion

From the literature above, we can draw the following conclusion:

1. With knowledge management being increasingly active, the knowledge management issue in the real estate industry is drawing the attention from more and more people. At present, the achievement in real estate knowledge management mainly focused on the real estate body of knowledge, how to organize the real estate knowledge, the knowledge transfer methods in the real estate enterprises, the model and key technology of knowledge management system in real estate enterprises, the establishment and cultivation of the corporate culture of carrying out knowledge management in real estate enterprises, etc. The research in real estate knowledge acquiring is very rare, so this dissertation can strengthen this aspect research.

2. The ontology technology is commonly recognized to be a very good method for solving the semantic conflict problem. The concept of ontology based knowledge management system has been proposed, but the ontology based real estate knowledge

manage system has not been found. The ontology based knowledge management systems proposed by others did not provided the knowledge acquiring function through sharing and cooperation search between different knowledge management systems like the ontology based real estate knowledge management system given in this dissertation.

3. Because the ontology technology is very general, so when it is applied to the special field, there will be a lot work to do. In this dissertation, the problem of how to create the real estate knowledge ontology must be solved before developing the ontology based real estate knowledge management system.

4. The agent cooperation search method is an effective solution to search information from the internet. The ontology based agent cooperation search method is a promising research direction.

5. The genetic algorithm is an important method of knowledge discovery from a great deal of data stored in databases or data-wares, and is also a general way like ontology technology. When applying it to solve a special problem, there are a lot of special problem to solve. For example, how to design the code rule? How to design the operation processes of selection, crossover and mutation operators?

Chapter 3

REAL ESTATE ONTOLOGY MODELING

3.1 Introduction

This dissertation proposes that to build knowledge management union consisting of a number of cooperating real estate enterprises is one solution to the challenge of acquiring comprehensive knowledge on the real estate market. However, inconsistent semantic understanding is one of the difficulties of realizing the exchange of knowledge among different real estate enterprises. In order to solve the above problem, the knowledge management systems in different real estate enterprises must use a same kind of knowledge representing method. By Compared with other common knowledge representing methods, this dissertation argue that ontology theory is more suitable for its purpose, and also recognized as one successful method for knowledge representing. This chapter and next one will propose an ontology based agent cooperation search model for acquiring the knowledge external to the real estate enterprise. As the foundation of the above model, this chapter will first explore how to build a real estate ontology model.

3.2 Comparison of Common Knowledge Representing Methods

At present, there are mainly four kinds of common knowledge representing methods, i.e. production rule, frame, semantic network and ontology.

Among these methods, the production rule is the earliest knowledge representing method, appearing along with the expert system which used it as the expressing form of knowledge stored in knowledge base. It very facilitates the implement of inference engine, and so is popularly applied. With no doubt, the production rule is very suitable for expressing of the inference knowledge, but not suitable for some other knowledge, such as the conception, the relation between conceptions, the process knowledge. In the real estate industry, there is a lot of knowledge like conceptions, processes and so on. Therefore, this dissertation thinks that it is not suitable for representing the real estate knowledge.

The frame technology is another classical form of knowledge representing. It uses slots and slots' value to describe knowledge. An abstract concept usually owns a lot of slots, some of which have special values. A special object consists of slots with value. The relation between different concepts can be expressed through slots and values. This kind of describing mode is very suitable for expressing some static knowledge.

The semantic network is developed based on the predictive logic technology which is another early form of knowledge representing. It consists of nodes and arcs connecting nodes. The nodes express the physical objects, concepts or status and the arcs are used to describe the relation between different objects or concepts expressed by nodes. The advantages of the semantic network include the following. First, the relations between concepts can be easily expressed and inferred. Second, the attributes can be inherited and inferred. Third, the status and action can be described. Its disadvantages lie in the following. First, the conclusion drown from the semantic network can not be guaranteed to be absolutely true. Second, the semantic explanation to terms depends on the application program. Third, the searching implement needs a very strong organizing principle in the semantic network.

Although the ontology in computer science can be traced back to the 1980s, Neches and his cooperators are considered to be the persons who first defined ontology in the field of artificial intelligence. The ontology uses the strict mathematic formulas to describe the basic concepts and logical relations in the application domain with the terminology and attributes. It generally haves rich and direct-viewing power of expression, and be conveniently comprehended, processed and applied by computers. As the representation way of the data, information, knowledge, it can describe their latent semantic information, and can be taken as the semantic foundation for automatically understanding and interoperability of the heterogeneous software or machine. It can be obtained through abstracting some phenomenon in the objective world. So, its meaning is independent from the concrete environmental state; (Thomas R. Gruber 1993).It is the accurate math description, and can be read by computers. The concepts, relations and restriction are all defined explicitly. What the ontology embodies is the common knowledge, and what it reflects is the common concept set in the related domain. It refers to the team but not the individual. Different researchers have attained common cognition about the ontology: the ontology is the semantic base of the communication (dialog, interoperability, sharing and so on) between the different entities (e.g. person, machine or software system and so on) in the domain (maybe specific domain, or a broader scope). Its core meaning lies in providing one kind of the definite, clear and formal cognition to enable the cooperation between machines.

Given the real estate knowledge is of very much variety and compared with the other knowledge representing methods, this dissertation thought that it is more suitable to apply ontology to express real estate knowledge for communication between different real estate enterprises.

3.3 Ontology Modeling Basic Theory

At present, the ontology is divided into the following 4 kinds: top-level ontology, domain ontology, task ontology and application ontology according to its dependence on the domain. The top-level describes the universal concepts and the relations between concepts, which are irrelevant to concrete application. Other kinds of ontology are its special cases. The domain ontology describes the concepts and relations of some special domain. The task ontology describes the concepts and relations of some special task or behavior. The application ontology describes the concepts and relations between some special domain and task.

The description logic is one kind of knowledge representation language, and a decidable subset of the first-order predicate logic. It is also the hot topic of the knowledge representation domain research. The reference literature "The Description Logics Handbook: Theory, Implementation and Applications" (Franz Baader, etc. 2003) introduces its origin, its rationale and its application in detail.

The basic idea of the description logic originates from the semantic network and the frame. The semantic network proposed by Quillian in 1966 (M. R. Quillian 1967) is used to represent the inference of the knowledge and system, through the cognitive structure like the network shape. This kind of network is formally a directed graph, in which the node represents the concept; the connection arc between the nodes represents the relation between the concepts. As all the concepts are connected to each other through connection arc, the knowledge reasoning of the semantic network is to infer the relation between concepts by seeking the connection way between the nodes. The frame network was proposed by Minsky in 1975. Its basic premise is that the condition, attribute, development process and the correlation of things in the world have a certain regularity which can be represented by the frame. So, the frame is a structural data representing knowledge. The frame composed of the frame name and the slot can represent the knowledge of the related things. It can not only represent each attribute , but also the characteristics, such as the category relation, the variance and so on. Although there are remarkable differences between the semantic network and the frame, they have many common bases in their intuitive cognition and motivation. In fact, they are considered as the network architecture, which represents the individual set and the relations between them.

Although the network based structure is direct and easy to understand, its corresponding inference tool relies on its realization strategy to a great extent without precise semantic description ability. “KL-ONE” (Brachman, R. J. and Schmolze, J. G. 1985) completed the basic step of describing these systems based on the logic to provide the semantics for the network based system, especially the semantic network and frame. It provided the logic base which can be used to interpret the object, the kind (or concept) and the relations. “The tractability of subsumption in Frame-Based Description Languages” (Ronald J. Brachman and Hector J. Levesque 1984) published by Brachman and Levesque which described the tradeoff between the expression power of similar KL-ONE language and its computational complexity, is usually regarded as the origin of description logic research. The goal of this logic modification is to provide reliable, complete semantic inference processes, and its basic idea is that a unitary predicate stands for the individual set, and a binary predicate stands for the relation between the individuals. The frame and the semantic network do not need all mechanisms of the first-order logic most of the time, but only its partial piece. Hence, for the knowledge representation method based on the network structure, the classical inference can be completed through the specific inference technology, but does not need the whole first-order logic theory proof. The inference complexity differs in different pieces of first-order logic.

The first description logic is called "terminological systems", in which terminologies are represented as the class and the relation in the modeling domain. It uses the language to construct the basic terminology. Afterwards, the description logic emphasizes the concept constructors which can represent the knowledge called concept languages. In recent years, researchers' attention has turned to the attribute of the logic system, and so the term "description logic" has come to be widely used. Of the numerous formal knowledge representation methods, description logic has received researchers' special attention for more than ten years. The primary cause lies in the fact that it has decidability; that is, it can guarantee that the inference algorithm can always conclude and return the correct result. It has a clear model-theory mechanism which represents the application domain very suitably through the concept taxonomy to provide useful inference service. According to different constructors, it has many types, such as concrete domain description logic, $\mathcal{SHIQ}$(I. Horrocks and U. Sattler, S. Tobies 2000)、 $\mathcal{SHIN}$(I. Horrocks, U. Sattler and S. Tobies 1999)、$\mathcal{SHIF}$(I. Horrocks, U. Sattler and S. Tobies 2000; I. Horrocks, U. Sattler and S. Tobies1999)、$\mathcal{SHOQ(D)}$ (I. Horrocks and U. Sattler 2001), and so on. We may choose some concrete type of description logic according to the need of the knowledge representation in application.

The knowledge representation system based on description logic provides these tools used to establish, infer and process the knowledge base. Figure 3.1 describes this kind of system structure summarily.

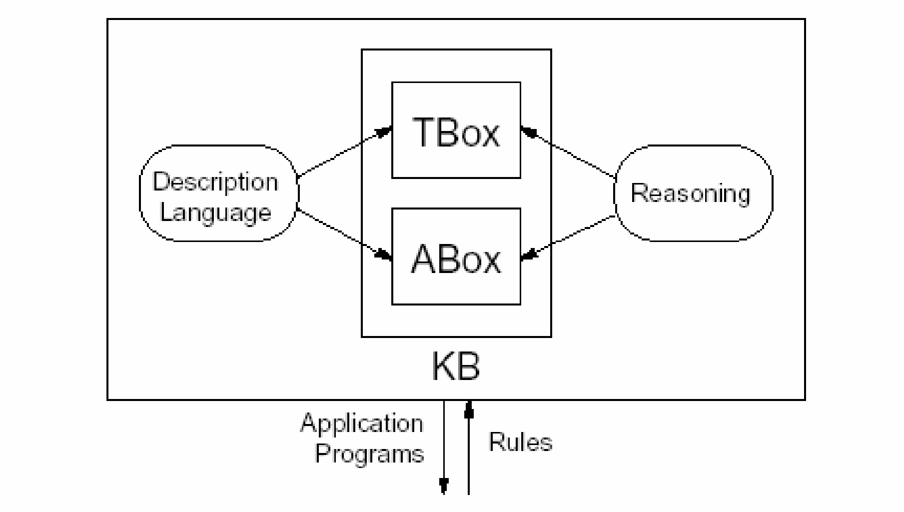


Figure3.1 Knowledge Representing System Architecture Based On Description Lagic

A description logic system contains three basic parts: the structure collection representing the concept and the relation; the knowledge library composed of TBox and ABox, and; the inference mechanism on the knowledge library. The expressive and reasoning power of a description logic system is determined by the choice of element described together with the different hypotheses. The structure collection includes: concepts representing the individual set; the roles representing the binary relationship between individuals, and; the constructors composing the complex concepts and roles. TBox (Terminology part) is a group of the axiom and the assertion representing the domain structure. However, ABox (Assertion part) contains a group of assertions expressing the special individual.

The inference problems on the knowledge base are mainly: the concept satisfactory extent; the concept inclusion examination; the consistency of assertion formula collection; the example examination, and so on. It can determine whether a knowledge base has significance through checking the concept satisfactory extent and the assertion set consistency. It can organize its concepts into a hierarchical structure according to the generality of the terminology through the test of the concepts inclusion. A concept may be regarded as the object set. Thus, through the example test, we can take back the individuals satisfying the inquiry. To use the computer to judge the satisfactory extent in the description logic automatically, Schmidt-Schauβ and Smolka first established Tableau algorithm (Franz Baader, etc. 2003) $\mathcal{ALC}$ based on description logic, which could determine the satisfactory extent of the description logic $\mathcal{ALC}$ concepts in the polynomial time. At present, the Tableau algorithm has been used in each kind of description logic (such as $\mathcal{ALCN}$, $\mathcal{ALCQ}$ etc.), and may also be used in instance checking.

Next, we will begin with the smallest language $\mathcal{AL}$ (attribute language) (Franz Baader,etc. 2003) in description logic to introduce the description logic language. The most basic element in the description logical language is the atomic concept and atomic role, on whose base the complex description can be induced and defined with the concept construct. We use letters A and B to represent the atomic concept, the letter R for the atomic role, and C and D for the concept description.

The $\rightarrow_{\sqcap}$-rule
Condition: $\mathcal{A}$ contains $(C_1 \sqcap C_2)(x)$, but it does not contain both $C_1(x)$ and $C_2(x)$.
Action: $\mathcal{A}' = \mathcal{A} \cup \{C_1(x), C_2(x)\}$.

The $\rightarrow_{\sqcup}$-rule
Condition: $\mathcal{A}$ contains $(C_1 \sqcup C_2)(x)$, but neither $C_1(x)$ nor $C_2(x)$.
Action: $\mathcal{A}' = \mathcal{A} \cup \{C_1(x)\}$, $\mathcal{A}'' = \mathcal{A} \cup \{C_2(x)\}$.

The $\rightarrow_{\exists}$-rule
Condition: $\mathcal{A}$ contains $(\exists R.C)(x)$, but there is no individual name z such that $C(z)$ and $R(x,z)$ are in $\mathcal{A}$.
Action: $\mathcal{A}' = \mathcal{A} \cup \{C(y), R(x,y)\}$ where y is an individual name not occurring in $\mathcal{A}$.

The $\rightarrow_{\forall}$-rule
Condition: $\mathcal{A}$ contains $(\forall R.C)(x)$ and $R(x,y)$, but it does not contain $C(y)$.
Action: $\mathcal{A}' = \mathcal{A} \cup \{C(y)\}$.

The $\rightarrow_{\geq}$-rule
Condition: $\mathcal{A}$ contains $(\geqslant n\, R)(x)$, and there are no individual names $z_1, \ldots, z_n$ such that $R(x, z_i)$ $(1 \leq i \leq n)$ and $z_i \neq z_j$ $(1 \leq i < j \leq n)$ are contained in $\mathcal{A}$.
Action: $\mathcal{A}' = \mathcal{A} \cup \{R(x, y_i) \mid 1 \leq i \leq n\} \cup \{y_i \neq y_j \mid 1 \leq i < j \leq n\}$, where $y_1, \ldots, y_n$ are distinct individual names not occurring in $\mathcal{A}$.

The $\rightarrow_{\leq}$-rule
Condition: $\mathcal{A}$ contains distinct individual names $y_1, \ldots, y_{n+1}$ such that $(\leqslant n\, R)(x)$ and $R(x, y_1), \ldots, R(x, y_{n+1})$ are in $\mathcal{A}$, and $y_i \neq y_j$ is not in $\mathcal{A}$ for some $i \neq j$.
Action: For each pair y_i, y_j such that $i > j$ and $y_i \neq y_j$ is not in $\mathcal{A}$, the ABox $\mathcal{A}_{i,j} = [y_i/y_j]\mathcal{A}$ is obtained from $\mathcal{A}$ by replacing each occurrence of y_i by y_j.

Figure 3.2 Tableau Satisfiability Algorithm

The concept description in the $\mathcal{AL}$ language can be constructed according to the following syntax rules (Franz Baader, etc. 2003):

C，D→A ∣	（Atom concept）
⊤ ∣	（Top concept）
⊥ ∣	（Base concept）
¬A ∣	（Atom negative）
C⊓D ∣	（Intersection）
$\forall R.C$ ∣	(Universal quantifier, value constraint）
$\exists R.$ ∣	（Limit exist quantifier）

To give a vivid example describing the concept in $\mathcal{AL}$ language, we suppose Person and Female be the atomic concepts, so, Person⊓Female and Person⊓¬Female are also the concepts of the $\mathcal{AL}$ language, which mean intuitively that some people are female, some are not. Moreover, if we suppose that hasChild is an atomic role, we can use the concepts $\text{Person} \sqcap \exists hasChild.\text{Person}$ and $\text{Person} \sqcap \forall hasChild.Female$ to represent the people whose children are all girls. Similarly, we can use $\text{Person} \sqcap \forall hasChild.\bot$ to describe the people who have no children.

The concept is interpreted as certain domain subset in $\mathcal{AL}$ language, and the role is the binary relation in this domain. Formally, an interpretation is I = ($\Delta^I, \cdot^I$), in which interpretation domain Δ^I is no empty set, and $\cdot^I$ is the interpretation function which maps each atomic concept A to a set $A^I \subseteq \Delta^I$ and maps each role R to a binary relation $R^I \subseteq \Delta^I \times \Delta^I$.

$$
\begin{aligned}
&\top^I = \Delta^I \\
&\bot^I = \phi \\
&(\neg A)^I = \Delta^I \setminus A^I \\
&(C \sqcap D)^I = C^I \cap D^I \\
&(\forall R.C)^I = \{a \in \Delta^I \mid \forall b.(a,b) \in R^I \rightarrow b \in C^I\} \\
&(\exists R.T)^I = \{a \in \Delta^I \mid \exists b.(a,b) \in R^I\}
\end{aligned}
$$

If all the interpretations I have $C^I = D^I$, we say that these two concepts C, D are equal, writen as $C \equiv D$. For example, returning to the semantic definition of the concept, it may be easy to prove $\forall hasChild.Female \sqcap \forall hasChild.Student$ and $\forall hasChild.(Female \sqcap Student)$ are equal.

The description language is different due to its different constructors. If we add other constructors to language AL, we can gain a new language with strong expressive power. For example: The union of the concepts (represented by U) can be written as $C \sqcup D$, and interpreted as $(C \sqcup D)^I = C^I \cup D^I$.

The complete existence variable (represented by *E*) is written as $\exists R.C$, and interpreted as $(\exists R.C)^I = \{a \in \Delta^I \mid \exists b.(a,b) \in R^I \wedge b \in C^I\}$. Here it should be noted that that $\exists R.C$ allows any concept to appear in the scope of the existence variable, which is different from $\exists R.T$.

The quantity restriction (represented by *N*) is written as $\geq nR$ (restricted at least) and $\leq nR$ (restricted at most). Here n is the non-negative integer, which may be interpreted as $(\geq nR)^I = \{a \in \Delta^I \| \{b | (a,b) \in R^I\} | \geq n\}$ and $(\leq nR)^I = \{a \in \Delta^I \| \{b | (a,b) \in R^I\} | \leq n\}$ respectively.

The negative of any concept (represented as C, complement) is written as $\neg C$, and interpreted as $(\neg C)^I = \Delta^I \setminus C^I$.For example, we can describe those people who have at most one child or at least three children and one of them is a girl with additive constructors.

$$Person \sqcap (\leq 1 hasChild \sqcup (\geq 3 hasChild \sqcap \exists hasChild.Female))$$

We can produce a special *AL* language by expanding it with random subsets of the above constructors. We use the string style to represent the corresponding constructors in the *ALNC*. For example, *ALEN* is the expansion of the *AL* language which contains the complete existence variable and quantity restriction. From the semantic viewpoint, however, all languages are similar, and the semantics have stipulated two equalities compulsively, $C \sqcup D \equiv \neg(\neg C \sqcap \neg D)$ and $\exists R.C \equiv \neg \forall R.\neg C$.

So we cannot differentiate the $\mathcal{AL}$ language with the negative, one with the conjunction as well as one with the complete existential quantifier. Similarly, we may use $\mathcal{C}$ to replace $\mathcal{UE}$ in the language name. For example, we may use $\mathcal{ALC}$ to replace $\mathcal{ALUE}$ or $\mathcal{ALCN}$ to replace $\mathcal{ALUEN}$. Table 2.1 below shows $\mathcal{ALC}$ grammar and the semantics. The modules in the description logic constructed according to the basic syntax include the atomic concept (unitary predicate), unitary role (binary predicate) and individual (constant). The DLs user can construct complex concepts and the role description with its constructors provided by DLs.

When establishing the knowledge base (KB), we can introduce the terminology in the TBox. The terminology axiom form of one domain is: $C \sqsubseteq D$ $(R \sqsubseteq S)$ or $C \equiv D (R \equiv S)$. Here C, D is concept while R, S is role. The first type of formula is called inclusion, and the second is called equation. Thus, we give a relation organization of an application domain glossary (atomic concept, atomic role, complex concept and role).

The ABox of the knowledge base gives the assertion naming the individual according to this glossary. In ABox, according to the concept and the role, we may describe a special condition of the event in an application domain. We introduce them through the individual name, give these individual characteristics, and also express these individual

Table3.1 ALC Syntax & Semantic

Construction Operation	Syntax	Semantic	Sample
Atom Conception	A	$A^I \subseteq \Delta^I$	Person
Atom Role	R	$R^I \subseteq \Delta^I \times \Delta^I$	hasChild
Top Conception	⊤	Δ^I	True
Bottom Conception	⊥	ϕ	False
AND	C⊓D	$C^I \cap D^I$	Person⊓Female
OR	$C \sqcup D$	$C^I \cup D^I$	Male⊔Female
NOT	$\neg A$	$\Delta^I \setminus A^I$	¬Female
Existential Quantifier	∃R.C	$\{a \in \Delta^I \mid \exists b.(a,b) \in R^I \wedge b \in C^I\}$	∃*hasChild.Female*
General Quantifier	$\forall R.C$	$\{a \in \Delta^I \mid \forall b.(a,b) \in R^I \rightarrow b \in C^I\}$	∀*hasChild.Student*

names with the signs a, b and c. We can make the two following assertions about C (a) and R (b, c); i.e. concept C and role R. The first is called the concept assertion which indicates that a is one instance of concept C, or a belongs to the object set of the C. The second is called the role assertion which indicates that, regarding b, c is an actor of the role R. For example, if PETER, PAUL and MARY are the individual names, then Father (PETER) means PETER is a father, hasChild (MARY, PAUL) means PAUL is a child of

MARY. ABox is a finite set including these assertions.

We can regard the description logic language as a piece of the first-order predicate. Because an interpretation *I* has assigned each atomic concept and each atomic role as a unitary relation and a binary relation separately in the set Δ^I, we may regard the atomic concept and the atomic role as an unitary predicate and a binary predicate. Then any concept C may be translated effectively into the predicate logic form $\Phi_C(x)$ with a free variable x. Each interpretation *I* which satisfies the element Δ^I of the set $\Phi_C(x)$ is C^I. The atomic concept A is translated into the form a(x). The union, the conjunction and the negative constructor are translated respectively into the conjunction, the disjunction and negative of the logic. If the concept C has been already translated into the predicate logic form, R is an atomic role. The existential quantifier and the universal quantifier can be represented by this form:

$$\Phi_{\exists R.C}(y) = \exists x.R(y,x) \wedge \Phi_C(x)$$
$$\Phi_{\forall R.C}(y) = \forall x.R(y,x) \rightarrow \Phi_C(x)$$

Here, y is a new variable. The cardinal number can be represented as in the description logic:

$$\Phi \geq nR(x) = \exists y_1,\ldots,y_n.\ R(x,y_1) \wedge \ldots \wedge R(x,y_n) \wedge \bigwedge_{\mathrm{i<j}} y_i \neq y_j$$
$$\Phi \leq nR(x) = \forall y_1,\ldots,y_{n+1}.\ R(x,y_1) \wedge \ldots \wedge R(x,y_{n+1}) \rightarrow \bigvee_{\mathrm{i<j}} y_i = y_j$$

Some people may believe that since the concept may be translated into the predicate logic, it is not necessary to define the special grammar again. However, from the above translation, the cardinal restriction is quite special, and the description logic without the variable grammar is more accurate. Such processing also promotes the inference algorithm development. Thus we can assert that the description logic is the decidable subset of the first-order predicate logic.

The ontology that this chapter constructs is based on description logic. The domain ontology is represented as the static knowledge of the real estate being observed, but the process ontology is represented as the real estate service logic and supports the combined search for the agents.

3.4 Domain Ontology Modeling

3.4.1 Domain Ontology Concept Gain

3.4.1.1 Text Analysis

The numerous documents in an enterprise are the carriers of enterprise knowledge. Generally speaking, the documents in an enterprise can be divided into three types: i) non-structured documents (ordinary text): ii) documents with a formatted text structure (half structured information); and, highly structured documents (structured information). The process of knowledge ontology extraction involves the manipulation of all three types of documents in order to turn them into structured knowledge information. Figure 3.3 illustrates the process of ontology extraction. Text analysis is performed on the scanned documents (including documents with a fixed structure and non-structured documents) to obtain a series of glossaries.

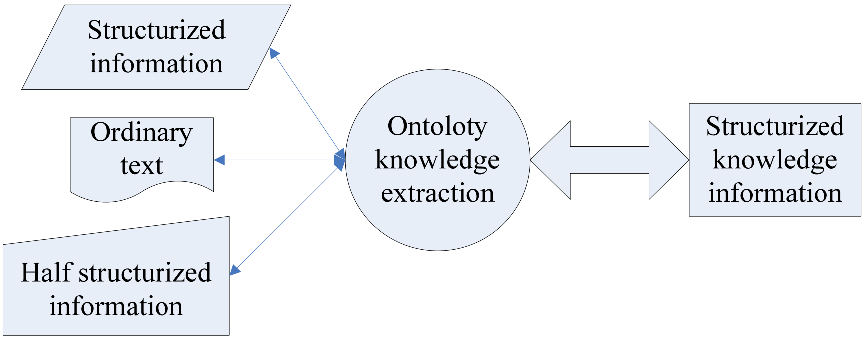


Figure 3.3 Ontology Gain Process Diagram

The concrete algorithm is as follows:

Step 0: Judge the input document. If it has a fixed structure Schemata, then carry out Step 1; otherwise, go to Step 2.

Step 1: Divide the document into a number of units through the input Schemata and continue with the following operations. If the schemata does not come forth according to the appointed sequence, print the error message. If there is no schemata in the document, regard it as a whole block.

Step 2: Partition the document. Each block of the document may be partitioned by observing blank lines, the embedded head line or the different lengths of the lines. The line mark and hyphens are removed, and the non-text elements are deleted.

Step 3: Extract the terms. Scan the document word by word and produce the glossary. Digital or the combination of digital and the character are processed emphatically. Simultaneously, the capital letter, lowercase and the term shape are also produced, such as: the comparative and superlative degree of the adjective; the singularity and plurality of the noun; the past tense and past participle of the verb; and so on. All terms are made identical; digital or the combination of digital and the character are regarded as constant.

With the above algorithm, a series of glossaries are produced.

3.4.1.2 Terminology Produce

In this process, the glossaries, which are produced through text analysis, are further analyzed to obtain the phrase terminology and extra structural characteristics such as grammar, the upper and lowercase of the attribute, etc. The exact process is as follows:

1. Glossary shape analysis. After the shape analysis of each glossary, we can obtain the grammatical type (noun, verb, adjective and so on) and the vocabulary (word-stem) through reference to some solid basic vocabulary in the system. In this process we may use some other grammatical markers, such as the first capital letter of the noun, the suffix "ed" of the verb, and so on. We may also judge the attribute of the glossary according to the context or its role in the sentence. For example, an adjective or noun phrase follows the linking verb, and nouns follow prepositions, intransitive and transitive verbs. The subject of a sentence is usually a noun or a noun phrase. When analyzing the characteristics of the word form, we should first judge its context, then analyze the type of the sentence to find out the verb predicate of the sentence, and finally analyze its position in the sentence. We can judge the part of speech according to the above rules.

2. Distinguish the important glossary from the stop-words.

3. Choose the glossary and produce the phrase. It requires a combination

of manual and automatic work to construct the phrase. We may use the common glossary or other glossary warehouses. For example, we can scan backward and match the functional pattern in the Kingsoft Powerword Warehouse to find out the corresponding phrase in the glossary warehouse. We can also differentiate the phrases with the help of the space mark between the phrases. For example, those before or behind "a, an, the" belong to different phrases respectively. The user makes the following choice of each word:

- Option 1: Whether this word includes only one glossary; and,
- Option 2: Whether this word can be regarded as the initial character of a phrase.

3.4.1.3 Terminology Context Matrix

The goal of this stage is to count the frequency of the terminology in certain contexts. The different context includes the following: (1) the entire document - the context is the entire document in the most cases; (2) the independent document paragraph; or, (3) a fluctuation frame with the length n around the terminology.

The knowledge engineer chooses the context type first and then gains the frequency of the terminology. The algorithm uses (1) and (3) in default. For a few big documents (novels, long reports) in the producing stage of the basic ontology, context type (3) is used since more useful information can be extracted. For numerous small documents with the same structure (e.g. forms), context type (1) is used. Context type (2) is used in establishing the scene pattern of the task. The glossaries can be formed when the slot name has been given.

We can fill in the context matrix according to the following form.

Table 3.2 Context Matrix

	Context Type 1	……	Context n
Glossary 1	×	×	×
……	×	×	×
Glossary n	×	×	×

3.4.1.4 Calculating the Similarity among the Terms

If the frequency of two glossaries that appear simultaneously in all contexts is higher than some threshold value, two glossaries are said to be similar. The threshold value is chosen dependent on the actual situation.

