Knowledge Modelling for a Hotel Recommendation System

B. A. Gobin, and R. K. Subramanian

Abstract—Knowledge modelling, a main activity for the development of Knowledge Based Systems, have no set standards and are mostly done in an ad hoc way. There is a lack of support for the transition from abstract level to implementation. In this paper, a methodology for the development of the knowledge model, which is inspired by both Software and Knowledge Engineering, is proposed. Use of UML which is the de-facto standard for modelling in the software engineering arena is explored for knowledge modelling. The methodology proposed, is used to develop a knowledge model of a knowledge based system for recommending suitable hotels for tourists visiting Mauritius.

Keywords—Domain Modelling, Knowledge Based Systems, Knowledge Modelling, UML.

I. INTRODUCTION

K NOWLEDGE and skills are seen as strategic resources to the knowledge based society. It is important to define knowledge in this context and distinguish it from data and information. Knowledge Based Systems (KBS) were developed for this purpose. Knowledge Based Systems are needed in many fields of expertise such as diagnosis problems in the medical field, assessing loans in banks, maintenance systems in industries.

KBS development started with first generation expert systems with a single flat knowledge base and general reasoning engine, typically built in a rapid-prototyping approach [1]. It was essentially based on the process of knowledge transfer. Maintenance of such systems was very difficult. Hence the approach changed to a methodological approach which was similar to that of software engineering with knowledge as its main focus.

Knowledge Engineering is no longer simply a means of mining the knowledge from the expert. It now encompasses methods and techniques for knowledge acquisition, modelling, representation and use of knowledge [2].

Knowledge modelling is considered to be the main activity for the development of KBS. However there are no set standards for knowledge modelling. Abdullah et al. [2] argue that modelling of knowledge bases are done in an ad-hoc way and proposes to develop a UML profile to be used for modelling. Knublauch [3] argues that some existing

methodologies have a steep learning curve and require a considerable development overhead. Support is missing for the transition from abstract level to high-quality implementation.

The paper hence proposes a simple procedure for the construction of the knowledge model. UML [4] which is the de facto language for modelling in the software engineering arena is proposed and used for domain modelling. The development of the knowledge model for a Hotel Recommendation System shows how the different steps need to be implemented.

II. KNOWLEDGE ENGINEERING METHODOLOGIES AND KNOWLEDGE MODELLING

Several methodologies and frameworks have been developed over years e.g. CommonKADS[5], Protégé[6], MIKE[7], and MOKA[8]. They all lay emphasis on modelling. Models are used to capture the essential features of a real system by breaking them down into smaller components so that they can be better understood. They are used in system development to draw blueprints of the system and facilitate communication between different people. Models are important for understanding the working mechanisms within a knowledge-based system [4]. Modelling contributes to the understanding of the source of knowledge, the inputs and outputs, the flow of knowledge and the identification of other variables [2]. Thus knowledge modelling is considered to be the key component for the construction of knowledge based systems.

The modelling paradigms for these methodologies are ontologies and problem solving methods (PSM)[3]. Ontologies provide a way of representing domain knowledge. It consists of concepts, relationships among concepts and constraints on the relationships. PSM provide for reasoning components needed in KBS.

III. LINKING ONTOLOGY MODELLING WITH OO MODELLING

It has become a trend for system developers and researchers in KE to adopt object oriented modelling in developing conceptual models for knowledge systems. CommonKADS and MOKA have both adopted UML [5,8]. The use of UML and OCL to model ontologies is found in many research papers [9, 10].

The domain knowledge consists of the domain schema and the knowledge base. Ontologies and object-oriented languages have many things in common as shown Fig. 1 [3]:
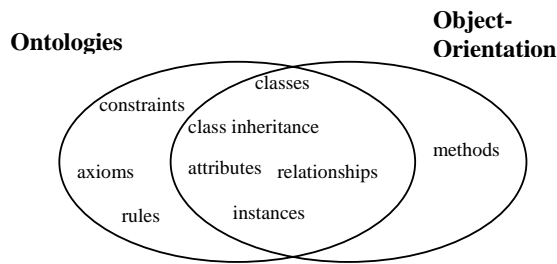
Fig. 1 Common features between ontologies and object-oriented language

Referring to these common features, UML constructs can be used to model the domain knowledge.

- Concepts are represented as class diagrams.
- Slots are represented as attributes
- Relationships are represented as relationship between the classes
- Constraints are represented OCL expressions attached to the class diagrams
- Instances from the knowledge base can be represented by the class diagram

IV. KNOWLEDGE MODEL CONSTRUCTION

The following steps have been identified for the construction of the knowledge model.