The similarity computation includes the context choice of the terminology, weight allocation and the measurement method of the similarity. Each side provides a series of methods, which may combine with each other to obtain different results.

There are three context types: "full text", "paragraph" and "glide window". These three contexts, when the scale of the document is not very large, are suitable. However, due to the sparse matrix, when the document becomes big, computational difficulties arise. We can consider using the SVD (Singular Value Decomposition) in the LSA analysis (Latent Semantic Analysis) to degrade the dimension of the terminology.

There are two weight allocation schemas: (1) binary weight assignment, which sets the weight to "1" if the terminology appears in a certain context, otherwise to "0". The idea being that if the terminology appears one time in the context, which indicates that this document has discussed the related thesis (Topic), it is recorded as "B"; (2) The product of TF (Term Frequency) and IDF (Inverse Document Frequency). The concrete algorithm may refer to the vector space representation method of the information search in the second chapter of the literature.

The computation method of similarity is that we can form the vector of this terminology by calculating the terminology frequency weight in some contexts and obtain the glossary similarity by calculating the similarity of two vectors. There are three kinds of concrete algorithm: (1) scalar product between two vectors; (2) cosine angle between two vectors; and, (3) Jaccard Score in the binary weight.

Suppose two terminology vectors can be represented as:

$\vec{A} = < a_1, a_2, ..., a_n >$, $\vec{B} = < b_1, b_2, ..., b_n >$, i.e.

1. the vector scalar product SP: $< \vec{A}, \vec{B} > = \sum_{i=1}^{n} a_i b_i$

2. the vector cosine angle α : Cos α $= \frac{|< \vec{A}, \vec{B} >|}{|\vec{A}| * |\vec{B}|}$, here : $|\vec{A}| = \sqrt{\sum_{i=1}^{n} a_i^2}$,

$|\vec{B}| = \sqrt{\sum_{i=1}^{n} b_i^2}$

If their similarity is greater than the given threshold derived through calculating the two terminology similarities, they are similar or correlative. It also indicates that there is a certain logical connection between these two terminology concepts. After determining similar terminology, we may obtain further terminology relations through to the following analysis.

The further relation of the two terminology vectors may be as follows:

1. $\overrightarrow{A} \Rightarrow \overrightarrow{B}$ indicates that if terminology A appears in the context, terminology B must also appear, and the concept described by terminology A must belong to the concept of terminology B, that is to say, B is the father class of A;

2. $\overrightarrow{A} \Leftrightarrow \overrightarrow{B}$ indicates that if terminology A and terminology B appear in the context simultaneously, terminology A and terminology B are equal and they express identical concepts, or the relation between the two terminologies is that between the concept of the principal key of the concept and cannot exist independently;

3. $\overrightarrow{A} \Leftrightarrow \overrightarrow{\neg \mathrm{B}}$ indicates that the concept which the two terminologies describe is supplementary and cannot coexist, that is to say, the description is of conflicting concepts.

We can establish the system of classification through the above four-stage analysis. The terminology classification can be connected with some sub node of the top domain ontology according to the top domain ontology describing the objective world. For example, the terminology classification obtained after processing numerous text documents describing the real estate industry can be connected separately to the ontology describing the geography and the ontology describing the building materials. This would be regarded as a more refined division of the top ontology.

3.4.2 Domain Ontology Modeling Method

In the light of the characteristics of the knowledge of the real estate industry, this article proposes a step-by-step domain ontology modeling method (Natalya F. Noy and Deborah L. McGuinness 2001). Because the human understanding of the domain is an evolutionary process, and the knowledge about a domain changes continuously with time, the establishment of the domain ontology model should also reflect this process. We may define a sketchy ontology model first, revise and appraise the ontology model continually along with our deeper understanding of the domain. This process includes the following stages:

1. Determine the domain and scope

   Clarify the professional field and ontology goal and the functions that the ontology will cover. Determine the ontology's user and preserver. These problems do not usually require formalized expressions. They can be done using natural language descriptions, and can be performed by the low level modeling personnel. In addition, it is necessary to make some specific problems, for example, the annotations of some special expressions of specific professional fields and specific detailed content, etc.

2. Consider reusing the existent models. The cases for reuse are as follows:

   - The model produced by the system design includes the business model, the functional model, the data model, and so on. These models relate to the analysis and abstract of the domain view and level, and are approved by the related domain experts. We may reuse these concepts, data, concept classification systems and the relations between the concepts of the models.

- The data constraints of the entity relationship model provided by the existing information system and the database constitute the actual application standard. They are also the accumulation of the domain knowledge and the information modeling experience. We may obtain many concepts and attributes describing these concepts from these models and the relations and constraints between some concepts as well.

- The authoritative literature and dictionary in the domain. For example, a dictionary or encyclopedia provides the authoritative interpretation of some concepts, which may be taken as the natural language text of the concepts.

- Ontologies issued on the Internet. At present many ontologies are available through the Internet, and many ready-made ontologies can be imported into the ontology development system.

At present there are also many ontologies relating to the real estate industry on the Internet. There are also many other ontologies concerning other domains. We may reuse them. For example, there is the standard geographical information ontology about the geographical information description on the Internet. We may import this ontology into our own directly in the construction of the real estate industry.

3. Identify the related concepts in the domain

You should try your utmost to enumerate all terminology that the system needs to relate to its users in the preliminary stage of the creation of the domain ontology. The methods of determining these are:

- Brain Storming: The modeling personnel write down the related latent domain concepts according to their domain knowledge. The merit of this method lies in that we can obtain a series of domain concepts very quickly and efficiently. However, these concepts may be inaccurate or incomplete and may need to be vetted by domain experts.

- Business Process Extraction: The modeling personnel identify the concepts related to the information text by analyzing each business process. The merit of this method lies in that we can obtain complete domain concept collection. The shortcomings are the low efficiency and the large amount of work required.

At present, in the initial stage of the ontology construction, knowledge engineers process numerous documents in the domain with text processing tools, and extract the concepts which can be used as the reference for constructing the ontology. The ontology produced in this way is comparatively comprehensive. It is very important to extract the full range of relevant concepts from the documents. The core algorithm will be introduced in the next chapter.

4. Determine the attributes of the description concept

The defining of the concept's attribute must follow the principle of guiding by the demand. It is necessary only to describe the characteristics and structure related to the demand. All the characteristics of the concept need not be described. Generally speaking, the following characteristics can become attributes of the ontology concept:

- Intrinsic characteristics, such as the building material;
- External characteristics, such as the design;
- Object constituents - if this object is structural, then its composition may be physical existence or abstract;
- Relation with other objects -this is the concept instance and the relation between other concepts or the concept instances, and this attribute manifests horizontal relationships between concepts.

There are three methods of extracting the concept attribute:

A, Extract from the service model. The information flow including the information the activity needs and produces, which is usually taken as the attribute describing the activity.

B, Extract from the existing database or the information system. The database and the information system are specific models in the question domain, including the object and data the current system relates to.

C, Consult professionals. The concept attribute describes the concept characteristics we care about in the question domain. Professionals in the field are familiar with the domain.

Having determined the attributes, we must define and describe the attributes, including such aspects as: name, definition, data type, value scope, cardinal number (value number), and the characteristics of this attributes (reflexivity, transitivity, functional nature). The definition of the attribute is the same as the concept. The connotation of the attribute must be explained in precise and clear language.

5. Determine the concept level

The concept collection obtained through the above step is basically the horizontal structure without systematic presentation. However, this step systematizes the concepts. Classification is the method of making the information systematic in the domain used most popularly at present. The establishment of the classification system facilitates the establishment of the overall view of the domain and the analysis, and so the determination of the concept level may also refer to the classification method.

There are three methods to determine:

A, Top-down development process. Start from the most general concept and from there define the special concept;

B, Bottom-up development process. Start from the most special concept, and the leaf nodes of the concept level, and then classify them into the general concept; and,

C, Integration development process. This method is a combination of the above two methods; define the salient concept first and then induct suitably and undertake a special study of it.

6. Define the instance

It is necessary to pay attention to the following aspects in defining an instance of the concept:

A, Choose the concept needing instantiation;

B, Create an instance of this concept; and,

C, Calculate the attribute of this concept.

7. Code Ontology Model

An ontology editing tool can be used to code the ontology model. We can use the protégé tool of Stanford University to visualize the ontology model and produce the language OWL text document to describe the ontology.

8. Evaluation of Ontology

The evaluation of the ontology is done in relation to the documents as well as the software environment according to the demand description, ability problem and so on. After formalizing the ontology, we should evaluate these problems, such as whether it meets the demands which we have proposed, whether it satisfies the establishment criterion of the ontology, whether the ontology terminology is defined clearly, whether the ontology concept and relation are integrated, and so on. The evaluation criteria of the ontology are many, including some general criteria which may be applied to all ontologies.

The ontology model should generally follow the following principles of design:

A, Explicit and viewable. The ontology should use the natural language to give clear and objective semantic definition of the terminology;

B, Completeness. The given definition is complete. It can express the meaning of the terminology definitely;

C, Consistency. The deduction obtained by the terminology and itself meaning should be consistent, and will cause no contradiction;

D, Extendibility. Appending the general or the special-purpose terminology to the ontology model does not need to revise the existent content; and,

E, Slightly pledge. Give restraints to the modeling object as few as possible.

Next, we will take the real estate domain as the background, and construct the real estate basic ontology model according to the above ontology modeling methods.

1. Determine the domain and scope of this ontology: the real estate industry. It can overcome the semantic inconsistency between the data models of different knowledge management systems in the present real estate industry, and provide the unified semantic foundation for different real estate knowledge management systems to realize interoperability among different systems, such as unified information search, knowledge sharing and so on.

2. Consider the reuse of the existing ontology. At present, there are only a few ontologies about the real estate industry on the Internet, but ontologies about other domains are many. We may reuse other ontologies, for example, description of the geographical information. There are many ontologies about the standard geographical information in the network. We can import them into our real estate ontology directly.

3. Enumerate the important terminology in the domain. We can extract many important terminologies from many text documents, describing the real estate industry such as geographical information, customers, real estate and developer.

4. Define the attribute of the class. For example, the attributes of the geographical information include communication, location and environmental quality. The customer's attributes include address, name, family address, sex, occupation, telephone number and salary. The real estate attributes include: light, household, house quality, floor, decoration level and area. The developer's attributes include: credit, address, name, telephone number, scale and property.

5. Define the class and its level. This is classification about sub-class. We can divide a big class into many subclasses, for example, the customers may be classified into white-collar customers, blue-collar customers, and so on according to their salary attribute.

6. Define the instance. We may define its instance on the concept layer, or define on the database layer, according to the actual situation. We may add various users, real estate information and so on. This step can be maintained dynamically in the operation of the knowledge base system.

7. Code the ontology. We can use ontology modeling tools, for example, the protégé tool may visualize the process of constructing the ontology; then save it as the file or database, which can be taken as the resource layer of the knowledge management system.

8. Evaluate the ontology, which is made during the process of its use. The knowledge engineers continually evaluate and revise in practical application.

3.5 Process Ontology Modeling

3.5.1 Description Logic Based Process Ontology Criterion

The process ontology describes the dynamic business logic of a real estate knowledge management system. However, a real estate knowledge management system would face the Internet directly. The best means of addressing this huge quantity of knowledge is to use an Agent proxy to search for types of information. Therefore, the process ontology constructed by this article needs to manifest the dynamic knowledge in which Agents participate. According to organizational and social theory, when an individual can't finish a task alone, he must seek others' help. This brings about cooperation, which manages the dependence of social activities. In the Internet environment, the information-searching Agent based on a knowledge management system can't read other's mind, ability, language, etc. This is just like in the real world when individuals from wildly different backgrounds try to communicate. There are only two ways for them to exchange information: either, 1) they have one common standard of behavior and language; or, 2) they have a third party interpreter who can provide the transformation between their two dissimilar languages and behaviors. Similarly, information-searching Agents' have only two possibilities for cooperation in the Internet world: either, 1) there exists one common language which everybody understands; or, 2) there exists a mediator who can translate from one language to another. Because there are many information-searching Using the second way to cooperate would require N^2 mediators (where N is the number of the information Agents in the system). This is impractical given the large number of Agents. Hence, only the first method can be considered. The process ontology based on description logic describes the activities' dependence relationships between Agents participating in searches for information. It has explicit semantics. It might be taken as the semantic standard of information-searching Agents in the cooperative search. This process ontology is represented in the following figure.

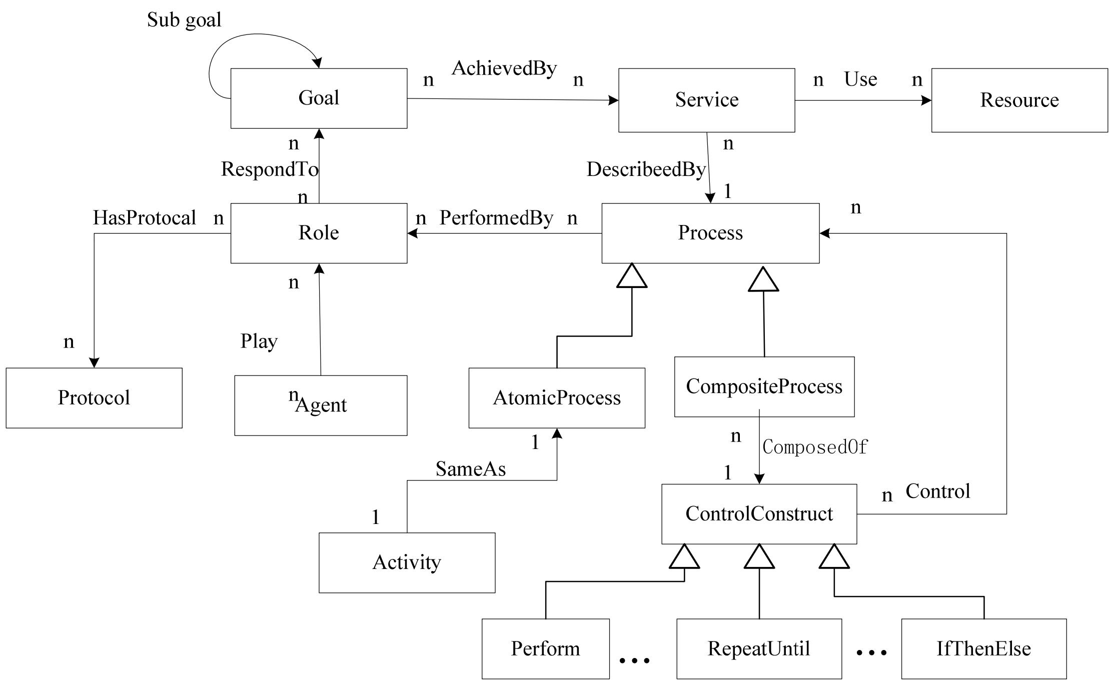


Figure 3.4 DLOP Core Conception-Relation Diagram

3.5.1.1 Basic Concept

Definition 2.1 Goal: To express the user's expected condition.

Its attributes include: GState, GData, AchievedBy and SubGoal, herein,

GState: goal status collection, composed of a series of the assertion formulas describing the objective world;

GData: goal data set, composed of a series of variable assignment sentences;

AchievedBy: service collection, indicating the goal solution;

SubGoal: goal subset, analyzing the subgoal set composing this goal;

Definition 2.2 interact Protocol: which refers to the message exchange pattern between the roles.

The interactive protocol must have one sponsor, one or many responder, that is to say the interactive protocol has two attributes: Sponsor and Responder, whose value scope is Role.

Definition 2.3 Role: which is a kind of entity abstract description, has some functions, and can undertake some responsibility and achieve some goal.

Its attributes include: RespondTo, Permissions, Perform and Protocols, herein,

RespondTo: expresses the duty and the function this role should undertake, and may be represented by the goal;

Permissions: describes the right roles have to complete its function, such as the operation power, the property right to the resource and so on;

Perform: expresses the activity this role can undertake, represented by the atomic process;

Protocols: what this role should obey when interacting with other roles;

Here, although the role definition conflicts with the one described by the ontology meta language; it is very easy to differentiate according to the context.

Definition 2.4 Process: This is composed of a group of the activities to achieve the predetermined target or the expected condition.

A process is a binary function describing the change of the state of the world around executing the procedure. The world condition includes assertion formula collection and the data set about the world, therefore, a process's attribute includes Pr, E, I and O, where:

1. Pr is the precondition formula collections, specifying the condition it must satisfy before executing the procedure;

2. E is the result formula collection, specifying the result collection it obtains after executing the procedure;

3. I is the input data set, specifying the input data before executing the procedure;

4. O is the output data set, specifying the output data set after executing the procedure.

Generally, to express vividly the characteristics of the binary function describing the change of the state of the world, the process is represented as:

Process=. $\frac{<\mathrm{Pr},\mathrm{I}>}{<\mathrm{E},\mathrm{O}>}$

Definition 2.5 AtomicProcess: this can not be divided again. It is the smallest unit the Agent executes, and the subset of the process.

AtomicProcess$\sqsubseteq$ Process, defining here: Activity$\equiv$AtomicProcess.

Definition 2.6 ControlConstruct: its description is as the table 3.3.

So, the process of DLOP can be described as following:

1. Atomic process P is a process;

2. If α is a process, Prm(α) represents excuting a process;

Table 3.3 DLOP Construction Semantic

Construction	Semantic
Prm(α)	install and carry out a process α
S(α, β)	α&β combines a compound process sequentially
Sp(α, β)	When executing compound process, α&β execute concurrently
Sj(α, β)	When executing compound process, α & β converge a spot concurrently after combining one
AnyOrder (α, β)	α&β build sequence compound process according to stochastic
Choice(α, β)	When combining a procedure, α&β select one random to carry out a process
IfThenElse(α, ϖ, β)	When executing compound process, if assertion formula ϖ is true, carry out α, otherwise β
Ru(α, ϖ)	When executing compound process, after executing α, check ϖ, if ϖ is true, executing α repeatedly until ϖ isn't true

3. α, β represent processes, so, S(α, β), Sp(α, β), Sj(α, β), AnyOrder (α, β), Choice(α, β) represent the composition processes;

4. ϖ is an assertion formula, α, β represent the processes, IfThenElse(α, ϖ, β),Ru(α, ϖ) are composition processes.

Definition 2.7 Service: it is a process with goal, and is described by the process. A service can only described by one process.

Definition 2.8 Agent: it describes the intelligent body with a life cycle, the independency and autonomy participating in the information search. It can complete the user's tasks.

The Agent attributes include: Name, Location, CurrentState and Play, herein

Name: the logical name, the exclusive identifier representing Agent in the cooperation space;

Location: the physical address, composed of the IP address and port number;

CurrentState: Agent's state in the cooperation space. Its value is: physical state (Initiated), active state (Active), waiting status (Waiting), suspended state (Suspended) and transit condition (Transit);

Play: the role collection, representing the Agent's ability to search information(the scope of the information search).

Definition 2.9 Resource: refers to those objects needed when executing the process or activity. At the same time, the resource may be divided into reproducible, consumptive, share and monopolistic and so on.

3.5.1.2 Basic Relation

Definition 2.10 SubGoal: It is the relation between the goals, and can be used to express the layer relation between the goals.

A complex goal is composed of many simple goals, but a simple goal is possibly one or many sub goals. Therefore, the relation between the goals is many to many.

Definition 2.11 RespondTo: It is the relation between the role and the goal, and can be used to express some goal some role undertakes.

The role may undertake one or many goals, and a goal may be undertaken by one or many roles. Therefore, the relation between the role and the goal is also many to many.

Definition 2.12 Play: It is the relation between information search Agent and the role, and can be used to express that the information search Agent participates in the cooperation through playing a role.

The information search Agent may play many roles, and a role may be played by many different information search Agents. Therefore, the relation between the information search Agent and the role is many to many.

Definition 2.13 PerformedBy: It is the relation between the process and the role, and can be used to express the roles that can carry out a certain process.

The same process may be carried out by many roles, and a role has the ability to carry out many processes. Therefore, the relation between the process and the role is many to many.

Definition 2.14 AchievedBy: It is the relation between the goal and the service, and can be used to express the services that can achieve the goal.

The ways to achieve the same goal may be different, and the goal may be realized by many services, and meanwhile the service may achieve many goals. Therefore, the relation between the goal and the service is many to many.

Definition 2.15 DescribedBy: It is the relation between the service and the process, and can be used to express the processes describing the service.

A service is the process with the goal, and it can be described only by a process, and one process may describe many services. Therefore, the relation between the service and the process is many to one.

The service is the bridge between the goal and the process. As the service has only one process to describe, so the relation between goal and the process may be indirectly many to many, and the goal may be realized through executing the process. Therefore, $\text{AchievedBy} \times \text{DescribedBy}$ is defined as the relation between the goal and the process, indicating that the goal can be realized through carrying out some concrete process.

Definition 2.16 ComposedOf: It is to describe the relation between the compound process and the constructor, and can be used to express how the compound process is composed by the constructor.

The compound process is composed of the processes through constructors, and one constructor may compose many compound processes, but a compound process can be composed of only one constructor. Therefore, the relation between the compound process and the constructor is many to one.

Definition 2.17 Control: It is to describe the relation between the constructor and the process, and can be used to indicate that the constructor controls the execution of many processes.

The process may be composed of many different processes through the constructors, and meanwhile, constructor can control the execution of many processes. Therefore, the relation between the constructor and the process is many to many.

Definition 2.18 Use: It is to describe the relation between the service and the resource, and can be used to express these resources that are called in the execution of a service.

The same resource may be called by many services, and at the same time the service may use many resources. Therefore, the relation between the resource and the service is many to many.

3.5.1.3 Characteristic of Relation

The previous section defined the basic relationships and the value constraints of the relation in DOLP. This section describes the characteristics of the relationship including: transitivity, symmetry, antisymmetry, inversive, reflexivity and antireflexivity.

1. **Antisymmetry**

SubGoal between the goals satisfies antisymmetry, described as:

$$\forall g_1 \forall g_2 (\mathrm{Goal}(g_1) \wedge \mathrm{Goal}(g_2) \wedge \mathrm{SubGoal}(g_1, g_2) \rightarrow \neg \mathrm{SubGoal}(g_2, g_1))$$

2. Antireflexivity

SubGoal between the goals satisfies reflexivity, which is described as:

$$\forall g(Goal(g) \rightarrow \neg SubGoal(g,g))$$

3. Transitivity

SubGoal between the goals satisfies transitivity, which is described as:

$$\forall g_1 \forall g_2 \forall g_3 (Goal(g_1) \wedge Goal(g_2) \wedge Goal(g_3) \wedge SubGoal(g_1,g_2) \wedge SubGoal(g_2,g_3) \rightarrow SubGoal(g_1,g_3))$$

These three characteristics of SubGoal guarantee that relational graph composed of the goals through SubGoal avoid the ring, thus guarantee that the inference in the process ontology knowledge library can terminate.

4. Inversive

In order to describe easily, we have defined many Inversive relations in the process ontology as follows:

$$\forall r \forall a(PlayedBy(r,a) \leftrightarrow Play(a,r))$$

$$\forall g \forall s(AchievedBy(g,s) \leftrightarrow Achieve(s,g))$$

$$\forall g \forall r(RespondFrom(g,r) \leftrightarrow RespondTo(r,g))$$

$$\forall s \forall p(DescribedBy(s,p) \leftrightarrow Describe(p,s))$$

$$\forall p \forall r(PerformedBy(p,r) \leftrightarrow Perform(r,p))$$

In addition, the relations in the process ontology DLOP need to satisfy some related constraints in the information search agent cooperation domain. In order to guarantee that the concept has significance in process ontology DLOP, the concepts have to exist in isolation. For example, the goal has only one realization. The role at least has the ability to carry out some activity, and the activity must be carried out by some role.

≥ 1 SubGoal.Goal$\sqcup \geq 1$ AchievedBy$\times$DescribedBy .Process //the goal has one kind of realization way at least, and describes the fact that the goal has one sub-goal at least or one kind of realization process.

≥ 1 Perform.Process // the role must have certain ability to carry out some process

≥ 1 SubGoal.Goal$\sqcup \geq 1$ AchievedBy$\times$DescribedBy .Process // the activity must be carried out by some role.

Here we have enumerated several axioms, and also need the information search Agent to expand its DLOP unceasingly in the search process application so as to become more nearly perfect.

3.5.2 Process Ontology Based Knowledge Base

A process ontology knowledge library contains TBox and ABox. The knowledge library is composed of the individual assertion formulas and the structure assertion axioms by knowledge engineers according to the process ontology standard in the agent information search domain. And the inference mechanism describes the inference functions which the knowledge library provides.

3.5.2.1 Process Ontology Knowledge Base

The process ontology knowledge library can be divided into ABox and TBox: TBox is a group of axioms and assertions expressing the structure of the Agent cooperation domain; but ABox contains a group of individual assertions expressing the domain knowledge. This section mainly introduces the axioms in the TBox, which is defined by knowledge engineers according to the Agent cooperation domain knowledge when constructing the process ontology knowledge library. It is the assertion about the Agent cooperation domain. According to TBox and the Agent cooperation domain the axiom type in this article is divided mainly into Goal axiom, role axiom and movement axiom.

1. Goal axiom

The goal axiom describes the one related to the goal in the process ontology. According to the goal type, it may be divided into: G-G axiom, G-R axiom, G-P axiom, which define separately the assertion formula set between the goal and goal, the goal and the role, the goal and the process.

G-G axiom: like $G1 \sqsubseteq \forall SubGoal.G2$, representing $\forall g (g \in G1 \rightarrow g \in \{a \in Goal \mid \forall b.(a,b) \in SubGoal \rightarrow b \in G_2\})$, namely goal G1 is the sub goal of goal G2. The knowledge engineers can describe the layer relation of goals through defining G-G axioms in TBox of the DLOP. At the same time, it needs to satisfy the SubGoal relation characteristics of the process ontology standard.

G-P axiom: such as $G \sqsubseteq \forall AchievedBy \times DescribedBy.P$, representing $\forall g (g \in G \rightarrow g \in \{g \in Goal \mid \forall p.(g,p) \in AchievedBy \times DescribedBy \rightarrow p \in P\})$, namely the goal G can be achieved by executing the process P.

2. Role axiom

The role axiom describes the relations between the role and the goal in the DLOP , the processes, including R-G axiom and R-P axiom.

R-G axiom: such as $R \sqsubseteq \forall$ RespondTo.G, representing $\forall$ r ($r \in R \rightarrow r \in \{r \in Role \mid \forall g.(r,g) \in RespondTo \rightarrow g \in G\}$), namely role R can achieve goal G, and it is inversive with G-R axiom.

R-P axiom: such as $R \sqsubseteq \forall$ Perform.P, representing $\forall$ r ($r \in R \rightarrow r \in \{r \in Role \mid \forall p.(r,p) \in Perform \rightarrow p \in P\}$), namely role R can execute process P.

3 Process axiom

In the process ontology, the process can represent the binary function of the change of the world state, and the process axiom is used to indicate the essential feature and the rule of the change of the world. It also indicates the change of the condition formula and the data set around the process execution. It may be divided into the condition axiom and the data axiom. The knowledge engineers describe its IOPE attributes through giving condition axiom and data axiom of the activity.

Suppose P= $\frac{<Pr, I>}{<E, O>}$ be the process.

State axiom: such as $W1 \equiv \forall$ Pr.P, representing: $\forall w\ (w \in W_1 \leftrightarrow w \in \{w \in W \mid \forall p.(w,p) \in Pr \rightarrow p \in P\})$, which describes that the assertion formula set W1 is the precondition of process P.

Such as $W2 \equiv \forall$ E.P, representing:

$\forall w\ (w \in W_2 \leftrightarrow w \in \{w \in W \mid \forall p.(w,p) \in E \rightarrow p \in P\})$, describing that the assertion formula set W2 is the effect of process P.

Data axiom: such as Data1 ≡ ∀ I.P, representing:

$\forall d\ (d \in Data_1 \leftrightarrow d \in \{d \in Data \mid \forall p.(d,p) \in I \rightarrow p \in P\})$, describing that data set Data1 is the input data of process P.

Such as Data2 ≡ ∀ O.P, representing: $\forall d\ (d \in Data_2 \leftrightarrow d \in \{d \in Data \mid \forall p.(d,p) \in O \rightarrow p \in P\})$, describing that data set Data2 is the output data of process P.

In addition, the process axiom describes the relations between the process and the goal and between the process and the role in DLOP.

P-G axiom: such as $P \sqsubseteq \forall\ Describe \times Achieve.G$, representing $\forall p\ (p \in P \rightarrow p \in \{p \in Process \mid \forall g.(p,g) \in Describe \times Achieve \rightarrow g \in G\})$, describing the goal can be achieved through executing process P, which is inverse with G-P axiom.

P-R axiom: such as $P \sqsubseteq \forall\ PerformedBy.R$, representing $\forall p\ (p \in P \rightarrow p \in \{p \in Process \mid \forall p.(p,r) \in PerformedBy \rightarrow r \in R\})$, describing process P can be performed by role R, which is inverse with R-P axiom.

ABox is the individual assertion collection according to the terminology (concept and relation) by TBox. According to the concepts and relations on TBox in, ABox(?) introduces the element in the domain through two kinds of assertions C(a) and R(b,

c) to describe the individual condition in the domain.

3.5.2.2 Reasoning Task of Knowledge Base

1. G-G reasoning

The information search agent usually needs to decompose a complex goal into many sub goals in the search process, and the goals compose the inclusion relation in process ontology DLOP. So it is the first important reasoning task to examine the level composition relations among the goals in DLOP, called the G-G reasoning. Suppose some goal g, we can obtain its sub goal set gSet, through the G-G check algorithm. The algorithm is as follows:

(1) Get the class of goal g to be G2, and search G-G axiom like G1 ⊑ SubGoal.G2. So, all the sets G1 composes are GSet;

(2) Iterate each element Gi in GSet, and search G-G axiom like Gj ⊑ SubGoal.Gi. If Gj exsits, Gj are added to GSet;

(3) Iterate GSet unceasingly, until the elements of GSet don’t change;

(4) Instantiate GSet, and search ABox, to obtain each individual set in Gset, to be gSet;

(5) Finish the algorithm. The goal set gSet is namely for the sub goal set of this goal.

The characteristic of the relation SubGoal has guaranteed that the graph composed of the goals has no ring. So, thus the algorithm can be said to terminate.

2. G-P reasoning

In DLOP, the processes might be organized through the goal besides the compound operation, and the goal embodies the world condition or data which the agent hopes to obtain in the search process. Therefore, the agent could search and realize all processes of this goal according to the goal in the search process, namely seeking all schemes solving the problem. Here it is called G-P reasoning. Suppose some goal g, through G-P check algorithm, we can obtain all process collection pSet realizing this goal gSet, through the G-G check algorithm. The algorithm is as follows:

(1) Get the class of goal g to be G, and search G-P axiom like $G \sqsubseteq \forall AchievedBy \times DescribedBy$.P. So, all the sets P composes are PSet;

(2) Get all the sub sets of GSet, according to G-G reasoning;

(3) Iterate each element Gi in GSet, and search G-P axiom like $G \sqsubseteq \forall AchievedBy \times DescribedBy$.P, and each P is added to PSet;

(4) Instantiate PSet, and search ABox, to obtain each individual concept set in Gset, to be pSet;

(5) Finish the algorithm. The process set pSet is namely for all the direct or indirect processes achieving the goal.

The termination of G-G reasoning can also guarantee the termination of G-P algorithm.

Finally, there are many processes to achieve the goal g in the set pSet. Here, we may choose the best process as the process solution to the goal g, according to the actual situation.

3. G-R reasoning

The role indicates not only the responsibilities undertaken to achieve the goal g, but also the authority. Therefore, it indicates the basic diatheses with which the role achieves the goal as the actor and performer of the process. Through the G-R reasoning algorithm, we can obtain the roles to achieve the goal g. Suppose some goal g, through the G-R check algorithm, we can obtain all role set rSet realizing this goal. The algorithm is as follows:

(1) Get the class of goal g to be G2, and search G-G axiom like G⊑ RespondBy.R. So, all the roles R achieving this goal directly compose is the setRSet;

(2) Get all process set Pset reaching the goal G according to the G-P check algorithm;

(3) Iterate each element P in set PSet, search P-R axiom,like P⊑ PerformedBy.R, and add each R to RSet;

(4) Instantiate RSet, and search ABox, to obtain each individual set in RSet, to be rSet;

(5) Finish the algorithm. The role set rSet is namely for all the direct or indirect roles achieving the goal.

4. Process rationality

The process rationality check includes checking the control flow and the data flow. The process constructor controls the connectivity between the processes, but not all the processes can constitute the new process by constructors. The processes need to satisfy certain conditions. So we can divide the process rationality into the following cases according to constructors:

a, S(α,β): need to satisfy $\forall$u(Suc(α,u)$\Rightarrow$$\beta$.Pr), namely the world state after executing any α needs to satisfy the precondition of β, here: u is the world state before executing the process α, Suc(α,u)=(u-α.Pr)$\cup$$\alpha$.E;

b, Sp(α,β)：α,β have no relation, skip;

c, Sj(α,β)：α.O$\cap$$\beta$.O=$\emptyset$ and α.E$\cap$$\beta$.E=$\emptyset$, namely there exists no contradiction between the output set and the effect of α and β;

d, AnyOrder (α, β)：S(α,β) and S(β,α) are all true;

e, Choice(α,β)：the process of one of α and β is chosen and executed;

f, IfThenElse(α, ϖ,β)：If ϖ is true ,then executing the process α is executed, otherwise the process β is executed;

g, Ru(α , ϖ): The process α is executed repeatedly until the state of w is flase.

Through iterating the above check algorithm unceasingly, we can guarantee the rationality of the data flow and the control flow in any process.

3.5.3 Process Ontology Modeling Method

The process ontology standard based on description logic has given the top process ontology which supports the information search agent to search the information that is the elements needed in the process ontology and the relations between its elements and so on. This section mainly studies the concrete application domain and how to model the ontology according to the DLOP standard.

The process ontology is also a kind of ontology, and the modeling method of the process ontology needs to absorb other research achievements of the domain ontology modeling. At the same time, the process ontology in this dissertation is used to support many information search agents in completing complex tasks cooperatively on Internet, so the process ontology modeling should also consider other characteristics of the agent system modeling. We will propose a kind of process ontology modeling method in this section, according to the gradual modeling process in our modeling the domain ontology. This modeling process is a circle advancement and step-by-step refinement process. If we find the irrationality in the following modeling step, we can move back to modify the corresponding result until we construct the rational process ontology model. The process ontology refers to the object knowledge (the emphases in the traditional domain ontology), activity and process knowledge, and organization knowledge in the domain, and these parts relate to each other, but the sequence does not exist. So the whole modeling process is not absolutely linear, and some steps are parallel.

We will illustrate the process ontology modeling method in detail below. The sign mark before each part does not represent its sequence.

1. Determine the domain and scope

It is only the beginning to determine the domain and scope of the process ontology. This step can be completed by answering a series of capability questions. The capability questions are the ones the ontology based knowledge system can answer, and can be used to check whether the ontology is suitable: Does the ontology contain enough information to answer these questions? Or, do these answers need to be partitioned in detail or be represented by a certain domain? The ontology boundary and detail degree are the base of the following step.

2. Consider the reuse of existing ontologies

According to the ontology domain and scope determined in the first step, we can review whether the suitable referred correlative ontology exists. If it exists, we can reuse it to decrease the modeling workload. The reuse can be divided into two cases: 1) Refer to the existing ontology. We can add the elements of the existing ontology into our ontology according to the need while modeling. The resource classification in this dissertation has reused the achievements of the classification of the former resource ontology. 2) Take the existing ontology model as part of our ontology directly, like interactive protocol IPs. In this situation, we can introduce it into the current constructing process ontology through explaining URL of the reused ontology in the namespace.

These two steps are the ones of initial ontology modeling method, which are also the base of the construction of ontology. According to DL-P standard, we should model the corresponding elements such as the goal, the role, the activity, the process and the entity and so on, after completing the analysis of the domain and reusing the ontology. The goal is an abstract description of the system task. But the role, the activity and the process must be extracted from the system's service logic and the organizational structure. The entity can be obtained by modeling objects in the application domain. This follow-up steps are not linear, but parallel. What must be paid attention to is: although there is no input and output restricted relations among the many modeling steps, because there may be other relations such as the goal and the role, the goal and the process and so on, the different steps have certain influence on others, and may produce the returning of the modeling.

3. Goal modeling

This step needs to mark the structural relations between the goals of the system, and obtain goal decomposition relations of the system. There are many goal decomposition methods, such as those based on the system composition and so on. This step modeling results include the goal, the goal attribute value, the goal refinement (i.e. inherited relations between goals) and the restraint relations between goals in the domain, the most important of which is the decomposition relations (SubGoal) between goals.

4. Entity modeling

This step is to model the objects in the domain. At present, most ontology researches focus on this aspect. With this step, according to the existing individuals in the domain, we should recognize the concepts, the attributes of the concepts, the relation between the concepts, the layer of the concepts (inherited relation) and their restrictions.

5. Business logic modeling

The business logic modeling is to carry on the modeling of the application system from the functional angle. The tools and methods of the business logic modeling are very many, especially have many mature methods and achievements in the research of the software engineering and the information system modeling, and here we wouldn’t detail them. The business logic modeling in the domain reflects the user's demand, and the results of the analysis in this part are the foundation of the following role modeling and process modeling.

6. Role modeling

The role modeling takes the analysis of the domain and the goal modeling as the foundation. There are many methods for the role modeling. Referring to the achievements in the multi-Agent system modeling (i.e. MaSE method (Ke Li and Qinghua Bai 2001)), we carry on the role modeling. For this step it is necessary to determine the roles, the role attribute and the relations between the roles in the domain, including restraint relations between the roles, and establish the corresponding relationships between the roles and the goals. In the modeling result, each goal must have the corresponding role to achieve. Generally, the relation is many to many between the roles and the goals, and each goal must have a corresponding role at least, and a role may achieve many goals. Here, considering the convenience and the validity of the modeling, the similar or related goal needs to be carried out possibly by the same role.

7. Interactive modeling

According to the result of the role modeling, we can model the role interactive protocol. In this step we reuse the related ontology resources which come from the context-sensitive text, and reuse a part of the interactive protocol in the FIPA(Xiaofeng Zhu and Songzhen Lang 1999) standard. The cooperation process needs to follow the interactive protocol when establishing the roles in view of the concrete application domain.

8. Process modeling

According to the business logic obtained by the analysis of the business logic modeling, we can determine what processes exist in the system how the atom processes compose the process.

We confirm the refinement relations of the process, and determine its attribute value for each process, including: the preconditions and effect, the process performer (PerformedBy, this need to correspond to role modeling), the process composition relations (ComposedOf), the data connection relations between the processes, and so on. The result of corresponding goal modeling determines the relations between the goal and process (AchievedBy).

9. Resource modeling

It needs to use resource to execute the activity. This step is synchronized with process modeling. We will recognize the resources and their attributes in the domain, classify and model the resources, determine their restrictions, and expound the Use relation with the activity, according to the resource type in this article.

After completing the above tasks, we can code the modeling result based on the process ontology standard. Until now, we have established the process ontology based on DLOP. But we have not seen modeling the agent, but only modeling the process knowledge and the organization knowledge in the domain. That is because the information search agent is dynamic on Internet, and not suitable to model it. The role obtained in the above modeling is the bridge to connect the domain process model and the agent world. The agent who has joined the cooperation should execute all the activities this role can play, and use the interactive protocol this role can use. In this way, we can obtain the agent model through declaring the roles the agent can play.

This ontology is put in TBox of the process ontology knowledge base, and organized in a certain way to be inquired easily.

3.6 Experiment

3.6.1 Background of Experiment

In the real estate industry, there is a lot of knowledge, an important part of which the knowledge related to real estate project management is. Herein, the real estate project management is referred to main management activities during the whole life of a real estate project. Generally speaking, the whole process of a real estate project can be divided into the following four stages:

(1) Preliminary planning and establishment of the project. The work at this stage is to analyze and make a decision on the real estate project based on market investigation. Its work includes conceptualization, preliminary design, feasibility research and request for authorization of the real estate project. The aim of the whole process is to determine 'go' or 'no go'.

(2) Design and planning stage. The work at this stage includes: detailed design, planning, bidding and preparation for the execution of the project.

(3) Construction stage. This stage is the concrete construction process of the real estate product. For a typical real estate project - for example: housing, office-buildings, apartment blocks, retail, and so on - the real estate may be sold in advance to raise funds (in line with internal polices and relevant laws).

(4) Operations phase. This commences on commissioning and extends to the end of the term of responsibility for the asset. This varies according to the nature of the real estate project.

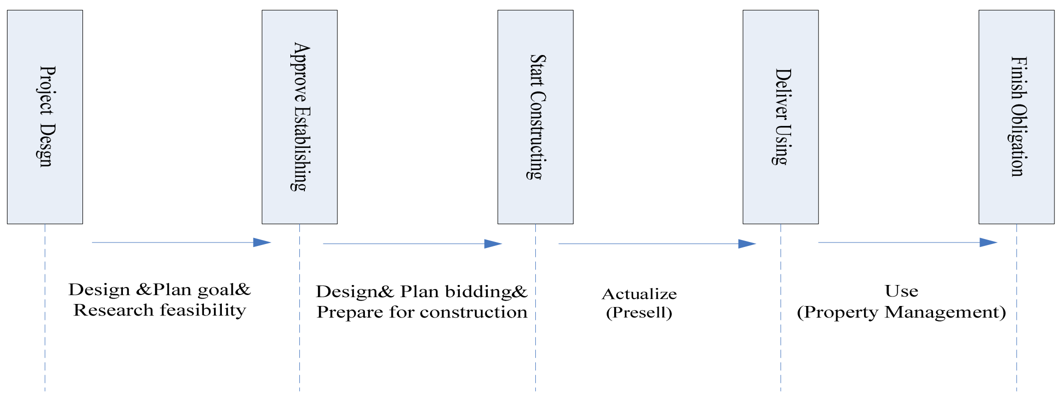


Figure 3.5 Whole Process of the Real Estate Project Management

A real estate project is a complex system, composed of: i) the goal system; ii) the object system; iii) the behavior system; and, iv) the organization system.

(1) The goal system: The project goal system is essentially the full description of the final condition which must be achieved. Objective management is adopted in real estate project management. Therefore the goal system of the project is essential to the project implementation process. It is an abstract system expounded by the project task document, technological standards, contracts, and so on.

(2) The object system: The objective of a real estate project is to realize certain function, scale and quality requirements. It is a complex structure that can be decomposed into many parts; many functional surfaces with their own systematic structural form. This is the concrete system which is typically defined by the project design documents and technical design standards, and completed through a project implementation process. Therefore, the object system of the real estate project determines the project type and nature, the project image, the substantive characteristics and every aspect of the project implementation and management.

(3) The behavior system: This is composed of the activities necessary to achieve the project goal and complete the project tasks. This is the abstract system represented by project structural drawings, network planning, action programs, resource planning, and so on.

(4) The organizational system: This is composed of the behavior Agents which complete the system. It is a concrete system formed by many people and organizations such as building units (the owner), design units, construction units

(the contractor), suppliers, senior management units, and so on. This typically forms a huge organizational system that undertakes the tasks necessary to achieve the singular project goal.

The basic knowledge of a real estate project management is introduced via describing of the project management process and associated system above.

In this experiment, the real estate project management domain and process ontology will be constructed. The domain ontology describes the basic concepts and the attributes in the real estate industry, while the process ontology describes its dynamic transaction logic processes.

3.6.2 Real Estate Project Manage Domain Ontology Modeling

According to the process of domain ontology modeling, at first, the knowledge engineers collect various kinds of documents relating to real estate project management, including: market investigation reports, design task documents, technological design documents, marketing logs, and so on. Then they can extract various glossaries describing the real estate project using the ontology extraction tool, OBKMS. The extract process will be demonstrated as the following.

First, open the file OBKMD.exe. The system then pops up the dialog box as in following figure.

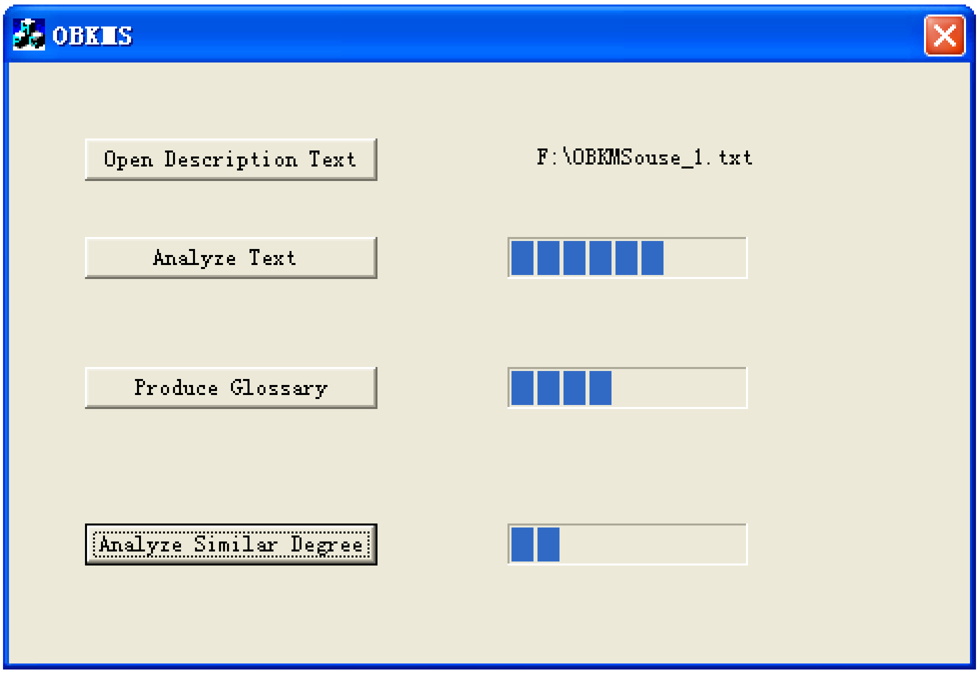


Figure 3.6 Ontology Gain Window

The running operation is as the following:

1. Hit【Open Description Text】button, and choose the real estate description file;

2. Hit【Analyze Text】button, and the ontology obtaining module will change the structural or no-structural texts to obtain the structural knowledge information through analyzing the text.

3. Hit【Produce Glossary】button, then obtain phrasal glossary;

4. Hit【Analyze Similar Degree】button, then count the frequency of the glossary in the context to obtain the context matrix.

Suppose that the knowledge engineers have extracted some basic real estate concepts in the ontology obtaining stage, including: RealEstate、Suburb、LocalAttraction、Environment、RealEstateSystem、RealEstateGoal、Questionnaire、ManageMethod、 MarketInvestigateMethod. These concepts can be extended in the future construction. In this experiment, by means of the ontology visualization software OWLViz ,the layer relation among these concepts is showed as the following chart.

Figure 3.7 Domain ontology schematic drawing of Real estate knowledge management system

3.6.3 Real Estate Project Management Process Ontology Modeling

There are mainly 50 atomic and compound processes related to the real estate project management in the process ontology of the real estate knowledge management system.

The compound process is the complex business logic which the atomic processes constitute through constructing. For instance, Real estate market investigation process (MarketInvestigateProcess) is composed of the preparation for market investigation(PreMarketInvestigateProcess), the implementation of the market investagation (CarryMarketInvestigateProcess) and data and information processing process (DisposalInformationProcess), meanwhile these processes are also the compound process. The nested processes constitute the complex process. The process ontology modeling process is introduced below through enumerating the planning process of the real estate project organization.

The organization and planning of the real estate project are to plan and design the project organization according to the special details of the project and participant, as well as the environmental conditions, which makes the project management an important supervisory work, including:

(1) In the early period of the project organization and planning, we should analyze the total objective and task of the project; complete the technical design and the structural decomposition of the corresponding stages. It is the most foundational work of all the project organization, as well as the basic guarantee to make the project’s overall direction definite, the level clear and the structure clear.

(2) In the middle period of the project organization, we should determine the organizational strategy of the project implementation, make the related problems clear in the project implementation process such as the organizational guiding principle, the project bid plan, the project organizational structure, the management pattern, project material quantity and supplying way, the invested manpower

quantity, the way of investment and so on;

(3) In the later period of the project organization, we should form the regular planning report. This kind of planning report is usually expressed by the bid document, the contract, the project organization structure chart, the project management standard, the project personnel's responsibility matrix diagram and the project handbook.

Therefore, according to the analysis of the plan process of the above real estate project organization, the knowledge engineers divide the early planning period of the real estate organization into several parts: project's overall goal, the analysis of the project total design, and project structure plan process of the early period. The plan process chart of the real estate is shown as the following:

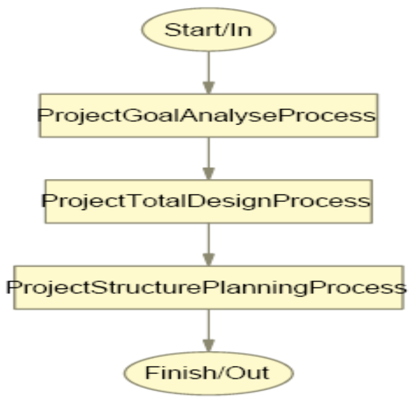


Figure 3.8 EarlyOrganizationProcess

The process of the project organization plan of the middle stage mainly includes: the project aim plan, the project organizational structure and project management mode formulation, which may be carried on simultaneously and concurrently.

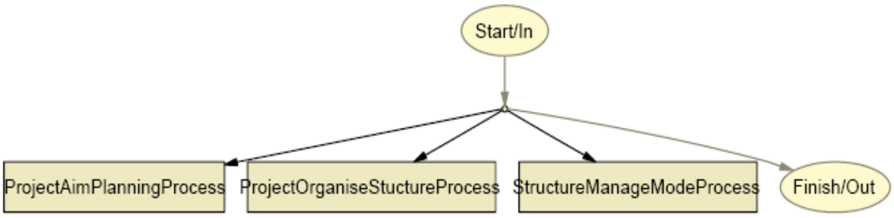


Figure 3.9 MiddleOrganizationProcess

The process of the project organization plan of the later period can be decomposed into: the project bidding, the project management organization, the project management work flow formulation, the project management function decomposition and the contract drawing-up, herein these sub-processes follow respectively the project plan of the later stage, as the following chart shows:

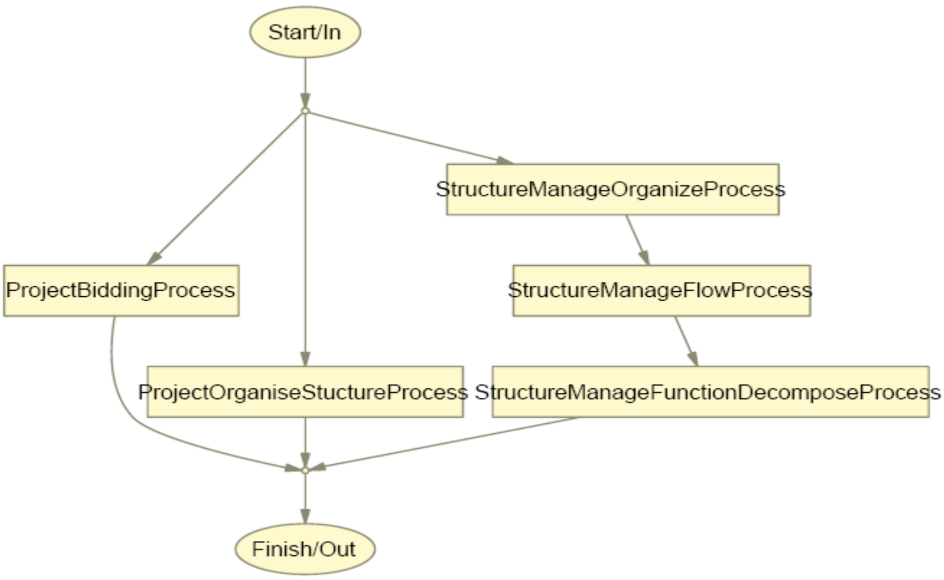


Figure 3.10 LaterOrganizationProcess

Finally, we can plan sub-processes created in each stage of the real estate project organization and synthesize the entire process of the real estate project plan through the construction, which is a complex compound process, and many processes can be carried out concurrently. Also see the following chart.

Figure 3.11 WholeOrganizationProcess

3.7 Conclusions

In this chapter, methods for ontology modeling are presented. Engineers employed on a knowledge management system can use these methods to establish the real estate ontology. The ontology can solve semantic disagreement of the different real estate knowledge management systems to sustain the resource sharing and information unit searches. The established ontology can be used as the basis for the following chapters. However, the ontology modeling methods can't be used in all cases. They need knowledge engineers to modify the methods which are based on the special case. Therefore, ontology modeling methods will be an interesting research topic in the future.

C h a p t e r   4

ONTOLOGY BASED AGENT COOPERATION SEARCH MODEL

4.1 Introduction

Knowledge related to the real estate industry is very diverse. A single real estate enterprise's capacity and capability are limited, and so it is essential to build a real estate knowledge management union among a number of real estate enterprises. In order to create this kind of knowledge management union, different real estate enterprises must hold common semantic concepts, relationships and understandings, which can be realized to create various kinds of real estate ontologies as mentioned in the previous chapter. This chapter will explore how to cooperate between different real estate knowledge manage systems based on commonly acceptable real estate ontologies. As for the issue of agent cooperation, Foundation of Intelligent Physical Agents (FIFA) has already provided some common standards. This chapter will present the ontology based agent cooperation search model, including cooperation frame and mechanism, which accords with the agent cooperation frame of FIFA. The model can be applied to the design of a real estate knowledge manage system. By means of the model, the tasks of knowledge creation, search and sharing can be finished.

4.2 The Agent Cooperation Frame of FIFA

FIPA (Foundation of Intelligent Physical Agents) is an international non-profitable organization devoting itself to establishing operation standards between Agents and the application based on Agents, defining a series of technological standards from the architecture, the communication language and the interactive protocol of the multi-Agent system (MAS) cooperation. The developers of the MAS system can follow this set of standards to develop the multi-Agent system with high interoperability. Figure 4.1 describes the Agent cooperation framework proposed by FIPA organization (S. Geneva 2004).

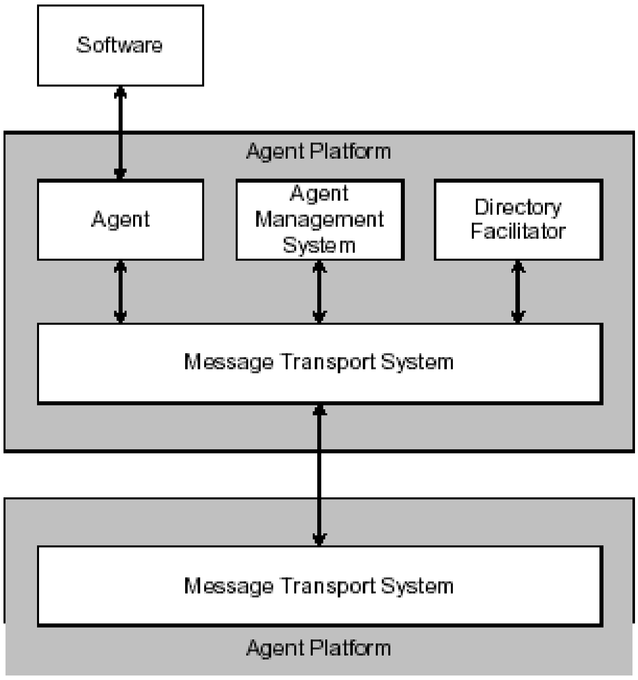


Figure 4.1 Agent Cooperation Model

This model provides the Agent living environment including: the white page service, the yellow page service and the message transmission service. The white page service refers to Agent naming service and the access control service; the yellow page service mainly includes the Agent localization service and the Agent registration service; the message transmission service guarantees the message to be passed smoothly inside and outside the system. As the figure shows, the entire cooperation model contains the following elements: Agent, DF (Directory Facilitator), AMS (Agent Management System), MTS (Message Transport System) and software (Software).

Agent is the basic role on AP, and an intelligent body with one or many kinds of service functions. It can interact with the outside users, the communication establishment and interact with each other. Each agent on the platform has a lifecycle and only one global identifier. Its lifecycle includes five states: the physical state (Initiated), the active state (Active), the waiting state (Waiting), the suspended state (Suspended) and the transit state (Transit). The agent's states can switch among these five ones, but can stay in only one state in any time. The following chart is the switch diagram among these states.

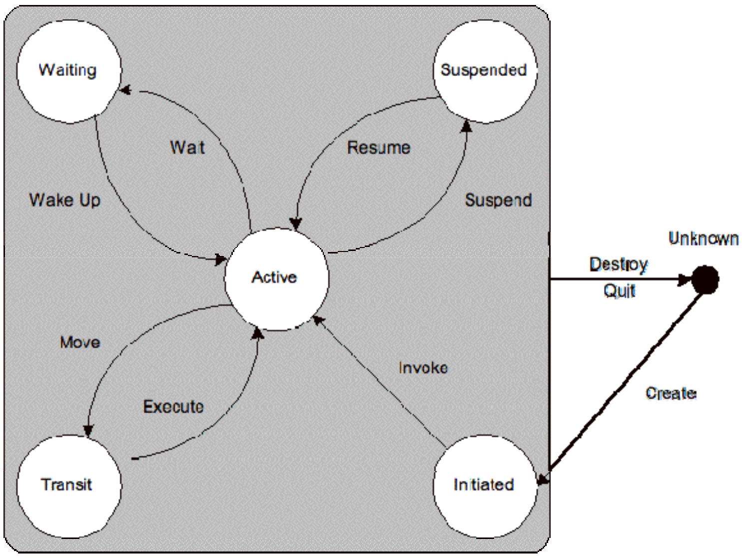


Figure 4.2 Agent Lifecycle Diagram

DF is the essential factor on AP, and manages the agent directory in time. It also supplies the latest directory information about the agents unconditionally to all the authorized agents. DF usually exists on AP as the service form, and Agent can register in and logout from the DF. Only the ones that have registered in the DF can be known by other agents. That's to say that DF decides the boundary of agents joining the cooperation in the logic. As long as Agent publishes its necessary information about the cooperation space in the DF, such as capability, resource and so on, other agents can inquire the intended cooperation agents through the directory service, and localize the Agent in the view of capability. The registered contents in DF usually include: the agent identifier, the capability, the protocol, the ontology, the interactive protocol, the language, the promise time and so on. The Agent capability is represented by name, type, protocol, ontology, language, owner, attribute and address. The iterative protocol can be the one that FIPA has provided or the user himself designs.