1. Problem-Context Opportunity

This phase consists of identifying the problem due to the knowledge intensive activity which requires the use of a knowledge based system. CommonKADS[5] proposes this phase while developing the organisational model.

2. Identifying Knowledge

Information sources that are useful for knowledge modelling are identified. The following activities help to achieve this:

o Interviews
Talking to the people who are working in the organisation will help to better understand the key features of the business.

o Building Scenarios
Scenarios can be built with the help of an expert. Textual analysis of these scenarios will help to have a better understanding of the domain knowledge.

o Process Breakdown
This is an important part as it helps to decide how to construct a knowledge model. The business process is broken down into specific task which will be carried out by the knowledge system. Activity diagrams can be used

to model the business processes. CommonKADS proposes this activity for developing the organisation model.

3. Modelling the Domain

Based on the interviews and textual analysis during the knowledge identification phase, domains can be represented using UML diagrams. Nouns identified can be used as classes and attributes. Relationships also are identified. OCL is used to illustrate constraints.

4. Modelling the Reasoning Component

The main inferences needed to cater for the reasoning component of the system are identified. Inferences are described in details stating clearly the input conditions and the output results. The phase will help in proper design and implementation.

5. Knowledge Model Validation

A prototype is developed and tested by experts and some users to validate the knowledge model.

V. BUILDING THE KNOWLEDGE MODEL OF THE HOTEL RESERVATION SYSTEM

A. Problem-Context Opportunity

With respect to our application this can be illustrated as follows:

TABLE I
IDENTIFYING KNOWLEDGE ORIENTED PROBLEMS AND OPPORTUNITIES

Problem and Opportunities	Tourists coming to Mauritius may spend a lot of time looking for an appropriate hotel
Context	Enable tourist to choose a hotel which will best suit the reason of their visit and their preferences
Solutions	Provide an intelligent online Mauritian Hotel Suggestion system which enable tourist to save time when looking for a hotel by giving them suggestions on which hotels to choose based on their visit purpose, ethnicity, budget, meal preference

B. Identifying Knowledge

After having interviewed key persons and built appropriate scenarios, the following were identified as the input criteria for hotel selection:

1. Purpose of visit
2. User Traits (health conscious, party animal)
3. User Preferences (shopping, types of cuisine, water sports)
4. Period
5. Budget

The system will use these input to infer the possible activities that the person might be interested to do in Mauritius and based on that propose the best hotels.

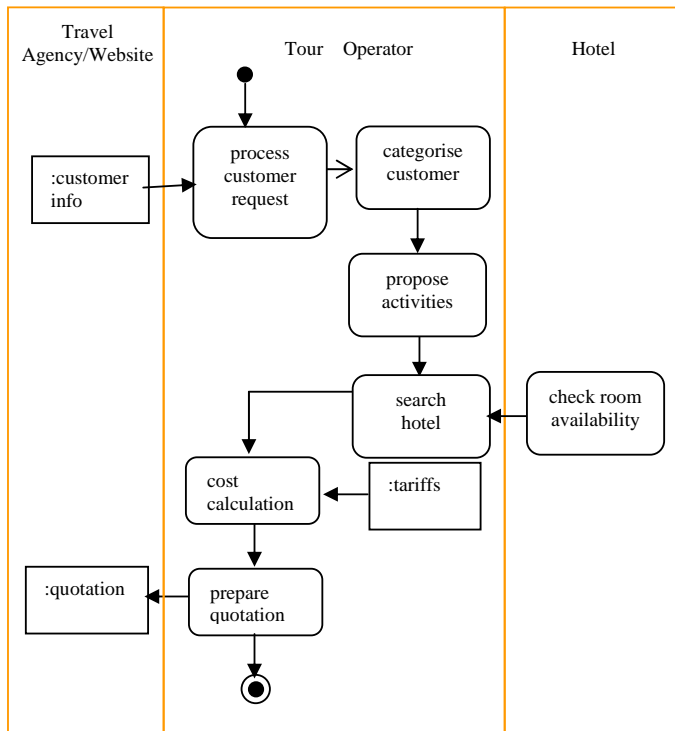The process breakdown is illustrated in Fig. 2.

Fig. 2 Process breakdown for hotel reservation

VI. Modelling of the Domain of the Hotel Reservation System using UML

The model thus has the following classes as shown in Fig. 3:

• *Traveller*

This class represents the person who is looking for a hotel. It will have a one-to-many relationship with the class Hotel as he can be suggested one or more hotels. It also has a one to one relationship with the class **TravellerCategory**.