There is only one AMS on AP. It monitors the whole AP, controls the access and use of some Agent on AP, and provides the white service for other agents. Here, the white page service provides the basic information about Agent, such as the logic name, the physical address, the description except the inner capability. AMS maintains the index of all the agents staying on AP. The index includes Agent identifier AID, which provides the necessary information about the Agent physical location, such as the physical address, the communication port number and so on. Other Agent can find it through the AID. Each Agent can not enter AP, until it registers into AMS, and then AMS checks whether the Agent description is effective, especially ensures that AID is only one in the local space. After Agent passes through the check of AMS, it can use the message transmission service MTS provided by AP. After Agent logouts from AP, the Agent lifecycle stops. At the same time, the Agent's AID will be deleted from the directory. This AID will be reused by other Agents.

AMS also manages the operations related to AP, such as the creation and destruction of Agent. It also determines whether Agent can register into AP dynamically. If AP can support the mobile Agent, it monitors whether Agent transfers out of the AP, or whether other Agent transfers into AP. We can obtain the AP description through inquiring AMS to know the function of the platform.

MTS is the default Agent communication way on AP to realize the message transmission service on AP. MTS is responsible for transferring messages between the agents on AP, or between the Agent on AP and the Agent out of AP. All the agents have at least one MTS to receive and send messages.

With the development of the agent technology, there are many MTS technologies as the bottom implementation technology of heterogeneous Agent communication. As for the Agent communication language, both of KQML (Kuokka D. and Harada, L. 1995) and FIPA ACL (S. Geneva 2004) give the common format of the interactive communication between Agents. They use vocabularies and sentences to prescribe the expected interpretation of the message. The users can define their message contents. There are many research achievements in the interactive protocol, and there is also the interactive protocol standard in FIPA. The knowledge-level communication research between Agents is a relatively mature part. But, the interactive protocol research is separated to describe from Agent solution. There is no illustration about the relation between the two parts in FIPA.

AP is the basic platform where Agent can live. It is composed of the machine, the operation system, Agent support software, the management elements (DF, AMS and MTS) in FIPA and many Agents. The inside design of AP is the thing of the developers of Agent and system. One AP is not necessarily one host computer, or may include single process with many Agent threads, or may be a distributed platform, or a standard of the open middleware.

The software in the diagram 4.1 is the software entities not belonging to the MAS of FIAP. These software entities are also the instruction set that Agent can call directly. Agent can add new services through running other software to gain the new communication protocol, the new security algorithm and so on.

The Agent cooperation model proposed by FIPA is the function abstract of the general multi-Agent coordinated system, has important reference value toward the solution of the Agent cooperation problem , different Agent system interoperability and the interactive process in opening environment. It defines the most basic function module needed by the Agent cooperation system. Generally speaking, the multi-Agent system in any open environment must have the function like these module's ones so that it can realize the heterogeneous Agent cooperation in true sense. The ontology as one kind of data shared data model is a preferable solution of the cooperation among heterogeneous computer system. Therefore, this dissertation will study the ontology based Agent cooperation model in the open environment following the FIPA standard.

4.3 The Frame of Ontology Based Agent Cooperation Search Model

The frame of ontology based agent cooperation search model is shown in Figure 4.3.

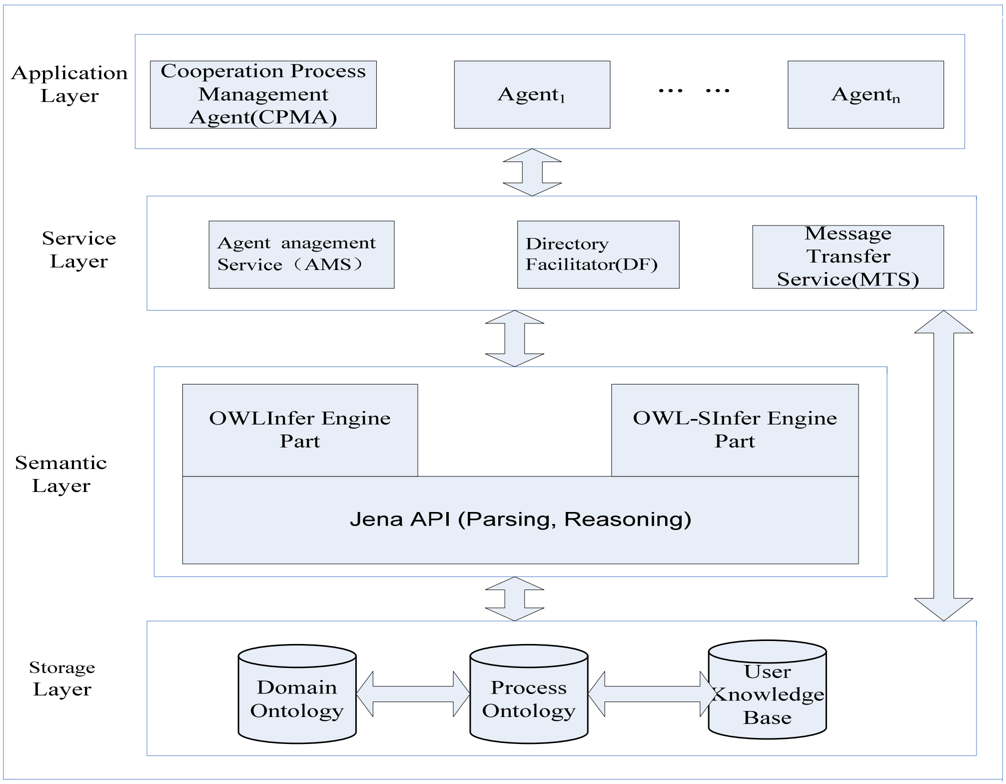


Figure 4.3 Frame of ontology based agent cooperation search model

The frame includes four layers: storage level, semantic level, service level and application layer. The service level and the application layer of the model constitute the Agent cooperation basic cooperation space. The cooperation space provides the Agent with the facilitator of issuing its ability and seeking the appropriate cooperator. The cooperation space also provides the unified message transmission service (MTS) to transmit the message for Agents on the Internet. The storage level provides the resources for the cooperation space, including domain ontology, process ontology DLOP and the user's knowledge library, which form the physics storage medium in the database or document form. The semantic level parses the storage level and provides function interfaces to the upper layer in a modular form. The detail compositions and functions of each layer will be described the following text.

4.3.1 Storage Layer

The storage layer mainly includes the domain ontology, process ontology DLOP and the user knowledge library. The domain ontology describes domain knowledge in the real world, which can be taken as the abstract world model and the semantic foundation on which Agent exchange each other. Process ontology DLOP also describes the Agent problem solving process, resources description way, cooperation process, Agent goal and ability knowledge and so on. It is the dynamic process compared with the domain ontology. The user knowledge library includes some constraints and prejudices of the user definition and the application domain. The relations between them are: The process ontology is at the core position in the storage level; the domain ontology describes the domain concepts and the assertion formula collection in the process ontology, but the user knowledge library is the auxiliary explanation of the process ontology.

The ontology in the store layer can be represented by the OWL (Ontology Web Language). OWL is a structural language with strict grammar, form and inference system. The domain ontology and process ontology DLOP can be stored on the file server in the OWL document form, which can be accessed in the form of URI resources. OWL is one kind of structural language.

4.3.2 Semantic Layer

The semantic level provides the interface which is used to parse, analyze and infer the domain ontology and the process ontology. At present the JENA package developed by HP Corporation Semantic Web Lab (Semweb, 2006) provides RDF, RDFS, and the OWL parse and infer function. The Jena processing RDF description basic flow is to read the RDF description document and the corresponding ontology document through the document read-in interface, then to structure and merge the figure through the graphical interface, to apply the inference engine which the Jena platform owns, finally to obtain the RDFS semantics closure figure. We may use Jena API to realize the access to OWL document.

Based on Jena, the OWL inference engine module provides various functions such as reading, writing, inquiring and operating the document of the OWL and RDF data type, and executing the inference based on the description logic. The OWL inference engine provides the access to the OWL model and other elements such as the type, the attribute, the individual and so on. OWL model can create, inquire, and delete various kinds of resource in the OWL document. The programmers might operate the document of OWL data type using the methods which OWL model provided. Concrete operations can be viewed in OWL API package and related documents. The operations provided by the OWL inference engine module mainly include the following aspects:

(1) Modeling, i.e. providing data structure (attribute and method) which expresses the OWL ontology and document;

(2) Grammatically expressing, for example, representing the RDF/XML document as a data structure;

(3) Serializing, i.e. writing into the document, for example, writing the local data structure into the RDF/XML documents;

(4) Operating the objects;

(5) Inferring, i.e. providing the formal semantics interface realizing and understanding the language.

The OWL-S framework (Thomas H. Davenport and Laurence Prusak 1998) developed by Information and Network Laboratory of Maryland University, has also provided some data structure to express semantics Web elements, for example, service, process, profile, grounding and so on (referring to the OWL-S API documents). The OWL-S inference engine module parses and infers the ontology according to the OWL-S framework.

4.3.3 Service Layer

The service level includes: AMS, DF and MTS. Among them, the MTS function and realization plan can refer to the FIPA standard, but AMS and DF differ in the function realization that needs to operate process ontology DLOP.

(1) Agent Management Service（AMS）

AMS provides Agent with white page and the running management service. Before its running, each Agent must register in AMS to obtain the running Agent identifier (AID), thus establish the Agent running environment. AMS endows each registered Agent with the unique identifier AID in the cooperation space. AID provides the information needed by Agent about the physical location, such as the physical address, the communication port number and so on. AMS provides the registering and canceling service. According to the FIPA standard, AID is named by AMS according to a kind of certain naming rule, including the only Agent name and the physical address in this cooperation space. AID supplies the physics information to localize this Agent, which is the communication foundation between Agents. AMS also provides the Agent life cycle management in joining the cooperation space. AMS sets each the condition of successfully-registered Agent as Initiated, and then maintains the status information together with Agent based on various operations and movements in table 4.1. AMS deletes the related individual information about the canceled Agent from process ontology DLOP.

(2) Directory Facilitator(DF)

During its cooperation solution, Agent needs to call the directory facilitator to help to seek the appropriate cooperator and the resources, the cooperation knowledge, and meanwhile returns the related information of the cooperator and the resources, such as the question of the Agent requesting cooperation, the cooperation Agent’s credit and so on.

Table 4.1 Basic Services Provided By AMS

Service Name	Input	Action
RegistrationAgent	agentInfo	AMS produces AID, and joins the individual Agent instance of agentInfo into DLOP, and set its CurrentState attribute value Initiated.
(DegistrationAgent	agentInfo	AMS deletes all the Agent individuals from DLOP, which are of the agentInfo.
W h Modify	agentInfo, preState, postState	AMS revises the Agent's condition and changes its CurrentState attribute value from preState into postState in DLOP.

When any Agent calls the DF to issue its ability, it must carries out the uniformity and the collision detection through the OWL and OWL-S inference engine. For example, if there isn't any description about some role in the process ontology, Agent is not able to issue this role ability and so on.

The most basic functions which DF provides can be expressed as DF=<ProcessByGoal, RoleByGoal, AgentByRole>, herein:

1) ProcessByGoal: Agent submits the goal corresponding to its current mission correspondence to DF, and then DF discovers the process knowledge with which Agent can achieve this goal;

2) RoleByGoal: Agent submits the goal to DF, and then DF discovers the role which can achieve this goal;

3) AgentByRole: Agent submits the role's demand to DF, and then DF discovers the Agent which can play this role.

Moreover, DF also needs to provide the function with which Agent can issue its ability. Therefore, the service functions provided by DF may be represented as the following four kinds separately:

1) RegistrationRole (agt, Role)

This function provides the Individual Agent agt the chance of adding its role into its Play attribute in DLOP, which indicates that Agent can play this role, also indicates that Agent can observe this Role standard and hold the ability to play this role, including this Role standard interactive protocol, activity, responsibility, the resource operation power, property rights and so on.

2) DeregistrationRole (agt, Role)

This function provides the individual Agent agt the chance of deleting its role from its Play attributes in DLOP.

3) DiscoveryGoalAssociatedProcess (Goal)

This function is used to find all the processes which can achieve the goal with G-P reference algorithm in DLOP knowledge base, and returns the correlative knowledge, or failed. If no process is found, it means no agreement about how to realize this goal in this present knowledge situation; if many processes are found, they can be chosen according to the liking and restriction in the user knowledge base.

4) DiscoveryRoleAssociatedAgents (R)

This function is used to find all the Latent Agents who can play this role R according to the role demand, and returns to the askers. The Latent Agent Discovery algorithm is as Figure 4.9. CPMA finds the Agent through seeking the Role-Agent in DLOP, and returns the Agent logic name or failed. The asker finds the concrete Agent accurately through looking for the Agent description information in DLOP. CPMA may find one more Agent who completes this role, and CPMA must choose these Agents according to the user knowledge base.

```
DiscoveryRoleAssociatedAgents(R){
    //Input: Checking Role R;
    //Output: Latent Agent set aSet;
    pSet=GetAllPlaySet();
    //searche all  the knowledge related to Play relations from the ontology knowledge library
    FOREACH Pi in pSet
      //travel set gather pSet
      Ri=GetRoleFromPlay(Pi);
      //Take out role R from two tuple Playi
      If(Ri==R){
      A=GetAgentFromPlay(Pi);
          aSet.Add(A);//put A into aSet
      }
    }
```

Figure 4.4 Latent Agent Discovery Algorithm

(3) Message Transport Service（MTS）

MTS is also called ACC (Agent Communication Channel). MTS is a software module which controls the message exchange in the cooperation space and meanwhile it also includes the message transmission in the remote space, and provides the message transmission service for the Agents running in the space.

4.3.4 Application Layer

(1) Agent

Agent participating in the cooperation is an independent, social and pre-active software entity representing the user's benefit and guided by its goal. It expresses the expected state of the Agent with the goal, guiding itself to seek the cooperator to carry out the question solution independently and in advance. In the frame of FIPA, Agent has also the ability and the jurisdiction to maintain the Agent information table and the ability table. But in the model here, these functions provide to Agent in the service form. Therefore, What the Agent does is just to call these service functions. This may simplify the design and function of the agent, and make the user to concentrate on the Agent intelligence design.

(2) Cooperation Process Management Agent（CPMA）

The cooperation process management Agent (CPMA) provides the full-sided management and monitor toward Agent in initiating the cooperation, forming the team, making up the plan and executing. Agent can seek the underlying cooperators through calling DF and AMS and finding DLOP on the cooperation platform. Besides, it can submit the complex cooperation tasks to CPMA, which represents Agent to seek the cooperators. The advantage of doing this way is: on the one hand, because Agent did not know more completely the model than CPMA, it uses CPMA to seek the cooperators to gain optimization cooperation schema; on the other hand, as it may be not necessary for Agent itself to complete its task, and CPMA shields the cooperation initiators, the benefit dispute between Agents can be avoided.

CPMA provides the window from which we can see the Agent's problem solution and cooperation operation in the cooperation space, and takes up the responsibility like the network electronic police. It has the very vital significance to the standard behavior on Internet and advocating an honesty society. When the new cooperation team carries out the cooperation in the cooperation space, CPMA starts up a new thread to track the cooperation case, and displays the cooperation solution process with the visual diagram. Agent backups in CPMA after sending messages in the cooperation, and at the same time, should backup the cooperation control flow in CPMA. So, CPMA can show the dataflow and control flow of the cooperation, and CPMA tracks the cooperation case in time. Once there are exceptions in the cooperation, CMPA can respond in real time. After finishing the cooperation, the corresponding monitor thread in CPMA is destroyed. CPMA may make the judgment of the cooperation situation and write in the user's knowledge library, which can be taken as the reference used in the next cooperation

Besides, CPMA undertakes the task of maintaining the domain ontology and the process ontology through calling components of the semantic layer. CPMA supplies adding, deleting and modifying the resource to the managers as the configurable view.

4.3.5 Compared with the Frame of FIFA

The difference between the Agent cooperation model and the FIPA model is: the process ontology DLOP saves the Agent registration information provided by AMS and DF, including white page information such as the Agent logic name, the physical address and so on, and yellow page information such as ability, interactive protocol, communication language and so on. The DLOP has the explicit semantics, and is organically organized. Therefore, the model has more complete semantic information and better Agent information than FIPA so as to support the Agent cooperation better.

Because the process ontology DLOP saves the description about the problem to solve and the cooperation process, and the DLOP role describes the responsibility, the activity and the interactive protocols which can be undertaken, the role may express Agent ability. Therefore, the information maintained by DF is relatively quite simple and can be expressed by the Agent-Role pairs; But the DF table described by FIPA needs to record in detail the protocols, the ontology, the interactive protocols, the language and the promising time and so on of the Agent.

In the frame of FIPA, Agent has also the ability and the jurisdiction to maintain the Agent information table and the ability table. But in the model here, these functions provide to Agent in the service form.

4.4 Design of DLOP

The process ontology DLOP can represent the Agent cooperation framework, the roles, the interactive protocol, resources and activities. When Agent publishes these attributes into the DLOP, it can seek the cooperator, establish and execute the cooperation to reach its goal, through the DLOP. The DLOP is described by OWL language, and stores the problem solution, resource description in the Agent cooperation space. After The DLOP is published into the file server, it is located on Internet to compose the cooperation space. The DLOP supports the Agent cooperation process in the following aspects: Agent registration and logout, Agent capability publishing, Agent seeking cooperators, Agent establishing and executing the cooperation process following the DLOP.

1. Agent registration and logout

Agent joins and quits the DLOP cooperation space through the registration and logout. DLOP is published into the server as the file on Internet. So, Agent can visit

DLOP through URI, such as http://localhost:8080//houseProcessOntology.owl. The Agent registration is to publish its base information to DLOP.

2. Agent publishes capability

Agent represents its capability with the role in DLOP. So, the publishing of its capability in DLOP is to publish the role.

3. Agent seeks cooperators

When Agent can't complete the task alone, it needs to seek other cooperators in DLOP. Here, Agent is guided by the goal. The core elements in DLOP, such as the role, the process, are organized by the goal. So, it is also driven by the goal to seek the cooperator. Firstly agent tries to complete its goal by the inquiring activity. If it is fail, then the agent tries to complete its goal by searching and finding out other Agents which can play the role which completes the goal.

4. Agent establishes and executes the cooperation

The process in DLOP points out the way to solve the problem and the Agent interactive protocol. Agent solves the problem and shares the interactive information, according to the execution standard in DLOP, which is very important to any process in DLOP.

The process of solving problem will be decomposed into atom activity set by means of the parse algorithm of DLOP which is showed in Figure 4.5. Every activity is allocated to some Agents, according to the role. In the end, the control connector specified in DLOP control the sequence of Agent executing corresponding activity. The control flow in DLOP specifies the sequence of the interactive information flowing between Agents. The control algorithm of the cooperation is shown in Figure 4.6. The execution process algorithm of cooperation is shown in Figure 4.7.

```
ParseProcess(Process){
//Input：Parsing Process;
//Result：Each atomic process set AtomicPS;
Process<---ProcessByService(ServiceByGoal(g));
Switch(Process)
Case AtomicProcess: put Process into AtomicPS；return；
Case CompositeProcess:
        construct = Process.getComposedOf();
      switch(construct)
            case   Perform:ParseProcess(construct.getProcess);
            case   Sequence:
            case   SplitJoin:
            case   Split:
            case   AnyOrder:
             case  Choice:
                 components = construct.getComponents();
                 while (!components.isEmpty()) {
                 cc = components.getFirst();
                 ParseProcess(cc);
                 components = components.getRest();
                 }
              case   IfThenElse:
                ParseProcess(construct.getThen());
                ParseProcess(construct.getElse());
            case RepeatUntil:
                ParseProcess(construct.getComponet());
}
```

Figure 4.5 Process Parser Algorithm

```
Control_Co_Process(){
    //Co_Process：cooperation process
    //agt：Cooperate initiator identifier
    //Input：the cooperation input data set of the beginning
    //PreCondition：inital value for the beginning of cooperation world condition assertion formula set
    Control=GetNextControl(Co_Process);
    //inqure process ontology to gain cooperation controllers
    if(Control==null)
     return success; //If completing cooperation, return success and kill control thread
    aP=GetAtomNextProcess(Co_Process);
    // inqure process ontology to gain cooperation task
    Role=GetAssociatedRole(aP);
    Agent= GetAssociatedAgent(Role);
    If(Agent==agt)
    Execute_Co_Process();// If completing the task itself, call  execution process
    Else{
      SendToCo_Agent(Agent,Input, PreCondition, Control);
      //send data to cooperator
      < Input, PreCondition >=waitForCo_Agent(Agent);
    }
    }
```

Figure 4.6 Cooperate Control Process of Cooperative Initiator

```
Execute_Co_Process(){
    //AtomQuee:process queue assigned by process parse algorithm
    While(true){
    //receive task and data from cooperation initiator
    Input=ReceiveDataFromSponsor ();
    PreConditon=ReceiveConditionFromSponsor ();
    Control=ReceiveControlFromSponsor ();
    aP=AtomQuee.pop();
    <Output,Effect>=ExecuteProcess(aP,Input ,PreConditon, Control);
    SendToSponsor(Output,Effect);//After completing task, send result to initiator
    SendToCPMA(Output,Control);//send CPMA backup
    If(AtomQuee. isEmpty()==true)
     Return;//Make ask queue empty ,complete cooperation task  kill cooperation thread
    }
    }
```

Figure 4.7 Cooperation Implementation

4.5 Design of Cooperation Process

The cooperation processes include initiating the cooperation, seeking the cooperator, establishing the cooperation plan, executing the cooperation and monitoring the cooperation in the model.

4.5.1 Initiate Cooperation

Agent in the cooperation space is directed by the goal. After Agent receives the tasks from users, Agent first translates the tasks into the goal, and then find the plan to complete the tasks in the plan base through the out-line planner. If it finds no plan to complete the goal in the plan base, it initiates the cooperation request. The condition of the Agent initiating cooperation request is that it must join in the cooperation space, namely register into DLOP by means of AMS, and obtain only one allocated identifier AID.

Agent has two ways to initiate the cooperation: i)It can be initiated by Agent itself; or, ii) Agent sends its task to CPMA, and then it is completed by CPMA. These two ways have advantages respectively. The second way may decrease the model communication. CPMA becomes not only the cooperation initiator, but also the cooperation main controller. It combines the messages of sending to the initiator and CPMA by Agent. In addition, CPMA will monitor the exception in the cooperation easily. But, the disadvantage of this way is to increase the workload of the CPMA. When there are many Agents needing CMPA in the model, the CPMA will have a large workload, which results in slowing the running of the model. So, under this condition, CPMA can only choose a part of cooperation tasks according to its capability.

4.5.2 Seek Cooperation

Agent in the cooperation space seeks the cooperation knowledge and the latent cooperators. The two basic cases are: i) Agent submits the corresponding goal of the current task to CPMA, which can seek and find the process knowledge to achieve this goal; ii) Agent submits the role demand to CPMA, which find the suitable Agent.

Finally, the Agent might obtain different cooperation schemas, which have different cooperation roles and Agents. These cooperation roles and Agents compose the Role-Agent set which can attain this goal. Agent can choose a part according to its preference, and as a result, the cooperation space will become smaller.

4.5.3 Building Cooperation Plan by Negotiation

After knowing the latent cooperators, Agent can refuse or accept the cooperation request because of its autonomy. So, Agent needs to negotiate the cooperation issue with the latent cooperators. The contents of the negotiation include the cooperation criterion, the interactive protocol, the each other's responsibility and rights and so on.
The result of the negotiation may come to agreement, and then these Agents can form the cooperation team, and each has the explicit responsibility and rights. If the negotiation fails, it means that there are conflicts of interest between them or that the cooperation will not bring the benefit to all the Agents participating in the cooperation.

4.5.4 Execute Cooperation

Agents start executing the cooperation after the negotiation comes to agreement. As the cooperation main controller, the cooperation initiators need to read the control connector in DLOP to control the cooperation process except for undertaking its own tasks.

Agent starts a new service thread when joining in the cooperation, which has a task queue of the atom processes making up the solution of the problem. The execution of the atom processes is initiated by the cooperation main controller. The communication content between Agents includes the data set and the assertion set. The assertion set composes the states of the current world, and the data set is the input and output of the atom process.

The communication message between Agents is firstly sent to the main control Agent, and then the main control Agent decides to send it to which Agent according to the control flow in DLOP. Agent receives the message and executes the atom process.

4.5.5 Monitoring Cooperation

The cooperation execution process is composed of the activities with connectors, and the activity, as the smallest process cell, can't be partitioned any more. The activity is carried out by one role, and its inner part is not involved in the interaction and cooperation between Agents. The same activity name has the same implementation in different Agents. The cooperation monitor here is only limited to the activity layer.

Agent sends its data and state, i.e. IOPE, to CPMA during the cooperation. CPMA evaluates the progress situation of the activity through looking over IOPE to realize its cooperation monitoring of the activity layer. At the same time, DLOP uses the connector to control the cooperation process. So, CPMA can monitor the cooperation process by looking over the connector in DLOP.

When Agent fails to execute the activity, CPMA concludes that the activity can not be finished by this Agent, and then call the AMS to find the Agents which can finish the task. If there is no Agent who can play this role to execute this activity, CPMA should look for DLOP upwards, and call P-R reasoning to find out other roles which can execute this activity, and then find Agent according to this role. If there is no Agent which can execute this activity, CPMA should look for DLOP upwards, and call G-P reasoning to find out other activities which can replace this activity, and then inform this initiator. Otherwise, if it shows that in the current model, exception appears in this cooperation, it returns Fail. The CPMA monitor management algorithm is showed Figure 4.8.

In a word, the monitoring cooperation of CPMA includes the whole monitoring of the data and control in the cooperation. According to the exception situation, CPMA can make up some remedy measure of the realizing of the goal. This characteristic might increase the security and the reliability of the network, which is very important to the open and reliable cooperation of Agent, and ensure that network system to work and succeed to the greatest extent.

```
CPMAInspectManagement(co_Process){
//Input：A specific cooperation process co_Process；
//Output：Monitoring management strategy；
While(true){
DataSet=GetDataFromAgent(agt);
ControlSet=GetControlFromAgent(agt);
//Obtain the data and the controller from cooperation Agent
<DataSet0,ControlSet0>=GetControlFromDLOP();
//Obtain the data and the controller from process ontology
tag=Compare(<DataSet0,ControlSet0>,< DataSet,ControlSet>);
if(tag==false){
  //If cooperation come to exception ，CPMA processes exception
  R=DiscoveryAgentAssociatedRole(agt);
  agts=DiscoveryRoleAssociatedAgent(R);
  //Seek the cooperation replace
  if(agts!=null)
     InformSponsorAgent(agts);
     //Notice replacable cooperation initiator Agent
  else{
    Goal=GetGoalFromSponsorAgent();
    pSet=DiscoveryGoalAssociatedProcess(Goal);
    //find replacable cooperation process
    if(pSet!=null)
        InformSponsorAgent(pSet);
     else
       return fail;//If two measures not success, then cooperate defeat
     }
}
if(ControlSet==success)
return success;//Monitor the cooperation  process unitl finish, then return cooperate successfully, simultaneously destroy kill the cooperation monitor thread
}
}
```

Figure 4.8 CPMA Monitoring Management Algorithm

4.6 Experiment

4.6.1 Experiment Scheme

In order to verify the feasibility of the ontology based agent cooperation search model, this dissertation designed the following experiment.

In this experiment, the supposed scenario is as follows: a real estate company gained a piece of information that a land will be publicly sold by the local government. Before making the decision whether or not attend the auction to gain the land, the real estate company needs to do the feasibility study about the land. During the feasibility study, the real estate company needs to make some preliminary plans regarding how to best develop the land, gain corresponding cost information, and finally estimate how much profit it can earn from the real estate project. When it makes plan, it needs to hold some necessary knowledge, some of which must be acquired by communicating with other organizations such as architecture design institutes, construction companies, housing sale agencies and property management companies. The problem can be solved by means of the agent cooperation search model proposed in this chapter.

According to the knowledge requirement of the real estate project feasibility, for simplicity, six agents are designed in this experiment, including RealEstateCompany, ArchitectureDesigningInstitute, ConstructionCompany, HousingSaleAgency, Property-ManagementCompany and CPMA.

1. The RealEstateCompany Agent represents the real estate company. It is the main controlling unit of real estate projects. Its work includes land obtaining, designing entrusting, construction entrusting and housing sale entrusting. It needs numerous kinds of knowledge during the whole life cycle of real estate project. In the experiment, we focus on its feasibility study for which other agents provide the support in knowledge acquisition.

2. The ArchitectureDesigningInstitute Agent represents an architecture designing institute. Its work is to design the structure of houses or buildings and delivery the construction draws which meet the requirements of real estate companies. It owns rich knowledge in architecture designing.

3. The ConstructionCompany Agent represents a construction company. Its work is to construct the houses or buildings and delivery them to the real estate company after finishing them. It owns a lot of knowledge in how to construct houses or buildings.

4. The HousingSaleAgency Agent represents a housing sale agency. Its work is to help the real estate company to sell the houses or apartments of buildings. It owns a lot of knowledge in how to sell the houses or apartments of buildings.

5. The PropertyManagementCompany Agent represents a property management company. Its work is to manage the community including the maintenance of houses or building, the collecting of property management fee, etc. The customer to whom it faces is the owner of housings. It owns a great deal of knowledge in what kind of housings is popular to people, and what problems most possibly occur while housing are used, etc.

6. The CPMA is in charge of collaboration amongst the above agents during knowledge cooperation search process.

Besides these six Agents, according to the above model, there must be still the public facilities in the open dynamic environment, including: the sharing ontology (the domain ontology, the process ontology-DLOP) and OS(sharing ontology service)providing the parsing and reasoning of the ontology file; MTS (Message Transport Service) providing the message communication language; AMS（Agent Message Service） calling DLOP and in charge of the Agent registration and logout; DF（Directory Facilitator）calling DLOP and in charge of publishing and destroying the Agent capability; The user knowledge library with some preference and restrictions related to the application domain. The above content can be located in the CPMA.

In addition, Agents except CPMA have also own ontology knowledge base for supporting their business work and providing knowledge for other agents.

The Architecture of the whole system is shown in the following figure:

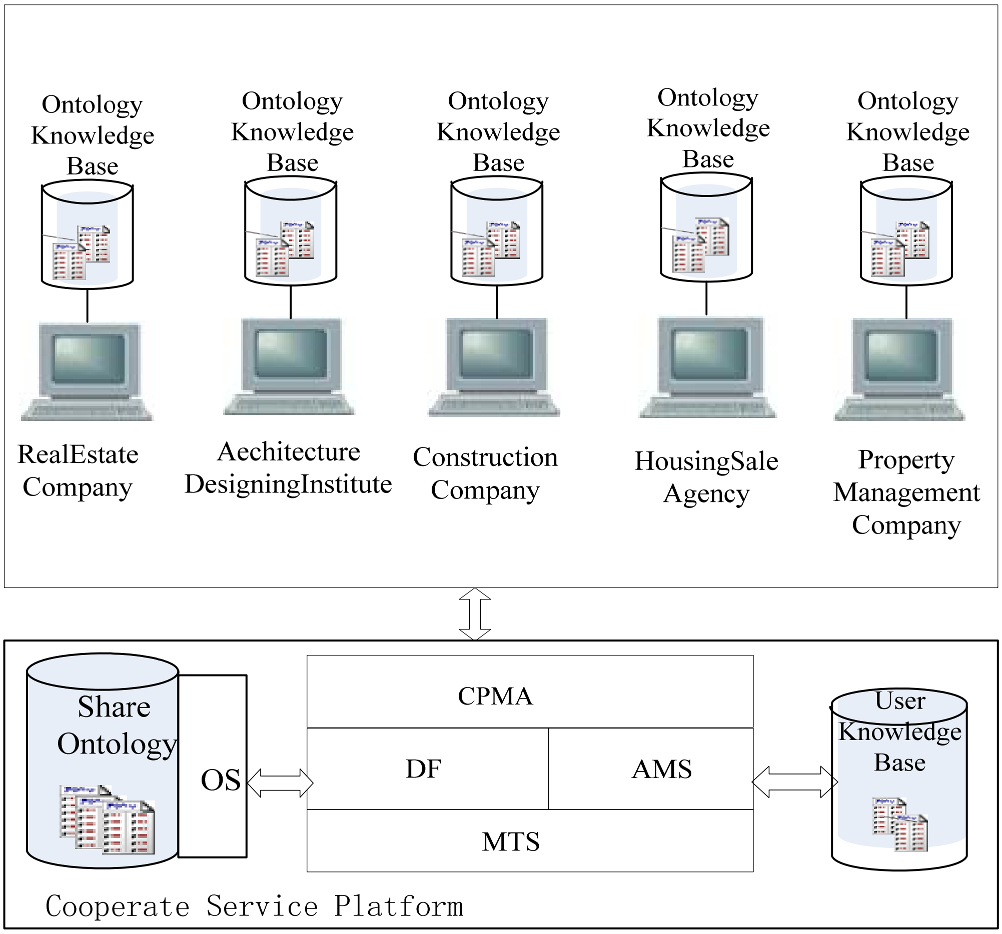


Figure 4.9 Architecture of Real Estate Project Knowledge Cooperation Search System

4.6.2 Process of Real Estate Project Knowledge Cooperation Search

The process of real estate project knowledge cooperation search will be illustrated in the following:

1. Initialization of CPMA

Before creating the real estate knowledge cooperation search, CPMA must be started up. After starting up it, some parameters of CPMA must be set and so other Agents can be registered in CPMA. The initialization parameters setting interface of CPMA is showed as the following:

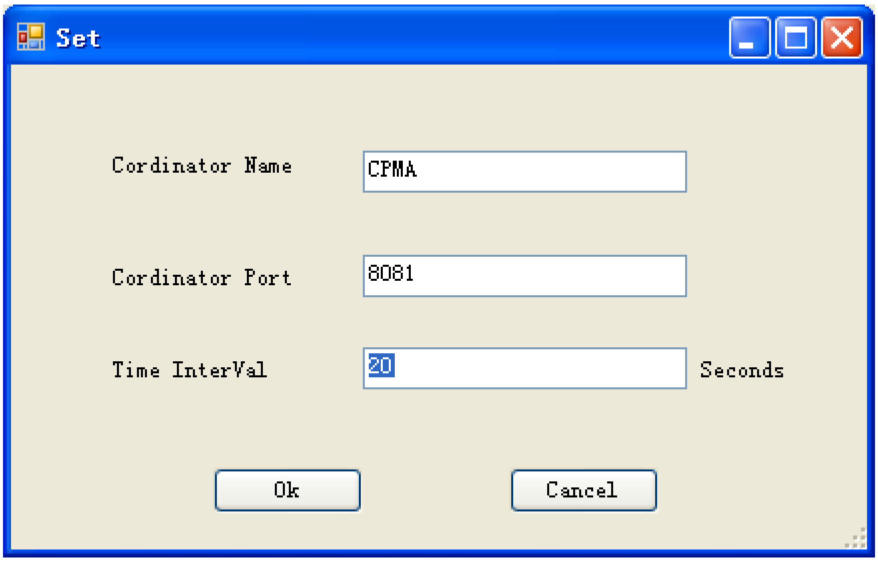


Figure 4.10 Parameters Setting Interface of Coordinator CPMA

In the above interface, the name in network, the port number for communicating with other Agents, the time interval of the system to ensure the system's clock unification can be set.

2. Registration of Other Agents

During the registrations of other Agents, besides Agent’s name and port, the IP address and port of the CPMA must be specified. The following chart is the interface figure of the registration of the RealEstateCompany Agent.

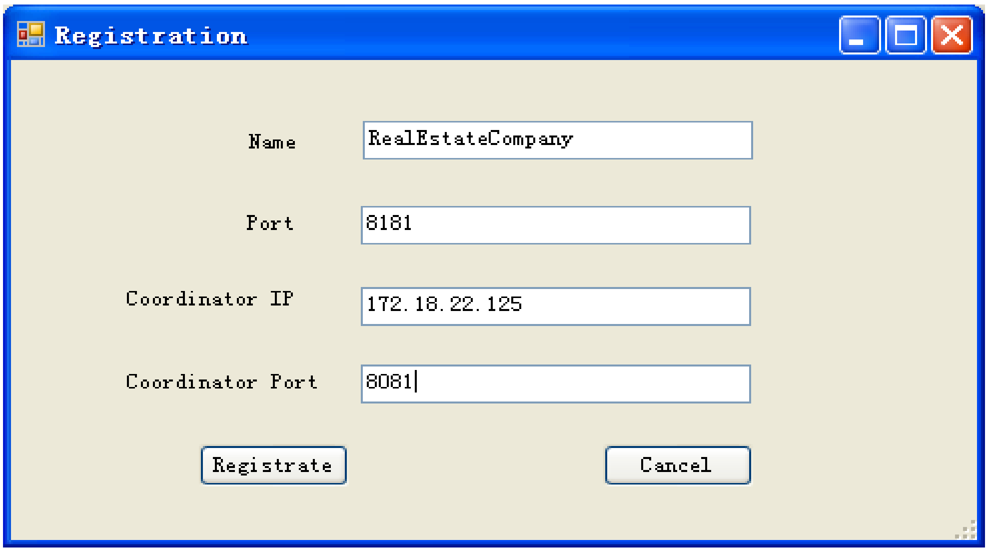


Figure4.11 Registration Interface of RealEstateCompany Agent

3. Capability Publishing of Other Agents

As for any Agent except CPMA, it must let CPMA know what knowledge it can provide before starting to cooperate with other agents. The thing is also called the capability publishing of agents. After agents finish its registrations in CPMA, it may realize its capability publishing by two kinds of means. One is by choosing the role from predefined role list in CPMA. Another is to define a kind of new role which is suitable for it. Once this new role is defined, it will be also added to the predefined role list in CPMA. The following figure is an example of new role defining.

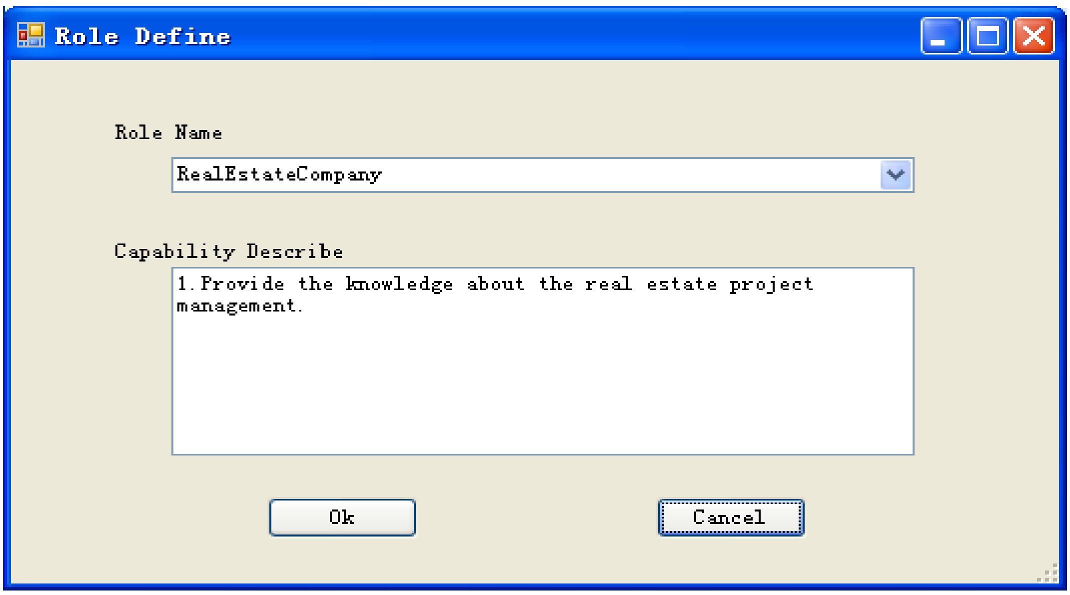


Figure4.12 Role Defining Interface of RealEstateCompany Agent

After all the Agents have been registered, the result can be viewed through the CPMA's Agent List online function in CPMA, as showed in the following figure.

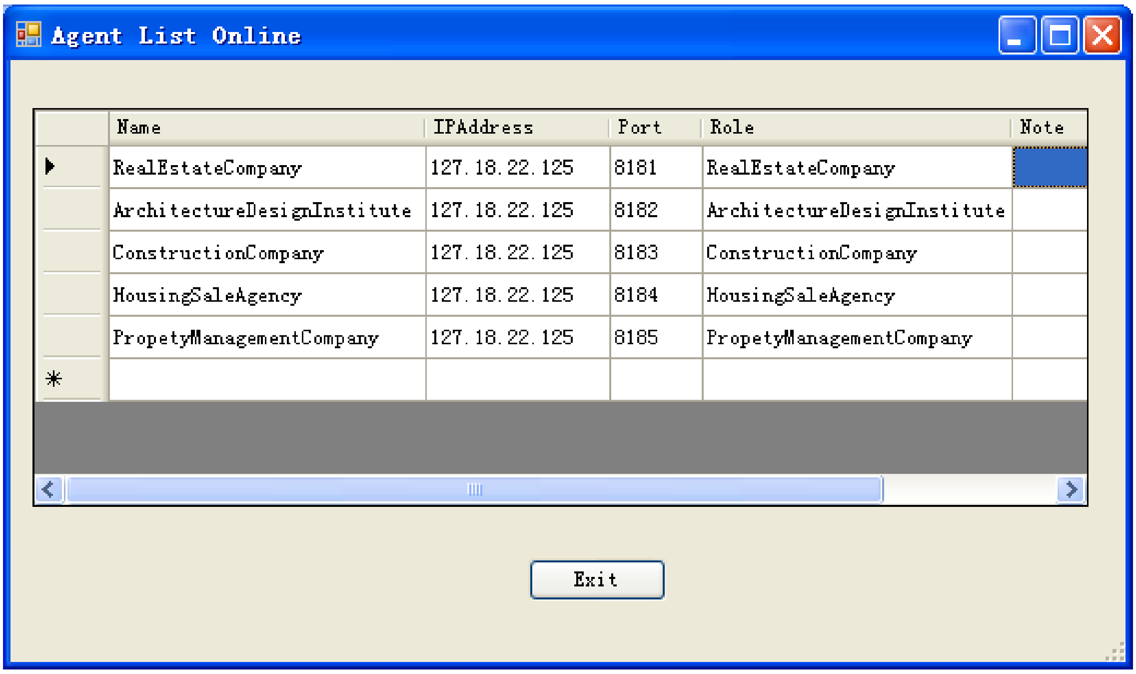


Figure 4.13　Agent List Online Interface of Coordinator CPMA

4. Request for knowledge

When the real estate company agent wants to acquire some special knowledge from other agents, it may send a request for the special knowledge to the CPMA Agent. The CPMA Agent decides which agent to respond to the request for the special knowledge according to the capabilities of registered agents. The interface of sending a request for the special knowledge is shown as follows.

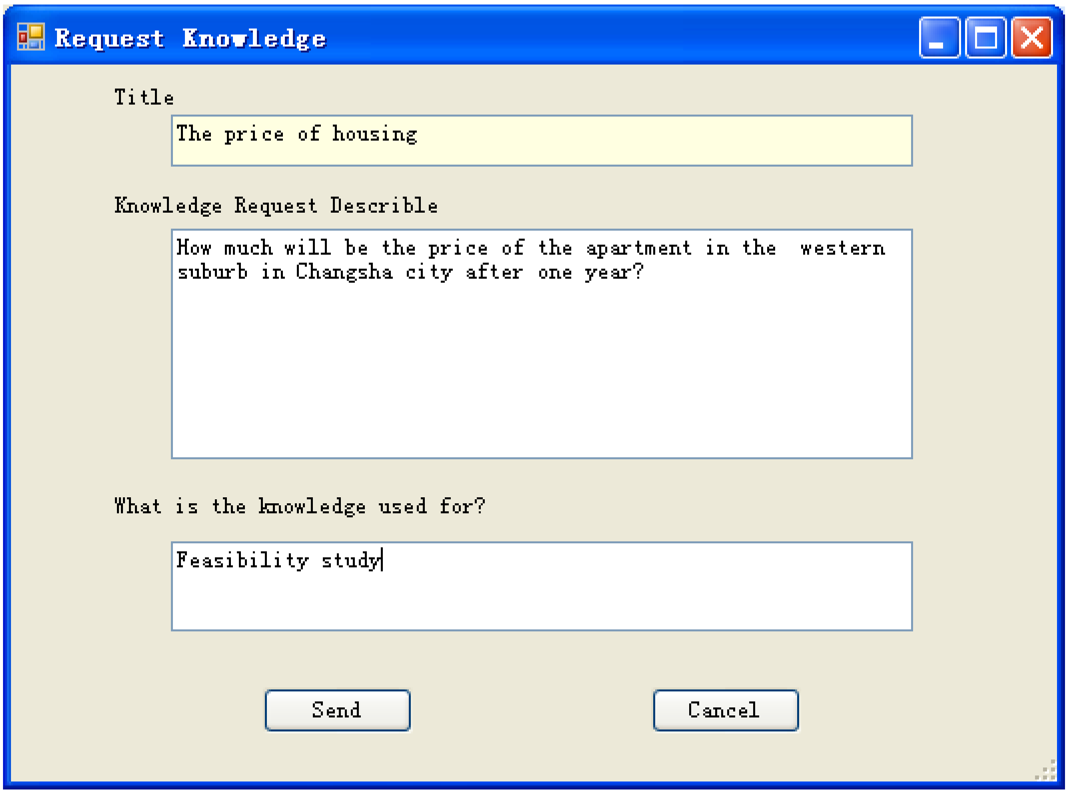


Figure 4.14 Knowledge Request Interface of RealEstateCompany Agent

5. Respond to the Request for Knowledge

When some agent receives a request for knowledge, it may reject the request or accept it. After it accepts the request, it may employ two kinds of handling mode. One is to automatically give the response to the request by means of the reasoning function based on ontology knowledge base. Another is to manually give the answer. Of course, the latter may be implemented under the help of the ontology reasoning. An example of the responding interfaces to the request for knowledge is showed as follows:

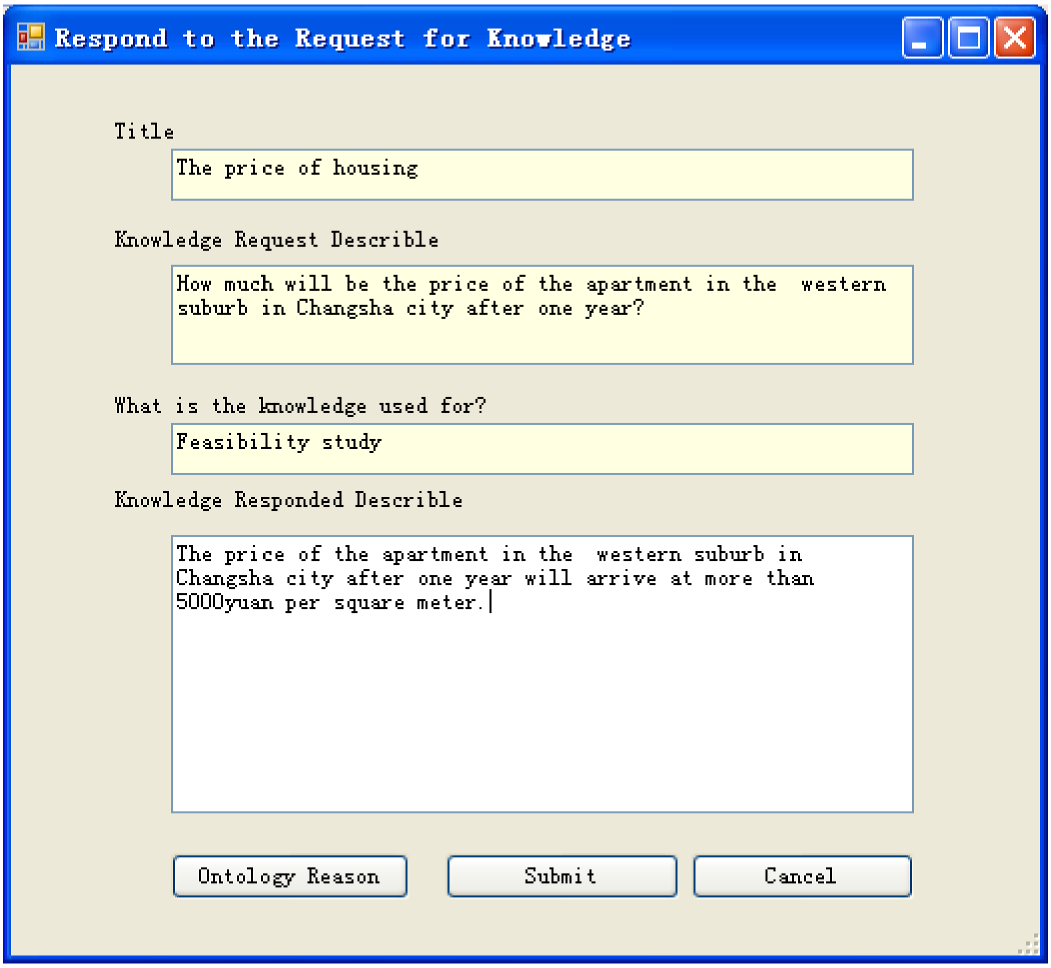


Figure 4.15 Responding Interface to Knowledge Request of HousingSaleAgency Agent

6. Monitor to the Cooperation Process

During the cooperation acquisition of real estate knowledge between agents through CPMA, the cooperation state of the whole system can be viewed under the inspect function in CPMA like the following figure. In the figure, cooperation state is described by the line with direction and color between agent and CPMA.CPMA is located in the center of the state window. The blue line with an arrow pointed to CPMA means that the agent has sent at least a request for knowledge, but it has not received the response to the request. The green line with an arrow pointed to some agent means that the agent has received at least a request for knowledge, but has not sent the response to the request. The red line with two arrow means that it has not only sent at least a request which has not be given the response, but also received at least a request which has not be given the response.

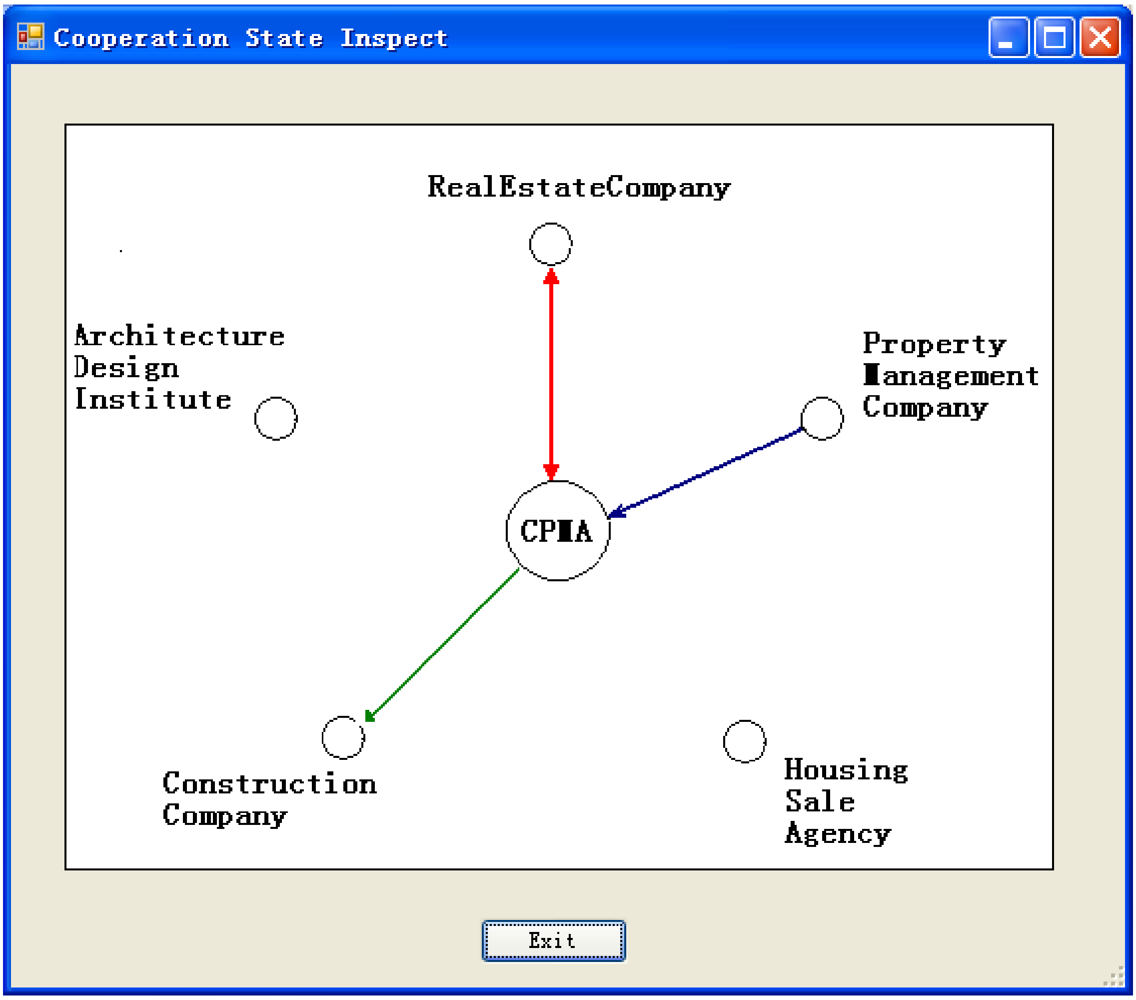


Figure 4.16 Agent Cooperation State Inspect Interface of CPMA

The detail cooperation state of a special Agent can be viewed by pressing the circle area of the Agent. The following figure is an example of the detailed cooperation states of a special Agent.

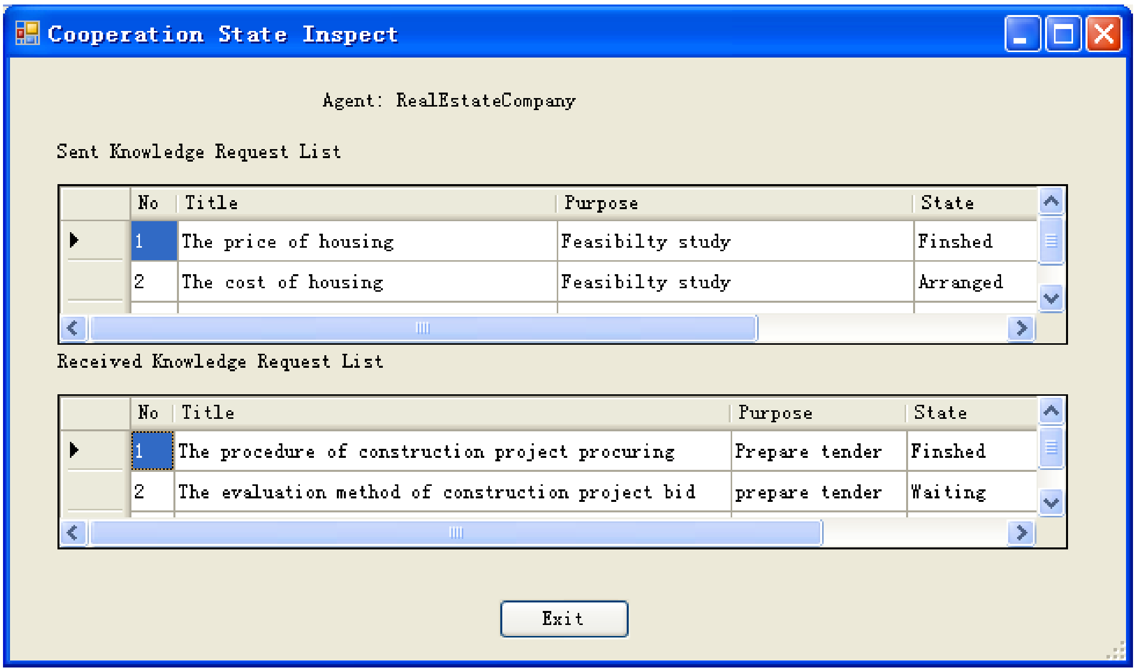


Figure 4.17 Agent Cooperation State Inspect Interface of　RealEstateCokmpany

7. Ontology Modeling

As said early, every Agent have own ontology knowledge base. The ontology knowledge base can be managed by means of the protégé tool which is a very popular and famous ontology modeling tool and developed by Stanford University. The operation interface of the protégé tool is showed as follows:

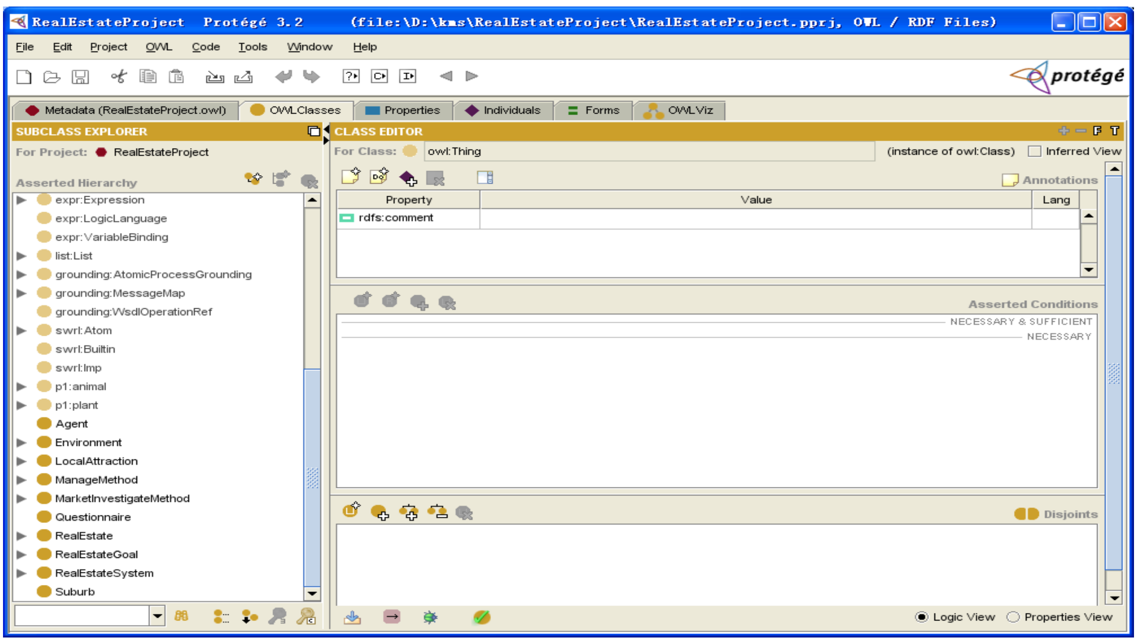


Figure 4.18 Operation Interface of the Protégé Tool

8. Log out

When the agents do not intend to continue taking part in the cooperation, it also logs out from the system.

4.7 Conclusion

Firstly, this chapter presents the architecture of the Agent cooperation search model in the Internet environment. This model can be divided into the following layers: storage, semantic, service and application. It lays out the plan by which each layer supports the Agent.

Secondly, it brings forward the cooperation management technology of this model, and designs the algorithms of the DLOP and the processes of the cooperation search: i.e. initiating the cooperation; negotiating and establishing the cooperation plan; executing the cooperation; and monitoring the cooperation.

Finally, it demonstrates the employment of the ontology based agent cooperation search model by means of a real estate knowledge cooperation search example. The example validates the feasibility and superiority of the Agent cooperation model in helping the real estate enterprises to acquire knowledge under an open environment.

CHAPTER5

GENETIC ALGORITHM BASED DATABASE KNOWLEDGE DISCOVERY

5.1 Introduction

As is the case in so many businesses, burgeoning IT technology is being applied extensively in the real estate industry. Nowadays, many real estate enterprises have developed extensive management information systems (MIS). For example, almost all enterprises own their house sale system, and some enterprises have started to use project manage systems.

These systems have become an integral part of enterprise facilities, to the extent that some work cannot be performed without these MIS systems. Through many years of running such systems, a great deal of historical data has been accumulated in their databases. This data is a potentially valuable resource for the real estate enterprise because of the wealth of knowledge embodied in it. Unfortunately, this knowledge cannot be utilized directly. This problem is not unique to the real estate industry. In order to help enterprises exploit this implicit knowledge, methods of knowledge discovery from databases (KDD), which cater to this kind of demand, are the focus of much innovative research.

In the real estate industry, KDD can provide a much useful knowledge relevant to house salesmen, home designers, decision-makers, etc. As seen in Chapter 2, Literature

Review, there are many KDD methods. In this dissertation, genetic algorithm based knowledge discovery methods will be explored for their application in the real estate industry.

5.2 The application prospect of KDD in real estate industry

KDD provides the capability to discover new and meaningful information by using existing data. KDD quickly exceeds the human capacity to analyze large data sets. The amount of data that requires processing and analysis in a large database exceeds human capabilities, and the difficulty of accurately transforming raw data into knowledge surpasses the limits of traditional databases. Therefore, the full utilization of stored data depends on the use of knowledge discovery techniques.

The real estate enterprise decision-makers can mine the knowledge helpful to the enterprise decision-making from the database through the knowledge discovery. They can obtain the following information:

(1) Know and extract the customer's true demand

The demand of the customers is not invariable. Although the enterprise makes an all-out effort in certain aspects and meets the certain demand of customers, the demand of the customers has perhaps changed and they propose a higher demand to the enterprise, and the enterprise has not really understood it in time. So, it can't enhance the satisfaction degree of the customers truly, and strengthen the competitive power of the enterprise.

(2) Improve the customer's loyalty

To keep the customers of an enterprise and guarantee that they are not attracted by competitors, it must firstly know its customers' characteristics, their behavior custom and preference and what causes the old customers to leave and how to keep them. Some data shows that the cost of attracting the new customers is 5 times as the cost of satisfying the existing customers, because it needs to pay more efforts and cost to attract customers from the competitors.

(3) Seek for the valuable important customers

Many enterprises can not judge among customers who are valuable, and who aren't. They don't know who may leave, and who want to receive new products. This is resulted of the market subdivision in fact. The database knowledge discovers methods can be used to analyze the history data in the database to seek for these important customers and help to make out attractive strategies to deal with these different customers

(4) Mine the valuable relations between variables

The relations between variables can be mined by means of KDD methods, which is very valuable towards the cost and time estimation, profit analysis and etc.

5.3 The Principle of the Genetic Algorithm

The essential factors composing the basic genetic algorithm mainly are: chromosome code, individual adaptability estimate, genetic operator (choice, overlapping, and variation) as well as genetic parameter setting and so on.

(1)The method of the chromosome coding:

The basic genetic algorithm is to use the binary string with fixed length to represent the individuals in the colony. Its equipotent gene is composed of the symbol set of two values {0, 1}. The genetic code of each individual in the initial colony is produced by the random number with the uniform distribution. For example, X=100111001000101101 may express an individual, whose chromosome length is n=18.

(2)The evaluation of the individual fitness:

The genetic algorithm decides the probability of each individual who inherits from the current colony and passes to the next colony according to the probability in direct proportion to individual fitness. To calculate this probability correctly, it is required that the fitness of all the individuals should be positive or zero. Thus, we must first predefine the transition from the goal function value to the individual fitness according to different kinds of problems, and especially must predefine how to process the negative goal function value.

(3)The basic genetic algorithm uses the three following genetic operators:

- Selection operator uses the proportion operator;
- Crossover operator uses choice operator with the single point;
- Mutation operator uses the basic bit or equal mutation operator.

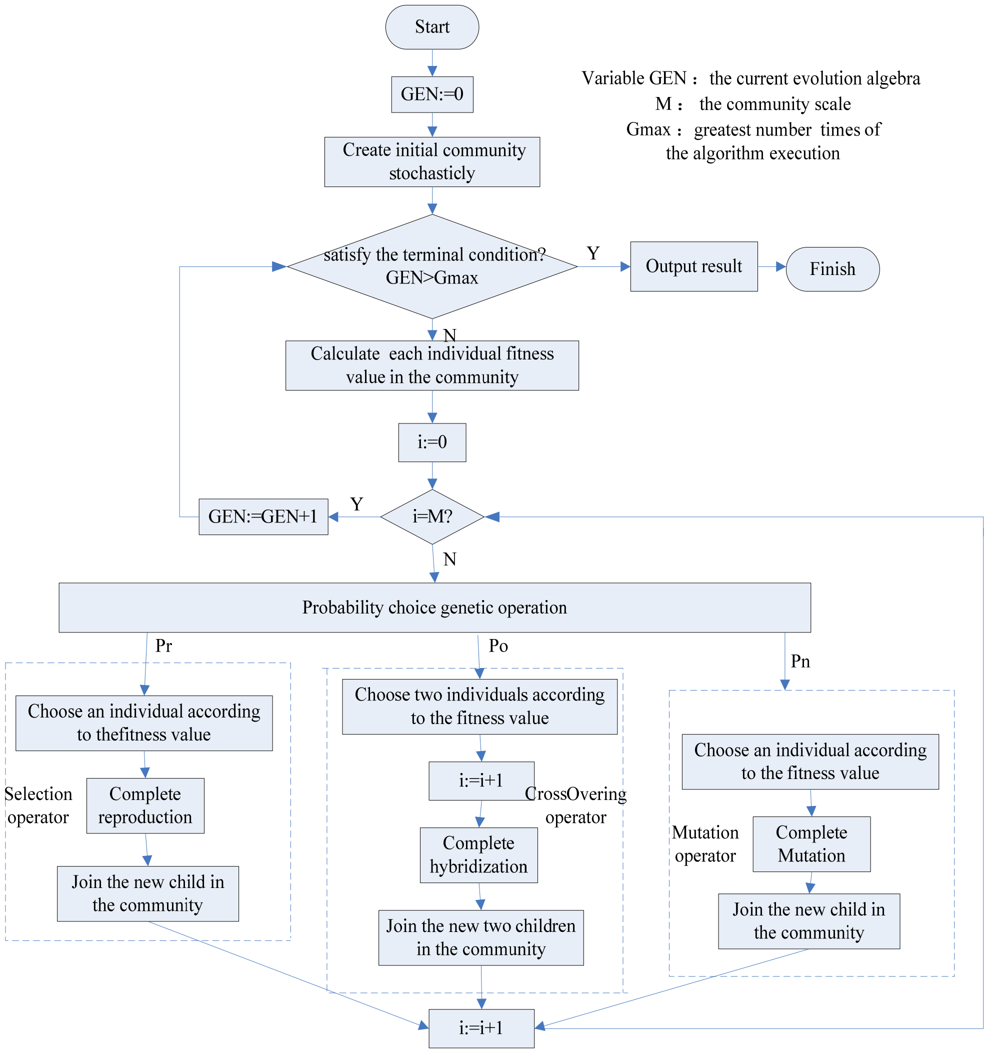
(4)The running parameters of the basic genetic algorithm:

The basic genetic algorithm should set four running parameters ahead;

- N: colony scale, namely the individual number in the colony, is generally from 50 to 100. Because the initial colony is the springboard of the excellent search of the genetic algorithm, the larger the colony scale, the wider the search scale, and the longer the genetic operation time of each generation. On the contrary, the smaller the colony scale is, the smaller the search scale becomes although the genetic operation time of each generation is reduced.

- Gmax: The terminal evolution algebra of the genetic algorithm, is: generally from 200 to 500;
- Pc: The crossover probability, is: generally 0.2~0.6;
- Pm: The mutation probability, is: generally 0.005~0.01.

The flow of the basic genetic algorithm is shown in the Figure 5.1. It produces new individuals through the increasing iterative heredity. In doing so it simulates three ways (reproduction, crossover, and mutation) of the biological genetic law in the biosphere, and then eliminates individuals through the objective function judgment. The program runs certain number of times and returns appropriate individuals as the solution of the algorithm.



Figrue 5.1 Genetic Algorithm Flow

We know from the above flow chart of the genetic algorithm: the core of the genetic algorithm is to construct the coding representation and the fitness function of the initial individuals. The individual coding representation directly affects whether the problem can be represented by using the appropriate mathematical model, whereas the fitness function affects whether the appropriate individuals can be chosen in the selection of each generation. These two points are indispensable in genetic algorithm. The users of the genetic algorithm need to design them in detail in application.

5.4 Genetic Algorithm Based Database Knowledge Discovery

The database knowledge discovery is used to obtain the latent relations from the related data items in the numerous databases through the classification or association rules, so as to obtain the related knowledge. Here, the database knowledge discovery system model based on the genetic algorithm will be introduced in detail.

The database knowledge discovery system model based on genetic algorithm in this dissertation referred to CRISP-DM standard model proposed by the European Standard Committee. This model is composed of six steps, in which some steps may iterate unceasingly. It shows in the following chart.

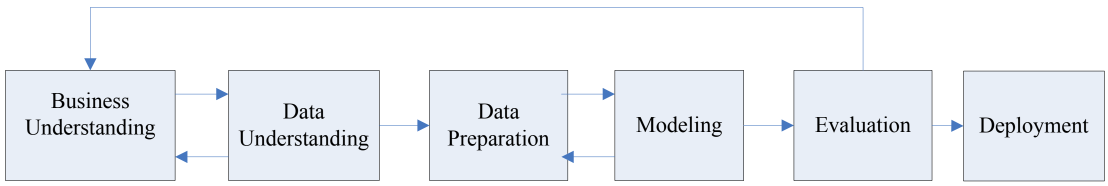


Figure 5.2 CRISP-DM Model

(1)The business understanding stage

First it is necessary to make clear the goal which the commercial application must reach, then make the evaluation of the application resource and the risk, and then determine the objective of the data mining, finally produce the project plan;

(2) The data understanding stage

The job of this stage is to collect the related data, then describe them, then check them and finally confirm the data quality. Each step produces the corresponding report, and its goal is to understand the essence and quality of the data.

(3) Data Preparation stage

The jobs of the stage include data selection, data cleaning up, data synthesis, data merging, data formatting and so on. Herein data selection includes attribute selection, attribute synthesis, example selection and so on.

(4) Modeling stage

According to the data and the nature of application, first choose the appropriate modeling algorithm as well as the algorithm parameters. After choosing the algorithm, design the suitable testing plan and the model training one.

(5) Evaluation stage

After finishing the training, use the test plan to evaluate the obtained model. The evaluation standard differs with the application type. If the result is not satisfied, then return to the preceding stage or even to make a fresh start, otherwise go to the next stage.

(6) Deployment Stage

After obtaining satisfactory evaluation effect, it can enter the deployment stage. The job of this stage is mainly to plan how to deploy this model including the model maintenance and the effect surveillance, and produce the final project report.

Based on the CRISP-DM model, the basic structure of the database knowledge discovery model based on the genetic algorithm is proposed here, which is showed in the following chart, and is also an innovative work of the dissertation.

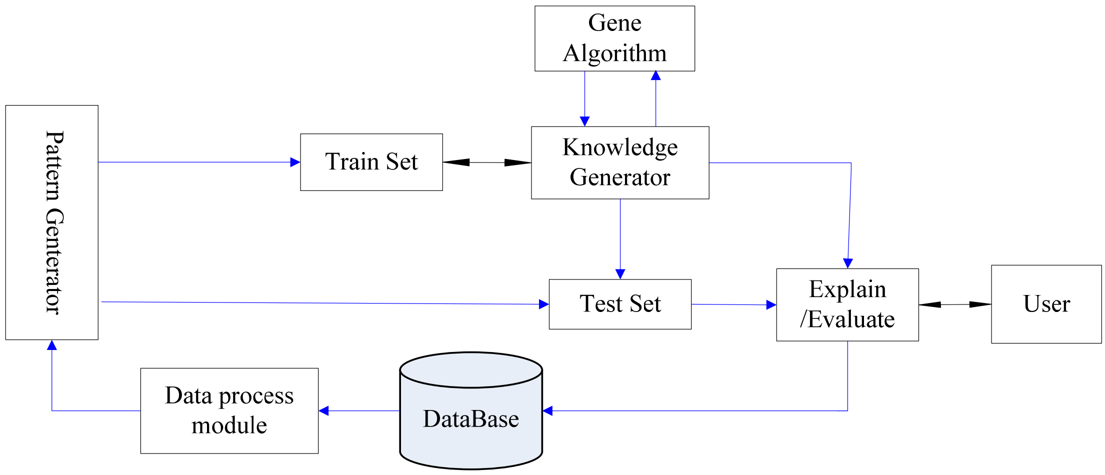


Figure 5.3 Database Knowledge Discovery Model Based on Genetic Algorithm

According to the model, the data in database is sent to the pattern generator through the data process module, and the pattern data generator extracts the referred pattern data according to the appointed data mining tasks and divides it into the training data set and the test data set. The triggered knowledge generator mines interactively the knowledge from the training set be means of the designed genetic algorithm, and provides the satisfactory result for the test set. The test set will submit the explanation/ evaluation organization for the evaluation. Evaluated knowledge is finally submitted to the user.

The function of every part of the model is as follows:

- Data processor: it mainly completes the pretreatment work of data of the database.
- Pattern data generator: it completes the task according to the data mining system, extracts the related pattern data, and divides the data into the training data set and the test data set.
- Knowledge generator: it is responsible for calling the genetic algorithm and computes the individual fitness value of various populations in the genetic algorithm. It actually evaluates the effectiveness of the produced knowledge.
- Genetic algorithm: According to the individual fitness value it defines overlapping, reproduction and mutation, and forms the next generation

population with better average fitness value. The astringency should be guaranteed in designing the genetic algorithm.

- Explanation/evaluation: it translates the produced knowledge into the one that the user can understand. For example, the knowledge generator use the genetic algorithm to produce the decision tree, which the explanation/evaluation module ought to be able to transform it to the corresponding pattern sentence, such as If…Else…Then.

5.5 Experiment

In order to verify the validity of the above knowledge discovery method based on genetic algorithm, I applied the above knowledge discovery method based on genetic algorithm, to discover some knowledge from a business database of a real estate enterprise. The dissertation will introduce the database, process and result of the experiment.

5.5.1 The Real Estate Project Sale Database

Hunan Tongrenzhiye Real Estate Limited Corporation, which I have worked for, has been paying a lot of attention to the application of IT since its foundation. It has bought and deployed a few MIS, one of which is the sale management system of building. The sale manage system has stacked a great deal of sale record data through years of its using. In order to help make up the better sale policy, salesmen want to discover some helpful knowledge from the previous database. There are mainly four tables related to houses sale in the database. Their structures are as follows:

Table 5.1 Vistors&Customers information

Field name	Data type	Note	Example
CustomerId	Char[20]	Customer's Id number	200805003
Name	Char[20]	Customer's name	Chen Yun
EducationDegree	Number	Customer's education degree	bachelor
Occupation	Char[50]	Customer's occupation	Enterprise white-collar
Income	Number	Customer's income per month	5000
Members	Number	Number of family members	3
Age	Number	Age of Customer	30

Table 5.2 Transaction record

Field name	Data type	Note	Example
Transaction	Char[20]	Transaction Id	2008050001
Date	date	Transaction date	2008-5-1
CustomerId	Char[20]	Customer id	2008050003
UnitId	Char[20]	Unit id	JXJS-10-701
Use	Char[20]	Living or Investing	Living
Price	Number	How much per m^2	3000
ContractId	Char[20]	Sale Contract Id	2008050001

Table 5.3 Unit Information

Field name	Data type	Note	Example
UnitId	Char[20]	Unit Id	JXJS-10-701
BuildingId	Char[20]	Id of building which unit is in	JXJS-10
Rooms	number	Number of rooms	3
Parlors	number	Number of parlors	2
Balcony	number	Number of balcony	2
floor	number	Which floor	7
Size	number	Area	$135m^2$

Table 5.4 Building Information

Field	DataType	Note	example
BuildingId	Char[20]	Id of Building which unit is in	LXJS-10
Location	Char[100]	Location of building	Jinxiujiangshan
StartDate	Date	Date of starting building	2003.5
EndDate	Date	Date of ending building	2005.8
TotalCost	Number	Total Cost	5,000,000
OrientedPople	Char[20]	Oriented People	Medium income
Floors	Number	Number of floors	16
Lift	Boolean	Whether it have lift	Yes
DistanceFromBusStop	Number	The Distance from Bus Stop	800m
DistanceFromSchool	Number	The Distance from School	1000m
DistanceFrom-ShoppingCenter	Number	The Distance from Shopping Center	1500m

5.5.2 Process of Real Estate Project Sale Knowledge Discovery

Step 1: Determining the objective of knowledge discovery and the possible factors of influencing the objective.

The objective of the experiment is to find the characteristic of the customer of most possibly purchasing the units of our real estate project. The work of determining the possible factors of influencing purchasing can be finished by consulting with salesmen of many years of sale experience. Based on many years of sale experience, salesmen know that the factors of influencing the sale of the real estate project may be divided into approximately the customer factor and the real estate project factor. The customer's factors include income, age, education degree, occupation (such as official, enterprise white-collar, enterprise blue-collar and so on), use, etc. The real estate project factors include: The house size, structure of rooms and parlors, location, price, traffic, facilities around, etc. Although salesmen know these influence factors, they did not know the detailed relations between these factors and the sale performance, which can be acquired by means of the above knowledge discovery method based on the genetic algorithm. In order to simple the process of knowledge discovery, the factors chosen finally include *the income, age and use of the customer, and the traffic convenience and the distances from school and shopping center of the building*.

Step 2: Preparing the data according to the factors determined above.

In order to facilitate the knowledge discovery, data need tidying up according to the factors determined above. Here the data table tidied up is as follows:

Table 5.5 Transaction record tidied up

Field name	Data type	Note	Example
Income	Number	The income of customer	4000
Age	Number	The age of customer	30
Use	Char[20]	To live or to invest	Investing
DistanceFromBusStop	Number	The distance from the nearest bus stop	800m
DistanceFromSchool	Number	The distance from the nearest school	1000m
DistanceFromShoppingCenter	Number	The distance from the nearest shopping center	500m

Step 3: Designing the coding rule of data.

Since the genetic algorithm only deal with the data represented in code, the data above need coding according to some kind of coding rule. Here the coding rule is as follows:

Table 5.6 Coding rule of transaction record

Field name	Data type	Note	Coding rule
Income	Number	The income of customer	1(low):lower than 3000 2(average):from3000 to 6000 3(high):higher than 6000
Age	Number	The age of customer	1(young): younger than 35 2(middle): from 35 to 50 3(old): older than 50
Use	Char[20]	To live or to invest	0: to live 1: to invest
DistanceFrom-BusStop	Number	The distance from the nearest bus stop	1: (close) than 500m 2:(middle) from 500m to 1500m 3: (far) than 1500m
DistanceFrom-School	Number	The distance from the nearest school	1: (close) than 500m 2:(middle) from 500m to 1500m 3: (far) than 1500m
DistanceFrom-ShopingCenter	Number	The distance from the nearest shopping center	1: (close) than 500m 2:(middle) from 500m to 1500m 3: (far) than 1500m

After being coded every record in transaction becomes a code string whose length is 6 digitals. Parts of records coded in the database are showed in the table 5.7.

Table 5.7 Parts of customer information records coded

No	Income	Age	Use	DistanceFrom-BusStop	Distance-FromSchool	DistanceFrom-ShoppingCenter
1	2100 (1:low)	28 (1:young)	Invest (1)	300 (1:close)	200 (1:close)	160 (1:close)
2	5000 (2:average)	36 (2:middle)	Live (0)	480 (1:close)	560 (2:middle)	100 (1:close)
3	24000 (1:low)	73 (3:old)	Live (0)	580 (2:middle)	880 (2:middle)	150 (1:close)
4	9000 (3:high)	56 (3:old)	Invest (1)	790 (2:middle)	180 (1:close)	580 (2:middle)
5	7800 (3:high)	30 (1:young)	Invest (1)	1000 (2:middle)	250 (1:close)	790 (2:middle)
6	5600 (2:average)	48 (2:middle)	Live (0)	1600 (3:far)	600 (2:middle)	220 (1:close)
…	…	…	…	…	…	…

Step 4: Defining the fitness function of customer.

The fitness function is used to select the survival entries for reproduction from the pervious generation entries. In the experiment, the fitness function is defined as follows:

$$V=count1/count0$$

Herein, count1 is the number of customers purchasing his units from our enterprise, and count0 is the number of customers visiting our sale center.

Step 5: Programming the genetic algorithm for knowledge discovery.

The genetic algorithm for knowledge discovery can be programmed by some kind of advanced programming language. Here we program it by C#.Net. The program can be embodied in the knowledge manage system which will be introduced in the next chapter.

Step 6: Setting the parameters of the genetic algorithm for knowledge discovery and run the genetic algorithm program.

In the experiment, the runtime parameters chosen are: the evolutionary community's size is 1000, the duplication probability is 0.01, the overlap probability is 0.01, the variation probability is 0.01, the stop condition is that the number of generation arrives 50000 or that the evaluation value of the best candidate will not changes.

5.5.3 Result of Real Estate Project Sale Knowledge Discovery

By running the knowledge discovery program, we obtained a few items of valuable sale knowledge as follows:

[Knowledge 1] ***the customers of the income being average , the age being middle , the use being to live and the distance from shopping center being near, most possibly purchase in the end.***

[Knowledge 2] ***the customers of income being high and age being young and use being to invest and the distance from school being near, most possibly purchase in the end.***

The two items of knowledge is very valuable for helping salesmen identify who are most possible purchasers among all visitors. They should provides better service for those most possible purchasers.

5.6 Conclusion

This chapter explores the knowledge discovery algorithm based on a genetic algorithm to mine the extensive information in a real estate enterprise database to obtain valuable knowledge.

Based on combining the genetic algorithm with the CRISP-DM data mining standard model, this chapter proposes one type of database knowledge discovery method. It provides the framework of one such database knowledge discovery method based on a genetic algorithm and provides the flow diagram of that genetic algorithm.

This chapter then demonstrates how to apply that database knowledge discovery method through an experimental real estate project sale knowledge discovery on the sales database of an actual real state enterprise, Hunan Tongrenzhiye. It describes in detail the database, the process and the result. The purpose of this example is only to demonstrate how to apply the knowledge discovery process. In a real life application, it would be much more complex than the case presented here.

Chapter 6

DESIGN OF A REAL ESTATE KNOWLEDGE MANAGEMENT PROTOTYPE SYSTEM

6.1 Introduction

As mentioned in the chapter 1, the finial objective of the dissertation is to build the base of the development of the real estate knowledge management system. This chapter is to explore how to apply the research achievement in the former chapters to the development of the real estate knowledge management system. It will give the design principle and the frame of the real estate knowledge management prototype system based on the knowledge acquisition methods above, i.e. ontology based agent cooperation search model and genetic algorithm based knowledge discovery method, and the design plans of its major modules. The prototype system's objective is to verify the feasibility of the design plan of the real estate knowledge management system to develop in the future.

6.2 Design Principle of Real Estate Knowledge Management Prototype System

In order to guarantee the advanced quality, the expansibility and the utilization of the real estate knowledge management system, during its design the following principle will be taken into account:

1. Knowledge acquisition function is an important part of the system's functions. It means that the ontology based agent cooperation search model and the genetic algorithm based knowledge discovery method will be embodied in the system.

2. To separate the users of the system and the developer of the system's ontology.

3. The system's functions are designed hierarchically, and provide different views aiming at different users.

4. The system adopts as mature technology as possible. The ontology modeling tool Protégé developed by the Stanford University is used in the ontology modeling. The ontology description language OWL (Ontology Web Language) of the international standard is used in the ontology representation.

5. The system must accords with the mainstream standard. The entire knowledge management system uses the J2EE design architecture. It is of three layers application architecture of the B/S pattern. The user may gain access to the system through the Internet.

6.3 The Architecture of Real Estate knowledge Management Prototype System

The system's architecture is shown in Figure 6.1. It consists of two layers, i.e. the application layer and the development layer.

The system application layer has provided a visualization interface for the ordinary users and the enterprise policy-makers to visit the knowledge management function through the browser. The major techniques needed to develop the GUI layer include: HTML, JavaScript, JSP and so on. HTML and JavaScript are mainly seated in the client, and executed by the browser; JSP is seated in the management platform, executed by the Tomcat application server. The GUI layer includes the user's inquiry interface and the model-view transfer. The user's inquiry interface provides the friendly inquiry function to the user, but the model - view transfer is responsible for transforming the user's input into the query sentences which can be understood by the system, and then presenting the results in a friendly way to the user by the view. The knowledge discovery module carries out the data mining from the ontology base and the database to gain the valuable information and knowledge which the users are interested in according to the knowledge discovery algorithm; The distributive agent cooperation search module is responsible for decomposing the inquiry sentences and unite with other real estate knowledge management systems based on ontology to gain the useful information and knowledge. These modules complete the semantic analysis and inference functions. The semantic analysis module uses semantic correlation and similarity algorithm based on ontology. Because the information processed in the application server layer is mainly the OWL information with XML format, Jena API developed by HP Corporation is used to process the OWL model. This system provides the OWL ontology parser for the application program to query and modify the ontology resource with OWL. The data storage layer includes mainly the following two parts: the domain ontology base and the resource database.

The system development layer is responsible for maintaining the ontology base. The ontology construction is mainly in the charge of the knowledge engineers with the real estate specialty background. Ontology developers can obtain the real estate terminology set through processing structural documents, then model the ontology through the ontology modeling tool Protégé to obtain the ontology files without conflict. Finally, the system administrator publishes the ontology into the system ontology base to realize the maintenance and the extensibility of the real estate knowledge management system.

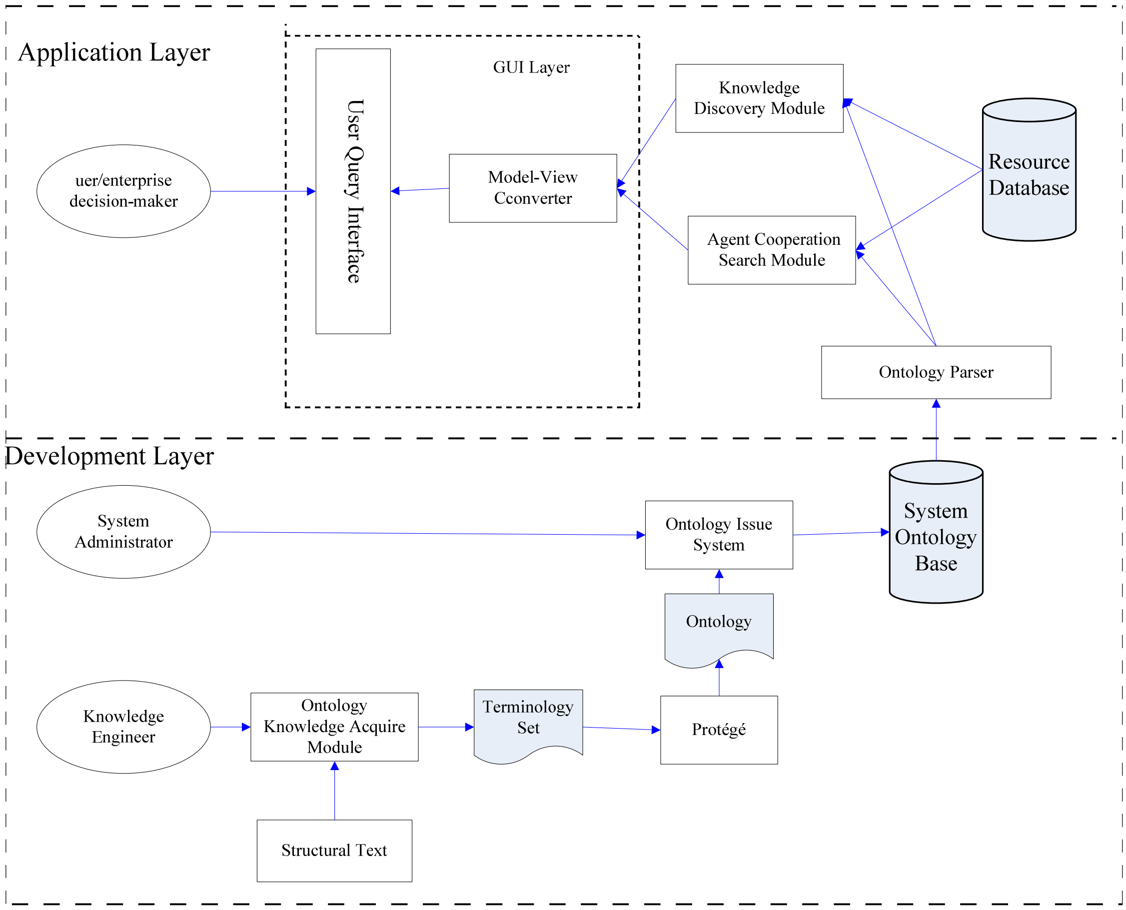


Figure 6.1 Architecture of Real Estate Knowledge Management Prototype System

6.4 Design of Major Modules of Real Estate Knowledge Management Prototype System

6.4.1 Ontology Knowledge Gain Module and Ontology Modeling Tool

The goal of the ontology knowledge capture module is to count the real estate terminology and the terminology attribute though analyzing the real estate structural documents with the above ontology capture algorithm.

The ontology capture module mainly uses the VC++6.0 development tool to complete the following several parts:

(1) Importing the structural documents;

(2) Analyzing its content and extracting the core glossary according to the text structure; and,

(3) Establishing the terminology context matrix of the core glossary and extracting the terminology similarities.

The related terminology similarity can be gained through the ontology knowledge capture module. So, we may analyze the real estate terminology and synonyms to provide the knowledge engineer with concepts to construct the ontology.

The knowledge engineers construct the ontology with the protégé tool developed by Stanford University. Protégé, which provides the users with the additional ontology visually, moreover supports the plug-in and open source codes. After knowledge engineers establish the ontology with this tool, the developers of the application program may use API functions of protégé to develop the ontology parser. The interface of protégé tool shows in the following figure.

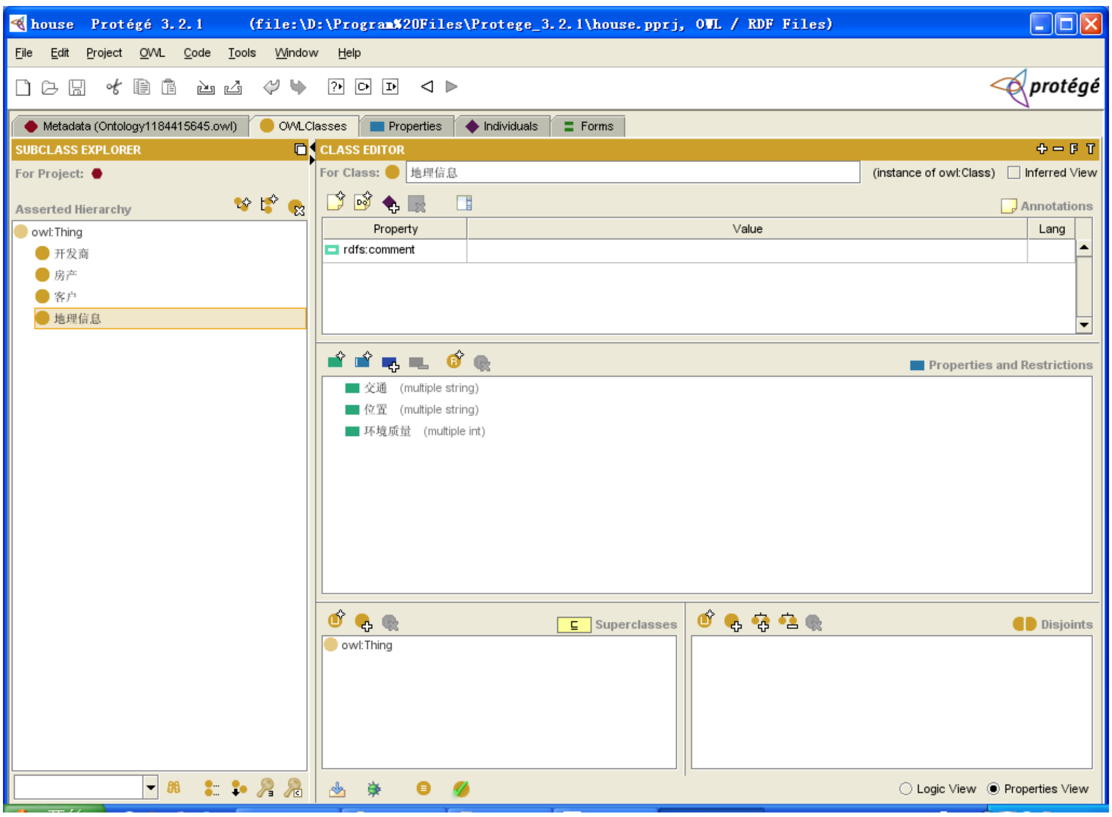


Figure 6.2 protégé Development Interface Chart

While analyzing the ontology development tool protégé, its main functions is as the following:

(1) Metadata plug-in: the resource manager, which provides knowledge engineers with other ontology resources to be added, deleted and cited;

(2) OWL Classes plug-in: the class editor, which provides the classification system of domain concepts. All the concepts are the subclass of OWL: Thing. It has the logic view and the attribute view. The logic view describes the necessary and sufficient conditions of the class

(3) Properties plug-in: the property editor, which has the numerical attribute and the objective attribute. The numerical attribute has the base data type attributes, such as char, integer, float and so on; The objective attribute provides the functional relations, such as: symmetry, transitivity and so on.

(4) Individuals plug-in: the individual editor, which manages individuals in the domain. It is based on the classification system of the class plug-in. We can add an individual to a given class. Once an individual belongs to a class, it has all the attributes of the class.

6.4.2 Ontology Parser

The ontology parser mainly uses Jena API functions developed by HP laboratory, and API methods of Protégé tool to parse the ontology, and provide the ontology data reading, the semantic reasoning and the information search for other application modules.

Jena is a kind of the Java framework used to construct the semantic World Wide Web applications, and provides interface methods to operate related RDF, RDFS and the inference engine programming environment based on rules. Jena is an open source project, and is developed by HP semantic network laboratory at present. The Jena structural framework related to knowledge ontology process is shown in Figure 6.3.

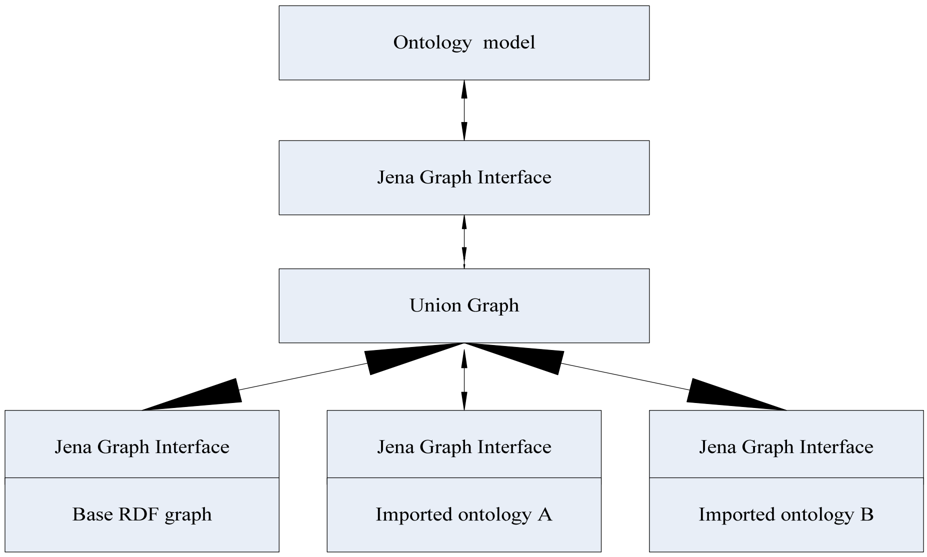


Figure 6.3 Jena Knowledge Ontology API Framework

The Jena framework mainly provides the following Java packages, interfaces and methods:

(a) RDF application programming interface;

(b) Functions providing RDF files to read and write various syntaxes, including RDF/XML, N3 and so on;

(c) The application programming interface to operate OWL files;

(d) Two modes based on the memory and permanent storage; and,

(e) An RDF instance data query language—RDQL.

6.4.3 Model-View Converter

The model-view transfer is designed by using MVC (Model/View/Controller) pattern to realize loose system coupling helpful to the system extendibility. The MVC pattern is a popular design patterns, and appears first in Smalltalk. MVC includes three kinds of objects, speaking simply: Model is the application object, View is its representation on the screen, Controller defines the way the user interface responses to the user input. “The model” in the MVC design pattern refers to the true codes really completing the task. The function is more important than the interface to the most of the web application programs. In the situation of the model’s separation with interface, the code can realize the maintainability and the reusability of the applications, so the model is usually called “service logic”.

The so-called “view” is actually the used interface. In the MVC pattern, the interface task is not usually big. Certainly, the view should have certain functionality, but it can not process the data.

“The controller” controls the interactive process between the model and the view, decides what kind of view returns to the user, and checks the input information through the interface and selects the model used to process the input information. The following figure is the relation between each component in the JSP MVC pattern.

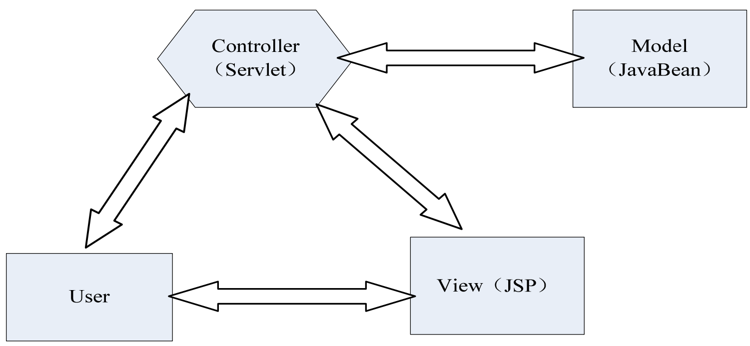
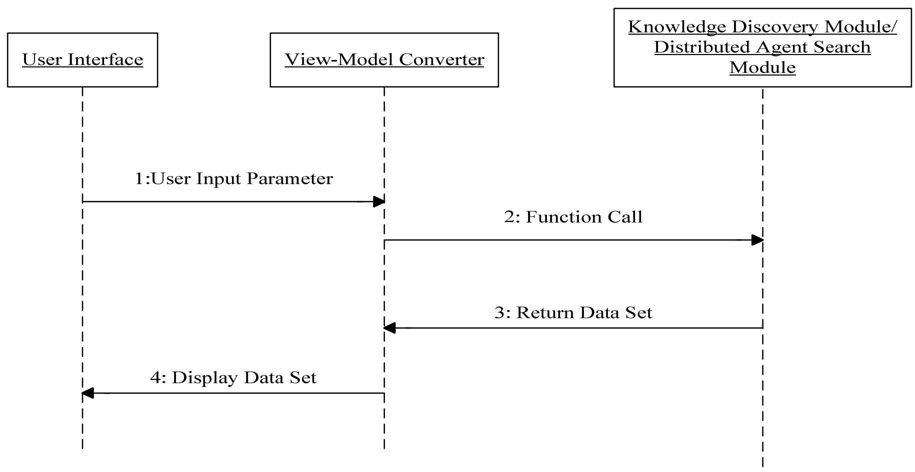


Figure 6.4 Relation Between Pattern modules in JSP

The controller of the model view in this system connects the user interface, the knowledge discovery component and the distributional search module based on Agent. It shields the system interior realization to the user. At the same time, according to the user interface, input value and request service type, it produce the corresponding function request sentences of the knowledge discovery module or the distributional search module based on Agent. Then, the knowledge discovery module or the distributional search module based on Agent returns the results to the model-view converter according to the request sentences. Finally the model-view converter calls the suitable page to present them to the users. The next chart is the interactive process between the view-model converter and the other functional components.



Fiugre 6.5 Model-View Converter Interactive Sequence

6.4.4 Agent Cooperation Search Module

Referring to the BDI rational model, the control mechanism of agent is a concurrent control mechanism with multithreads. Once running, Agent is endowed with a process space by the system, and starts running independently and continuously. All its behaviors are managed by this mechanism. It does not end its life and return its process space to the system until the users send the termination instruction. There are three resident threads and some temporary threads in the Agent's process space. The resident threads are: the main control thread, the message receiving thread and the message sending thread. The temporary threads are the task threads running in current time. They are created when Agent starts executing tasks. They disappear in the process space when the Agent finishes its task.

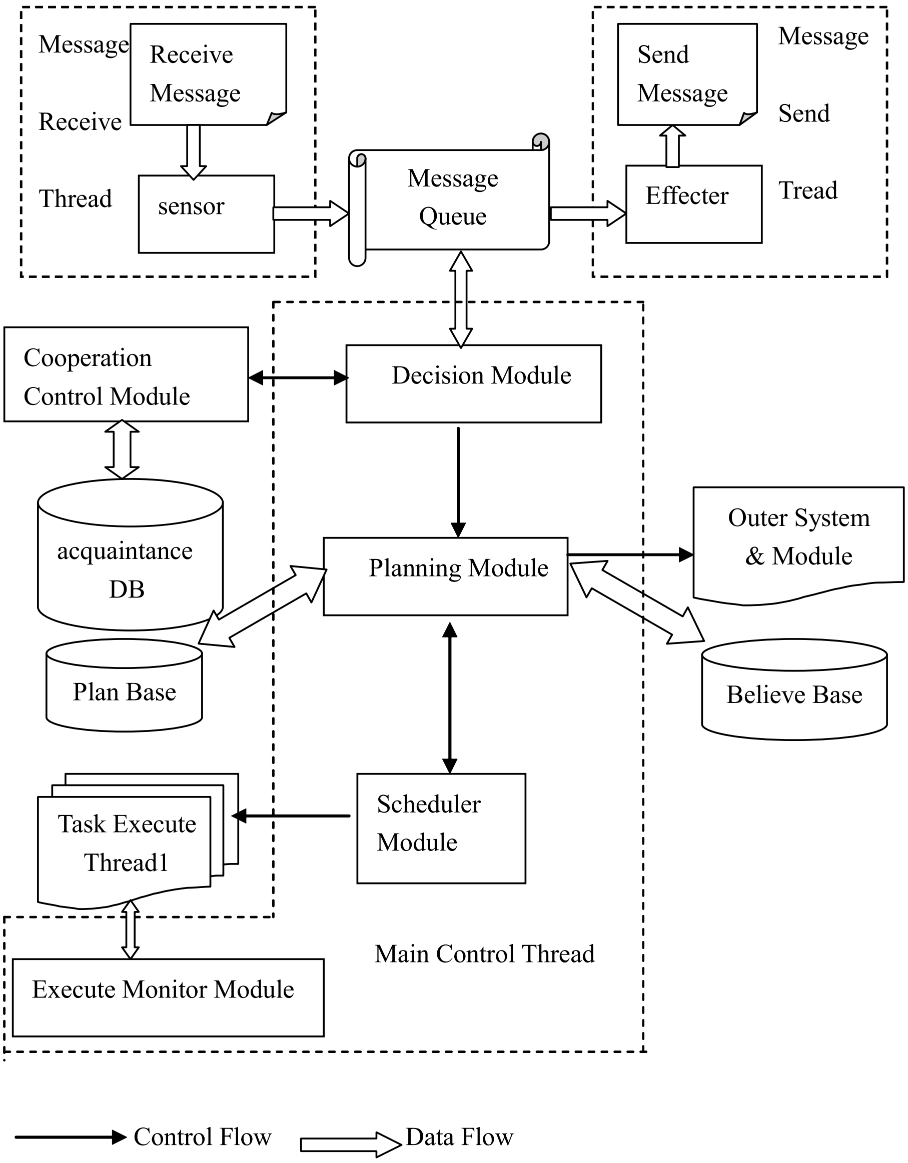


Figure 6.6 Agent Structure And Multithread Concurrent Control Mechanism

Figure 6.6 describes the Agent's interior structure and the concurrent control mechanism of multithreads. Its main modules include:

- Sensor: it is responsible for sensing the outer environment and receiving the messages from other Agents and user;
- Actor: it is responsible for informing the outer environment, sending messages to others, and providing the outer data to users;
- Decision module: it is responsible for deciding how to finish the tasks, i.e. which should be refused, which need to be completed in priority, which need to be executed in sequence, which need to be executed concurrently and so on;
- Plan module: it is responsible for establishing the plan schema about how to execute the task with the means-goal inference method;
- Plan base: it provides the logic description of the task operators which Agent can recognize. Each plan or sub plan is represented by the finite state automation;
- Believe base: it stores the information about Agent's and the environmental state. This information can be used in the logic inference;
- Scheduler module: it is responsible for schedule tasks, i.e. deciding such thing as when to execute the task, what kinds of tasks to be executed, when to suspend the tasks, what tasks to be suspended, when to stop the task, what kinds of tasks to be stopped and so on;
- Cooperation control module: it is responsible for controlling the time when it interacts with other Agents, and deciding the kinds of the interactive protocol.

It controls the session procedure with the cooperators according to the referred interactive protocol;

- Acquaintance DB: it describes the organization relation with other Agents and its functional belief about others. The cooperation control module can store and fetch it, when processing the cooperative relation with others;

- Execution monitor module: this is responsible for maintaining and monitoring the tasks executed by the interior Agent. Conceive a general control procedure: the sensor is monitoring the messages in the network continually. When receiving a message from other Agent, the sensor transfers the message into the message queue. The decision module fetches a message from the message queue in turn, and judges whether it is a new task. If it is, and meets Agent's desire, that is to say that it accords with the Agent's goal. The Agent will produce a task to realize the goal corresponding to this message, and puts it into the task queue. If it is an uncompleted task, it will produce responding message. Otherwise, if it is the cooperation message about the old task, it will transfer the message to the corresponding task thread, and the task queue takes the queue mark. After the decision module transfers tasks to the plan module, the plan module designs the planning schema which can realize this goal with the means-goal inference method. It will fetch from and store into the plan and belief library in the planning procedure, and call the outer system and components. If it is the complex goal which it can't realize alone, the cooperation control module will visit the acquaintance DB, and choose the suitable acquaintances, to which the cooperation control module send the cooperation request through the actor. After designing the planning schema, the scheduler module creates the new thread which is used to realize the goal. Many task threads can be executed in parallel. The execution and monitor threads monitor them in time when running the threads.

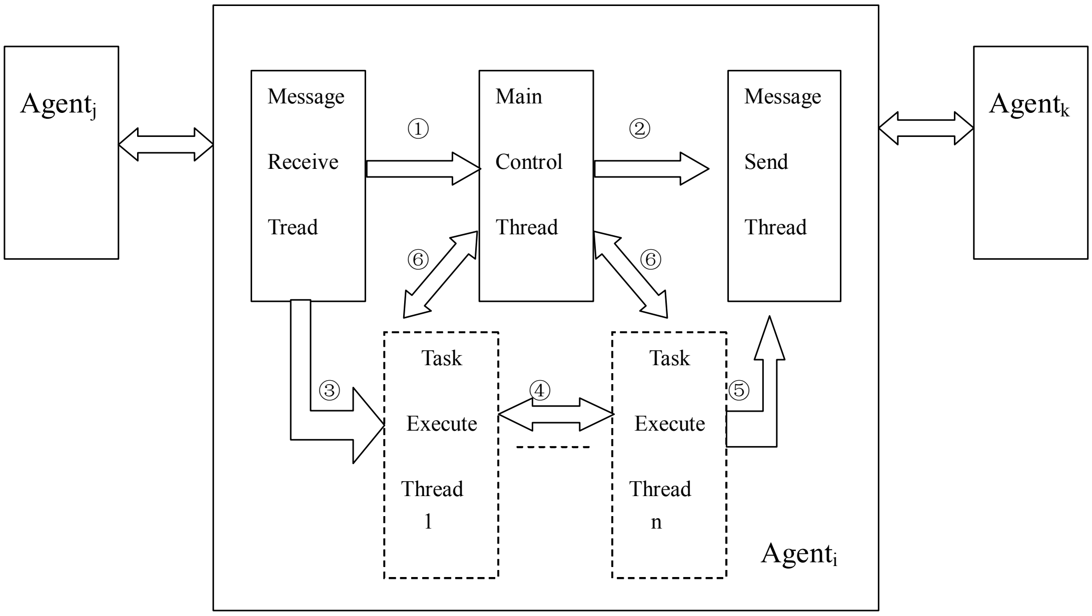


Figure 6.7 Multithread Dataflow of Agent Structure

The data exchange at the interior of the Agent can be realized through sharing the memory and sending and receiving messages between threads. The data flow between threads is shown in figure 6.7: i) the message receiving thread receives the cooperation request from other Agents, and transfers it to the main thread. And then, the main thread plans and creates the new task execution thread; ii) the main control thread transfers the Agent's cooperation request to the message send thread. And then, the message sending thread sends it to other Agents; iii) the message receiving thread receives the reply messages about some task from other Agents, and transfers it to the task execution thread; iv) t refers to the data exchange between the task execution threads; v) the task execution thread takes the results of execution as the reply to other Agent, and submits it to the message sending thread to send to the corresponding Agent; vi) the main control thread monitors the task execution threads, and transfers the instructive information to the corresponding task execution threads. The task execution thread should provide the execution information to the main thread of its own accord during executing the task. There are the bidirectional data flows between the task execution thread and the main control thread.

6.4.5 Knowledge Discovery Module

The knowledge discovery module unearths the general regular knowledge in lots of real estate database with the data mining algorithm to provide the decision-making knowledge to the enterprise policy-makers.

Aiming at the real estate characteristics, the knowledge discovery module provides the following functions mainly:

1. Aiming at the characteristic of individual real estate project, to help analyze which kind of people being most possibly the true purchasers of the units in the real estate project.
2. According to the house sale situation, to analyze what house style is the one the market needs most, and may count the cases of each kind of house transaction through the data mining, which has important reference value to the later real estate strategy;
3. Count the monthly dealings, obtain which month during the year is when the real estate sells well, and instruct the real estate developers to evaluate the repayment time of the fund investment, which has the vital significance to reduce the fund risk;
4. Count the real estate dealings in the region to know the region where the real estate is most popular with customers, and instruct the real estate to compete for the exploitation right with a definite goal, which also has the important reference value

to the competitive land tender of the real estate developers.

In addition, the knowledge discovery module has the vital significance to improve the real estate operation profit. It may count and mine according to the real estate operation pattern and finance to get the most suitable pattern and financial system for the real estate development by comparison. Because this aspect is the goal of the enterprise long-term operation, the knowledge discovery module has not developed the knowledge discovery of this aspect in this prototype system.

6.5 Development and Validity of Real Estate Knowledge Management Prototype System

6.5.1 Development of Real Estate Knowledge Management Prototype System

This prototype system is developed with C# and Java language, and its development environment is Microsoft Visual Studio 2005, and the ontology edit tool is Protégé3.2, and the file server is Apache Tomcat 5.0.

The running environment configure of the prototype system is as the following:

- The hardware configure is: the seven personal computers (Pentium4 CPU，512M memory)
- The network configure is: several Ethernet network lines with 100M
- The software configure is: the operation system is Windows xp, JAVA virtual machine edition is JDK5.0, Apache Tomcat 5.0.

6.5.2 Validity of Real Estate Knowledge Management Prototype System

The development of the real estate knowledge management system, which can be applied in many real estate enterprises, is an engineering of numerous programming works. As a prototype system, the real estate knowledge management prototype system in this dissertation is far from being truly applicable. Although the prototype system can not be applied in the real estate enterprises yet, its validity can be confirmed through the experiments in the chapter 3, the chapter 4 and the chapter 5.

6.6 Comparison with the Protégé Tool Developed by the Stanford University

Comparing with the protégé tool developed by the Stanford University, the real estate knowledge management prototype system is different in the following aspects:

6.6.1 Purpose

Protégé was developed to provide a tool for modeling and managing the ontology, including domain ontology and process ontology. The real estate knowledge management prototype system aims to provide a system for the real estate companies to acquire knowledge from the inside and the outside of the companies. Part of the knowledge in the system is represented in the form and the standard of the ontology. Beside of the ontology, there are other methods of representing knowledge in the system, such as the formula, rule.

6.6.2 User

Protégé mainly aims to model ontology, so its user is the developers of various ontologies. Among the users of the real estate knowledge management system, there are not only the developers of knowledge whose tasks are to acquire the knowledge from inside and outside, but also the viewer of knowledge whose tasks are to apply the knowledge to solve the business problems.

6.6.3 Running Pattern

Protégé is a single user program. The real estate knowledge management prototype system is a multi-user system, also may join the cooperation union to share the knowledge among the union members. Protégé is a windows program, but the real estate knowledge management prototype system runs under the B/S pattern.

6.6.4 Functions

Protégé is to model the ontology, so the functions provided by Protégé include how to build and manage the ontology. The Ontology is organized by classes described by its attributes, such as Name、Document、Constraints、Role、Template Slot. During the construction of the domain ontology, every subject is described through the class, which is named by the name of subject; what is input in the Documentation column is the description of the subject; the constraint of the class is described by the constraint instance, which is composed by the Boolean operator. There are two kinds of the role in the protégé, that is, Abstract and Concrete. Which to choose is determine by the ontology concepts. When the ontology is built, it may directly utilize the ontology resource in the Internet.

The real estate enterprise prototype system uses the protégé tool to build the ontology model; and the ontology representation uses the ontology description language OWL (Ontology Web Language) of the international standard. Because the protégé is inlaid in the real estate knowledge management prototype system, the latter also utilize the ontology resource in the Internet. Compared with protégé, it has the following different characteristic and functions: It separates the ontology capture and system inquiry in the knowledge management system management system OBKMS. The ontology capture uses the single version procedure, but the entire knowledge management system uses the J2EE design architecture facing three or multilayer application architecture of the B/S pattern. The user may access system far away through the Internet. This system uses the hierarchical design in the function, and provides the different views according to the different users. It has fully considered some new technological developments, technical standards and system interfaces and specially development standards representing the mainstreams, such as Agent technology, semantic Web service and so on. In order to enhance the acquiring capability of knowledge, the system provides the united inquiry interface. Besides, the knowledge hided in the data base can be mined through the KDD based on the genetic algorithm.

6.7 Conclusion

This chapter explores the ontology-based real estate knowledge management prototype system developed earlier to verify the new architecture and technology. It demonstrates the feasibility and advanced functions of the system technology.

This chapter also delves into the design principles, architecture and major modules' design of the system. In doing so it introduces the design of the ontology knowledge capture module and modeling tool, the ontology parser, the model - view transfer, the search module based on distributional Agent cooperation, and the knowledge discovery module.

C h a p t e r 7

CONCLUSION

7.1 Introduction

This dissertation aims to explore the knowledge acquiring methods suitable for the real estate industry. By the achievement of chapters above, this dissertation has done some contributions in this aspect, but it is certain that some limitations still exist in these achievements. In the last chapter, the contributions and their limitations will be summarized. In addition, the direction of further research will be thought.

7.2 Contributions at Theory Level

At theory level, this dissertation proposed a type of method of sharing and gaining real estate knowledge based on ontology. This method aims to solve the problem of complicated real estate knowledge acquiring through communicating and sharing between real estate enterprises and agent cooperating searching over the whole internet, which is not mentioned in the existing literatures. According to this method, the key is to create real estate ontology which can be understood by computers and different real estate enterprises without any divergence. The Chapter 3 discusses and gives in detail the procedures of how to create real estate domain and process ontology. The first step of building domain ontology is to obtain synonyms of terminology from a large quantity of structural text documents in the real estate industry. This text is then analyzed to produce the terminology, construct the terminology context matrix, and compute the terminology similarity. These terminology synonyms can be used to support knowledge engineers in modeling the ontology and in ontology mapping.

7.3 Contributions at Model Level

At model level, the main works of this dissertation include:

1. Designed an ontology based agent cooperation searching model

Aiming at increasingly difficult network information searches and inquiries driven by the explosive growth of information available on the Internet, this dissertation proposes an ontology based agent cooperation searching model. It also analyzes and designs on detail the structure and mechanism of main modules in the model.

2. Developed a knowledge discovery algorithm aimed at the real estate industry, combining a data mining algorithm and a genetic algorithm.

By combining a data mining algorithm and a genetic algorithm, this dissertation develops a knowledge discovery algorithm aimed at the real estate industry. This algorithm can mine and discover relevant knowledge from a large number of real estate databases. This process can unearth important knowledge for real estate developers to assist their decision making and strategy development.

3. Explored the design of a real estate knowledge management prototype system based on external and internal knowledge acquisition methods.

The prototype system described in this dissertation is an integrated knowledge management system which synthesizes the above theoretical developments. It can manage real estate knowledge effectively and can be used by the knowledge engineer, the ordinary user or the enterprise policy-maker. It is expandable since it employs modular development technology. The use of agent technology enables it to collaborate with other knowledge management systems.

7.4 Limitations

Although this dissertation has done some research work in how to acquire the real estate knowledge from the external and internal of real estate enterprises, owing to the limited time, the achievement is not enough perfect in the following aspects:

1. The real estate ontology given in the chapter 3 is only an example to demonstrate how to create the real estate ontology, which is far from the true application.

2. The real estate knowledge management prototype described in this paper is a basic system. It is intended to demonstrate how to apply ontology-based agent cooperation search techniques and genetic algorithm-based knowledge discovery to such a knowledge management system. It can only solve simple problems and is not yet sufficiently mature for use in industry.

7.5 Future Directions of Research

In order to develop the true applicable real estate knowledge management system, the following work would have to be done:

1. The task of the real estate ontology modeling is a complicated engineering. The ontology technique can solve semantic conflicts between different real estate terminologies by defining concept frames computers can understand, but

the full development of unified real estate ontology standard requires the support of government authorizes and industry associations. Additionally, ontology modeling is a huge undertaking that would need considerable financial support as well as the efforts of numerous knowledge engineers skilled both in industry operations and ontology development. Such an effort would have to have the widespread support of industry; it could not be done by a single research institute or real estate developer.

2. How to establish the trust mechanism between different real estate knowledge management systems? In order to achieve the true knowledge sharing, the different real estate knowledge management systems must still have the desire of knowledge sharing besides the resource sharing and united inquiry functions. However, in the real world, owing to commercial secrets inherent in a competitive market, almost all real estate knowledge management systems operate alone and have no resource sharing. The united inquiry system can't be realized until the trust and cooperation mechanisms between different knowledge management systems in the industry have been established.

3. A great deal of work must be done in order to complete the development of a reliable real estate knowledge management system with effective inter-operability. Such a software development project is complex and faces many technical challenges. This dissertation explored some of the critical technologies required for developing such a system. Its complete realization can only be achieved through the combined efforts of IT engineers and real estate specialists.

4. Knowledge acquisition technology encompasses a broad range of research fields. Besides the methods mentioned in this dissertation, there are many nascent and evolving technologies that are worth exploring that would contribute to the realization of an effective real estate knowledge system.

References

[1] Antoniou Grigoris, Frank van Harmelen (2004). *A Semantic Web Primer*, The MIT Press, Cambridge, Massachusetts, April 2004.

[2]Ajith Abraham , Rajkumar Buyya and Baikunth Nath (2000) Nature's heuristics for scheduling jobs on computational grids ,*The 8th Int. Conf on Advanced Computing and Communications (ADCOM 2000)*,Cochin, India.

[3]A. Tate (2003) Roots of SPAR - Shared Planning and Activity Representation. *The Knowledge Engineering Review*, 13(1),121-128.

[4]Austin Tate, Jeff Dalton, Jeffrey M. Bradshaw and Andrzej Uszok (2004) *Coalition Search and Rescue - Task Support Intelligent Task Achieving Agents on the Semantic Web*. Artificial Intelligence Applications Institute-The University of Edinburgh.

[5]Austin Tate, J. Levine, J. Dalton and Alex Nixon (2002), Task Achieving Agents on the World Wide Web. *Creating the Semantic Web*, D. Fensel, J. Hendler, H. Liebermann and W. Wahlster (eds.), MIT Press,2001.

[6]A.W. Brown (2000) *Large-Scale Component-Based Development*, Prentice Hall, New Jersey.

[7]Baader Franz and Calvanese Diego (2003) *The Description Logic Handbook: Theory, Implementation, and Applications*, UK: Cambridge University Press.

[8]Baisong Liu (2005) Research About Ontology Based Knowledge Management Key Technology, *Informatics Transaction*, 24(1), 75-81.

[9]Bao Ai-hua, Yao Li, Liu Fang, Zhang Wei-ming (2008) Research on the Organization-based Method for Modeling Multi-agent System, *Journal of Chinese Computer Systems*, 29(1), 66-72

[10]Bemhard Bauer,Jorg P. Muller,James Odell (2003) Agent UML: a formalism for specifying multi-agent interaction. *Agent-Oriented Software Engineering,*Paolo Ciancarini and Michael Wooldridge eds., Springer-Verlag, Berlin, 2001, 91-103,

[11]B. Motik, A. Maedche and R. Volz (2002) A Conceptual Modeling Approach for Building semantic-driven enterprise application. *Proc of the First International Conference on Ontologies, Databases and Application of Semantics*. Springer, California,USA, 35-39.

[12] B. Omelayenko (2002), *RDFT: A mapping meta- ontology for business integration*, Lyon France, 2002.

[13] Brachman, R. J. and Schmolze, J. G. (1985) An overview of the K L-ONE knowledge representation system. *Cognitive Science*, 9(2),171~216.

[14]Chen Huahua, Du Xin and Gu Weikang (2005) Genetic algorithm based global path planning in a static environment , *Journal of Zhejiang University (Science Edition)* , 32(1), 49-53.

[15]Chen Li-na (2007) On the knowledge management of the real estate develop enterprise, *Shanxi Architecture*, 33(34),249-251.

[16]Chen Ming-liang (2006) Research on Frame of CRM Basic Theory System, *Journal of Industrial Engineering and Engineering Management*, 20(4), 36-41.

[17]Christian Flui, Marta Sabou and Frank van Harmelen (2002) , *Ontology based information visualization:Toward Semantic Web Applications, Visualizing the Semantic Web XML-Based Internet and Information Visyalization (Chapter 3)*,Springer-Verlag, 2002,45-58.

[18]Christopher Welty, (2003), Ontology Research. *AI Magazine*, 24(3), 11-12.

[19] C. N. Hsu and C. A. Knoblock (1996) Using Inductive Learning To Generate Rules For Semantic Query Optimization. *In Advances In Knowledge Discovery And Data Mining*, U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, eds. AAAI

[20] C. Menzel and M. G. (2001), A formal foundation for process modeling. *Second International Conference on Formal Ontologies in Information Systems,* Welty and Smith (eds), 256-269.

[21]Craig C. Lofton (1999) Knowledge mgmt: A real estate revolution. *National Real Estate Investor*. 41(9),34.

[22] C. Schlenoff, M. Gruninger and M. Ciocoiu, (1999), The Essence of the Process Specification Language. *Transactions of the Society for Computer Simulation*, 16(4), 204-216.

[23] Daniel M Lewis (2002), *Visualization of meta object facility models with human usable graphical notation,* Distributed Systems Technology Centre, University of Queensland, 2002.

[24]David E. Goldberg (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wessley Publishing Company, Inc.

[25] Deng Zhihong, Tang Shiwei, Zhang Ming, Yang Dongqing, Chen Jie (2002), Overview of Ontology , *Acta Scientiarum Naturalium Universitatis Pekinensis* , 38(5), 730-728.

[26] D. Garlan and D. Le Métayer eds (1997), Coordination Languages and Models. *Proceedings of the 2nd International Conference (COORDINATION'97).* LNCS, Springer-Verlag, Berlin (D), 1282.

[27]Edwin S. H. Hou, Nirwan Ansari and Hong Ren(1994) Genetic Algorithm for Multiprocessor Scheduling, *IEEE Trans on Parallel and Distributed Systems* , 5(2), 113-120.

[28]Fang Liu, Clement Yu and Weiyi Meng (2004) Personalized web search for improving retrieval effectiveness. *IEEE Transactions On Knowledge and Data Engineering*, 16(1),28-40.

[29]Fan Honhda,Ye Wen and Ma Xiangling (2003) Low Altitude Penetration Route Planning of the Plane Based on Genetic Algorithm , *Aeronautical Computer Technique*, 33(1),13-17.

[30]Feng Chen and Yunsong Zou, (2004), Agent Based Information Research Strategy And Planning, *Computer Engineering And Application*, 40(35),128-131.

[31]Ge Jing (2002) Knowledge Transfer and Organizational Learning in the Process of Internationalization of Enterprises, *China Soft Science*, 2002 (1),7-60.

[32]Ge Xinquan, Zhou Xiuling(2007), Research about Virtual Enterprise Knowledge Management Model, *Industrial Technology&Economy*, 26(12),99-102.

[33]Grigoris Antoniou ,Frank van Harmelen(2003), *Web ontology language: OWL*, Springer-Verlag, 2003.

[34]Gruber, T.R.(1993), *Toward Principles for the Design of Ontologies Used for Knowledge Sharing, Revision. Available as Technical Report KSL 93-04*, Knowledge Systems Laboratory, Stanford University, August 23. 1993.

[35]Guarino Nicola (1997) Understanding, building and using ontologies: A commentary to "using explicit ontologies in KBS development", *International Journal of Human-Computer Studies*, 46(2/3),293-310.

[36]Guo Shu-Hang ,Gao Jing, Lan Yu-Qing and Jin Mao-Zhong (2005) Software Component Modeling based on Ontology for Dependability , *Journal of Nanjing University(Natural Sciences)*, 41(z1) ,90-95

[37]Haiqun Ma (2000) Discussion About Knowledge Economy, Knowledge Management And Knowledge Rights, *Books Information Knowledge*, 2000(2), 2-6.

[38]Hao Liu and Cuohu Qin (2001) The Enterprise Knowledge Management System Building, *Guangdong Social Science*, 2001(6),19-22.

[39]He Ke-Qing, He Fei, Li Bing, He Yang-Fan, Liu Jin, Liang Peng and Wang Chong (2005) Research on Service -Oriented Ontology & Meta-Modeling Theory and Methodology, *Chinese Journal of Computers*, 28(4), 524-533.

[40]Hironori Kasahara, Seinosuke Narita (1984) Practical Multiprocessor Scheduling Algorithms for Efficient Parallel Processing, *IEEE Transactions on Computers*, 33(11),1023-1029.

[41] H. J. Levesque , R. Reiter, Y. Lesperance , F. Lin, and R. Scherl (1997), GOLOG: A logic programming language for dynamic domains. *Journal of Logic Programming Special Issue on Reasoning about Action and Change,* 31(1-3),59-84.
Press/The MIT Press, Menlo Park, CA., 1996, pp. 425-445.

[42]Hyunseok Hwang , Taesoo Jung , Euiho Suh(2004), A LTV model and customer segmentationbased on customer value:a case study on the wireless telecommunication industry, *Expert Systems With Applications,* 26(2),181-188.

[43]I. Horrocks, U. Sattler and S. Tobies (2000) Practical reasoning for very expressive description logics. *Logic Journal of the IGPL*, 8(3), 239-263.

[44]I. Horrocks, U. Sattler, S. Tobies (1999), Practical reasoning for expressive description logics. *In Proceeding of the 6th International Conference on Logic for Programming and Automated Reasoning* (LPAR'99), Springer-Verlag, 161-180

[45]I. Horrocks, U. Sattler and S. Tobies (2000), Reasoning with individuals for the description logic SHIQ. *In Proceeding of the 17th International Conference on Automated Deduction* (CADE 2000), Springer-Verlag, 428-496

[46]I. Horrocks and U. Sattler (2001), Ontology reasoning in the SHOQ (D) description logic. *In Proceeding of the 17th International Joint Conference on Artificial Intelligence*(IJCAI 2001) Morgan Kaufmann, 199-204

[47]Imtiaz Ahmad and Muhammad K Dhodhi (1996) Multiprocessor Scheduling in a Genetic Paradigm, *Parallel Computing*, 22(3),395-406.

[48]James Pitkow, Hjnrich Schqtze,Todd Cass,Rob Cooley,Don Turnbull, Andy Edmonds,Eytan Adar and Thomas Bureuel (2002) *Personalized search. Communications of the ACM*, 45(9),50-55.

[49]Jiewen Luo, Zhongzhi Shi, Maoguang Wang and He Huang (2005) Multi-Agent Cooperation: A Description Logic View. *Proceeding of PRIMA* 2005.

[50]John H Holland (1975) *Adaptation in Natural and Artificial Systems,* The U. of Michigan Press, 1975.

[51]John Poole, Dan Chang, Douglas Tolbert and David Mellor (2003) *Common Warehouse Metamodel Developer's Guide*, New York:wiley, 2003.

[52]Jos Warmer and Anneke Kleppe (2003), *The Object Constraint Language*, New York: Addison-Wesley, 2003.

[53] J. R. Quinlan (1993). *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann, 1993.

[54]Junjie Chen, Yun Xue, Hantao Song, Yuchan Lu and Xueli Yu (2003) The Research and Design of A Metasearch Engine Based on Agent, *Computer Engineering And Applications*, 39(10),33-36.

[55]Kang Fei and Wang Yao-nan(2006), Robot Path Planning Based on Hybrid Artificial Potential Field Genetic Algorithm , *Journal of System Simulation*, 18(3),774-777.

[56]Katia Sycara, Massimo Paolucci, Martin Van Velsen and Joseph Giampapa (2003) The RETSINA MAS Infrastructure. *Autonomous Agents and Multi-Agent Systems*, 7(1-2), 29-48.

[57]Ke Li and Qinghua Bai (2001) Research About Knowledge Based on XML, *Computer And Modernization*, (3), 13 -18.

[58]Ke Xu and Zhi-ming Cui (2006) User Profile Model Based on User Search Histories, *Computer technology and Development*, 16(5),18-20.

[59]Kuokka D. and Harada, L.(1995), On using KQML for match making. *Proceedings of the First International Conference on MultiAgent Systems(* ICMAS95). Lesser V ed. Cambridge , MA: MIT Press, 239-245.

[60]Lei Chen (2004) *Research About Open System Heterogeneous Agent Cooperation Model[master's degree paper]*, National University of Defence Technology, ChangSha, 2004.

[61]Liang Xiaohui, Wu Wei, Zhao Pinping (2002), Research on global path planning method based on genetic algorithm and large scale real terrain data, *Journal of Computer Research and Development*, 39(3), 301-306.

[62]Liao Lejian ,Yao Jinhguo, Cao Yuanda and Li Shouli (2004) Semantic Web Architecture and Its Implementation, *Transactions of Beijing Institute of Technology* , 24(2),145-149.

[63]Li Jianzhong, Deng Wanjun and Wen Zhongsheng (2006) Research on cultivation of corporate culture based on knowledge management in real estate enterprises. *Proceedings of 2006 International Conference on Construction & Real Estate Management*, 1172-1175.

[64]Lin Jian-Ning ,Wu Hui-zhong (2004) Scheduling in Grid Computing Environment Based on Genetic Algorithm, *Journal of Computer Research and Development,* 41(12),2195-2199.

[65] Lin Li and Men Guibin (2006) Study on the knowledge management system and its evaluation model in real estate developing enterprise, *Proceedings of 2006 International Conference on Construction & Real Estate Management*, 1161-1166.

[66]Liu Haidi, Yang Yi , Ma Shengfeng and Li Lian (2008) A Grid Task Scheduling Strategy Based on Hierarchical Genetic Algorithm, *Journal of Computer Research and Development* , 45(z1),35-39.

[67]Liu Jing-jing and Zhai Zheng-jun (2006) Tasks Matching and Scheduling of Distributed System Using a Modified Genetic Algorithm, *Microelectronics&Computer* , 23(6),216-219.

[68]Lixin Han, Lifei Huang, Shuan Hun, Li Xie and DaoXu Chen (2002) Design and Implementation of an Agent Based Information Retrieval Systems for Internet Applications, *Journal of the China Society for Scientific and Technical Information*, 21(3),258-263.

[69]Luiz Marcio Cysneiros, Julio Cesar Sampaio do Prado Leite (2004), Nonfunctional Requirements: From Elicitation to Conceptual Models, *IEEE Transactions on Software Engineering*, 30(5),328-350.

[70]Martin Fowler (2004) *UML distilled: A Brief Guide to the Standard Object Modeling Language*, New York: Addison-Wesley, 2004.

[71]Ma Zhiliang , Fu Siyuan and Li Heng (2004) *An integrated system model for market analysis of real estate development,* Hong Kong Polytechnic University, 2004.

[72]Ma Zhiliang, Deng Ziyu, Li Heng, Fu Siyuan(2005), Study on Modeling of GIS Based Integrated System for Feasibility Analysis of Real Estate Development, *Computer Engineering and Applications,* 41(1),197-199,206.

[73]Ma Zhiliang, Fu Siyuan, Li Heng and Li Yonghe (2007), Information management system for market analysis of real estate development projects, *Journal of Tsinghua University(Science and Technology)* , 47(9),1409-1413.

[74] M.C. Lee (1991) IPDA - an expert system for debugging Prolog programs. *International Journal of Artificial Intelligence in Education,* 2(4),91-101.

[75]Meng Xian-quan, Zhao Ying-nan, Xue Qing(2008), Application of Genetic Algorithm in Path Planning , *Computer Engineering*, 34(16),215-217,220.

[76]Michael Gruninger and Mark S. Fox (1995) Methodology for the Design and Evaluation of Ontologies. *Workshop on Basic Ontological Issues in Knowledge Sharing*, IJCAI-95, Montreal, 1995.

[77]Michael Stollberg, Uwe Keller, Peter Zugmann, Reinhold Herzog (2003), *Semantic Web Fred – Agent Cooperation on the Semantic Web*.

[78] Mike Uschold and Michael Gruninger (1996) Ontologies Principles,Methods and Applications. *Knowledge Engineering Review*, 11(2).

[79]Mingkai Dong and Zhongchi Shi (2004) Dynamic Description Logic Based Agent Model, *Computer Research And Development*, 141(15), 780~786.

[80]M. M. Kwan, P. Balasubramanian. (2003) Process-oriented knowledge management: a case study. *Journal of the Operation Research society*. 54(2), 204-211

[81] M. N. Razali , Z. Manaf (2005) Development of knowledge management system model for real estate business in Malaysia: A preliminary study. *Proceedings of 2005 International Conference on Construction & Real Estate Management*, Batu Pahat, Malaysia ,847-851.

[82] M. R. Genesereth and N. J. N. (1987) *Logical Foundations of Artificial Intelligence*. San Mateo : Morgan Kaufmann Publishers, 1987.

[83] M.R. Quillian (1967), Word concepts: A theory and simulation of some basic capabilities. *Behavioral Science*. 1967(12), 410~430.

[84]M. Stollberg, S. Thomas (2005). Integrating agents, ontologies, and semantic web services for collaboration on the semantic web. *In: Proceedings 2005 AAAI fall symposium series on agents and the semantic web*. Arlington, Virginia, USA.

[85] M. Stollberg, H. Lausen, S. Arroyo, P. Smolle, R. Herzog, D. Fensel (2004) Fred Whitepaper: An Agent Platform for the Semantic Web, *Technical Report DERI TR-2004-01-09, 2004*.

[86] Natalya F. Noy and Deborah L. McGuinness (2001) Ontology Development 101:A Guide to Creating Your First Ontology. *Standford Knowledge Systems Laboratory Technical Report KSL-01-05,* March, 2001.

[87] Nicola Guarino(1998), *Formal ontology in information systems*, Trento, Italy, 1998.

[88] Nonaka, I. and Takeuchi, H. (1995), *The Knowledge-creating Company: How Japanese Companies Create the Dynamics of Innovation*, Oxford University Press, Oxford.

[89] Patrick S.W. Fong and Paul Dettwiler (2009) Entrepreneurial firms and their knowledge creation: a study of real estate management. *Facilities.* 27 (7/8) ,267-276.

[90] Peter Meso, Robert Smith (2000), A resource–based view of organizational knowledge management system. *Journal of Knowledge Management*, 4(3), 224-234.

[91] Ping Jiang, Zhiming Cui (2004) Analysis and Research on the User Interest Profile in Intelligent Search Engine, *MicroElectronics&Computer* , 21(11),24-26.

[92]R. Buyya, D. Abramson and J. Giddy (2000) An economy driven resource management architecture for global computational power grids. *Int. Conf. on Parallel and Distributed Processing Techniques and Applications* . Las Vegas, Nevada,517-525.

[93] Renkai Zhang (2005) The Strategies of China's Real Estate Enterprises in Shaping Core Competitiveness, *Marketing Research* ,2005 (11),50-52.

[94] R. J. Brachman and T. Anand (1996) The Process Of Knowledge Discovery In Databases: A Human-Centered Approach. *In Advances In Knowledge Discovery And Data Mining* , eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA.,37-57.

[95] R. H. Sprague (1980), A framework for the development of decision support systems. *MIS Quarterly*, 4(4),1-26.

[96] Robert Neches, Richard Fikes, Tim Finin, Thomas Gruber,Ramesh Patil, Ted Senator, and William R. Swartout (1991) Enabling Technology for Knowledge Sharing. *AI Magazine*, 12(3), 36-56.

[97] Ronald J. Brachman and Hector J. Levesque (1984) The tractability of subsumption in Frame-Based Description Languages. *In Proceeding of AAAI-84*. University of Texas, Austin, Texas: AAAI Press, 34-37.

[98] Roy T.Black, Joseph S. Rabianski (2003) Defining the Real estate body of knowledge: a survey approach, *Journal of Real Estate Practice and Education*. 6(1),33-54.

[99] R. Ruggles (1997), Knowledge tools: using technology to manage knowledge better. *Working paper, Ernst & Young*, 1997.

[100] Salton G., M. M. J. (1983) *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[101] S. A. DeLoach (2001), Analysis and design using MaSE and agentTool, *In 12th Midwest Artificial Intelligence and Cognitive Science Conference (MAICS 2001)*, Miami University, Oxford, Ohio, March 31–April 1 2001.

[102] Scott A. Deloach, Mark F. Wood and Clinth. Sparkman(2001) Multiagent systems engineering, *International Journal of Software Engineering and Knowledge Engineering*, 11(3), 231-258.

[103] Sean Bechhofer, Frank van Hammelen and Jim Hendler,etc. (2004) OWL Web Ontology Language Reference. W3C Recommendation, *URL http://www.w3.org/TR/2004/REC-owl-ref- 20040210/.*

[104] S. E. Robertson and K. Sparck Jones (1976) Relevance Weighting of Search Terms. *Journal of American Society for Information Science*, 27(3), 129-146.

[105] S. Geneva (2004) FIPA Agent Management Specification, *FIPA TC Agent Management,* 2004/18/03, http://www.fipa.org/specs/fipa00023/SC00023K.pdf.

[106]S．Gauch and A. P. (1999) Ontology based personalized search. *Proceedings of the 11th International Conference On Tools With Articial Intelligence(ICTM*), 391-398.

[107] Song Huazhu ,Zhong Luo and Wang Hui (2005) *Constructing an Ontology for Web-based Learning Resource Repository* , Canada, Carleton University , 2005.

[108]Terry R., P. R. S., Katia Sycara (2002) Calendar Agents on the Semantic Web, *IEEE Intelligent Systems*,84-86.

[109] Thomas H. Davenport and Laurence Prusak (1998) *Working knowledge: How organizations manage what they know*. Boston, MA :Harvard Business School Press, 1998.

[110] Thomas R. Gruber (1993) A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2), 199-220.

[111] Tiedong Chen, Ziyu Liu and Lei Huang (2007) Research and Application of Enterprise Knowledge Management System Based on Ontology, in *IFIP International Federation for Information Processing*, Volume 254, *Research and Practical Issues of Enterprise Information Systems* II Volume 1, eds. L. Xu, Tjoa A., Chaudhry S. (Boston:Springer), 747-751.

[112] T. Marvin, G. Angela (2004) *The Use of Ontologies in Web-based Learning*, American, 2004.

[113] T. R. Payne, R. Singh and K. Sycara (2002) RCAL:A Case Study on Semantic Web Agents. *Proceeding of the 1st International Conference Autonomous Agents and Multiagent Systems (AAMAS 02)*. New York: ACM Press, 2002.

[114] U. M. Fayyad and P. Smyth (1996). *From Data Mining To Knowledge Discovery: An Overview In Advances In Knowledge Discovery And Data Mining* , eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA., 1996, 1-34.

[115]Vincenzo Di Martino (2002) Scheduling in a grid computing environment using genetic algorithms. *Marco Mililotti the 16th Int'l Parallel and Distributed Processing Symp* (IPDPS2002), 2002.

[116]Vincenzo Di Martino and M Mililotti (2004) Sub-optimal scheduling in a grid using genetic algorithms, *Parallel Computing* **,** 30(5-6),553-565.

[117]Wang Hong-jian Fu, Ming-yu Shi, Xiao-cheng Bian and Xin-qian (2005) Research on Global Path Planning Method For Autonomous Underwater Vehicle Based on Large-scale Chart Data , *Journal of System Simulation* , 17(3),567-570.

[118]Wang Liangbin and Zhu Guojin (2004) Ontology and the Reusability of Component, *Computer Engineering and Applications*, 40(14),53-56.

[119] W. Buntine (1996) A Guide to the Literature On Learning Probabilistic Networks From Data, *IEEE Transactions on Knowledge and Data Engineering*, 8(2) ,195-210.

[120]W. Buntine (1996) Graphical Models For Discovering Knowledge. *In Advances In Knowledge Discovery And Data Mining*, eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA., 1996, 59-82.

[121]Wei Liu and Junjie Chen (2005) A Framework for Intelligent Meta-search Engine Based on Agent, *Computer Engineering and Applications*, 41(3),137-138,211.

[122]Wei Luo (2003) RDF(Resource Description Framework)---Web Meta-Data Integration Scheme, *Informatics Transaction*, 22(2), 2-3.

[123]Wenjie Luo, Zhongchi Shi (2006) Description Logic Based Multi-Agent Cooperation Model, *Computer Research And Development*, 43(8),1317-1322.

[124] W. J. Frawley, G. Piatetsky-Shapiro and C. Matheus (1991) Knowledge Discovery In Databases: An Overview. *In Knowledge Discovery In Databases*, G. Piatetsky-Shapiro, and W. J. Frawley, eds., AAAI Press/MIT Press, Cambridge, MA., 1-30.

[125] Wu Jie, Shi Qinfen,Jiang Mia, Zhang Yunhua (2008) Knowledge Transfer Model and Realization Way in R&D Team of Real Estate Enterprises. *Proceedings of 2008 International Conference on Construction & Real Estate Management.* Toronto, CANADA ,403-406.

[126]Xiaofeng Zhu and Songzhen Lang (1999) Exploring Knowledge Management System Early, *Information Theory and Practice*, 22(4), 245-247.

[127]Xia Heping and Zhan Qiangfa (2005) Thinking about Raising the Central Competitivity of Real Estate Coporations, *Academic Search for Truth and Reality,* 2005 (3) ,46-50.

[128]Xie Dong , Shi Jin-tao , Li Xiao-yun (2007) The Constructive Model of Knowledge Based Core Competence in the Real Estate Companies, *Information Science*, 25(8),1252-1256.

[129]Xin Huang, Xiaojuan Xu and Guoliang Xu (2005) Ontology Based Enterprise Knowledge Management System Research, *Technology and Engineering*, 5(6), 351-356.

[130] Xu Dong, Zhang, Qingpu, Tian Yang (2006) Research on the knowledge management in the course of the real estate marketing, *Proceedings of 2006 International Conference on Construction & Real Estate Management*, 1141-1145.

[131]Xueying Duan and Yingchao Zhang (2007) Research on intelligent meta search engine ACISE based on agents collaboration, *Computer Engineering And Design*, 28(24),5955-5958.

[132]Xu Lu (2006)Some thoughts on constructing the knowledge management systems in real estate enterprises, *Proceedings of 2006 International Conference on Construction & Real Estate Management*, 789-793.

[133]Xu Xiaoping (2004) Design and implement of market analysis system based on Internet, *Computer Engineering and Design* , 25(4),530-532.

[134]Ye Yanbing and Ding Lieyun (2001) Design and Study of Real Estate Early Warning Index System, *Optimization of Capital Construction*, 22(3), 1-3.

[135]Y. Shen (2007) Design and construction of knowledge map in real estate enterprises. *Proceedings of 2007 International Conference on Construction & Real Estate Management,* 942-947.

[136]Zhang Zhijie and Chen Longqian (2004) The Analysis of Universal Process and the Emphases about Real Estate Early Warning, *Construction Economy* , 2004 (8), 92-96.

[137]Zhao Likun, Chen Liwen and Zhang Guozong (2007) Research on the Innovation Mode of Real Estate Enterprise Informatization, *Construction Economy* , 2007 (10), 81-83.

[138]Zhihong Xu, Xiangdan Hou and Jizhou Sun (2003) An algorithm-based task scheduling in grid computing. *CCECE 2003-Canadian Conf on Electrical and Computer Engineering* 2003.

[139]Zhong Qiu-xi Chen Huo-wang(2000), Genetic Operators in Task Matching and Scheduling, *Journal of national University of Defense technology* , 22(3), 34-38.

[140]Zhong Qiu-Xi and Xie Tao (2000) Survival Strategy of solution in GA, *Computer Engineering & Science*,22(1),14-17.