The attributes of the class are as follows:

- Name
- Age
- Purpose
- Nationality
- Lengh_of_stay
- Budget
- Category
- Restaurant_type
- Activities
- Room_type
- Hotel_suggested

The category of the traveller is inferred based on the age, purpose, nationality and category is assigned to the attribute "Category" of the class.

• *TravellerCategory*

Class *TravellerCategory* represents the different types of travellers. It has a one-to-many relationship with the class

Traveller. This class is used to infer the category of the traveller. Its attributes are:

- Category_name
- Category_description

• *Activities*

Class *Activities* contains all the different types of activities that can be done by tourist during their stay in the island. It consists of two main classes: *HotelActivities* and *OutdoorActivities*. It has a many-to-many relationship to the class *Traveller*

• *Hotels*

Class Hotel represents the hotels. It is related to travellers. It contains classes *Rooms*, *Restaurants* and *HotelActivities*.
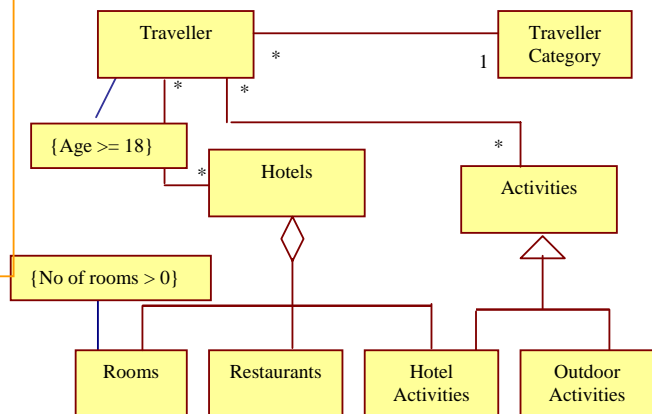
Fig. 3 Class diagram representing domain of the system

The expressions in curly brackets in Fig. 3 represent OCL constraints associated with the classes.

VII. Modelling of the Inference Component of the Hotel Reservation System

The main reasoning activity of the system is to recommend the most appropriate hotel. Each scenario identified during the knowledge identification stage is elaborated so as to explain the working of the system. Fig. 4 and Fig. 5 are examples of two scenarios that have been elaborated.

Scenario 1:

A trace example of the system for a business traveller would be as follows:

System >> What is the purpose of your visit?
Traveller >> Business

System >> What is your nationality?
Traveller >> Indian

System >> How old are you?
Traveller >> 25

System >> Are you married?
Traveller >> No

System >> Will you be travelling alone?
Traveller >> No

System >> Will you me having any business meetings?
Traveller >> Yes

System >> Are you health conscious?
Traveller >> Yes

System >> Would you like buy souvenirs?
Traveller >> Yes

System >> Would you prefer to be near the sea?
Traveller >> No

System >> For how long will you stay?
Traveller >> 4 days

System >> What is your budget?
Traveller >> $1000

Fig. 4 Scenario for business traveller

Fig. 5 is a trace example of the system for a couple who is looking for a hotel.

After all scenarios have been elaborated, the main inference scenarios are fine tuned. Fig. 6 shows the inferences that are needed for the main task which is Hotel Recommendation. Based on these input, the system should be able to infer the following:

- the category of traveller
- the kind of restaurant preferred
- the kind of activities preferred during stay
- the location of the hotel
- the most appropriate hotel

System >> What is the purpose of your visit?
Traveller >> Holidays

System >> What is your nationality?
Traveller >> French

System >> How old are you?
Traveller >>30

System >> Are you married?
Traveller >> Yes

System >> Do you have children?
Traveller >> No

System >> Will you be visiting Mauritius with your spouse?
Traveller >> Yes

System >> Will you be visiting Mauritius with your children?
Traveller >> No

System >> Are you health conscious?
Traveller >> Yes

System >> Would you prefer to be near the sea?
Traveller >> Yes

System >> Do you like to swim?
Traveller >> Yes

System >> Do you like to practise sport activities?
Traveller >> Yes

System >> Do you like to practice aquatic sports?
Traveller >> Yes

System >> Do you like sight seeing?
Traveller >> Yes

System >> Do you like to buy souvenirs?
Traveller >> Yes

System >> Do you like shopping?
Traveller >> Yes

System >> For how long will you stay?
Traveller >> 10 days

System >> What is your budget?
Traveller >> $1000

Fig. 5 Scenario for couples

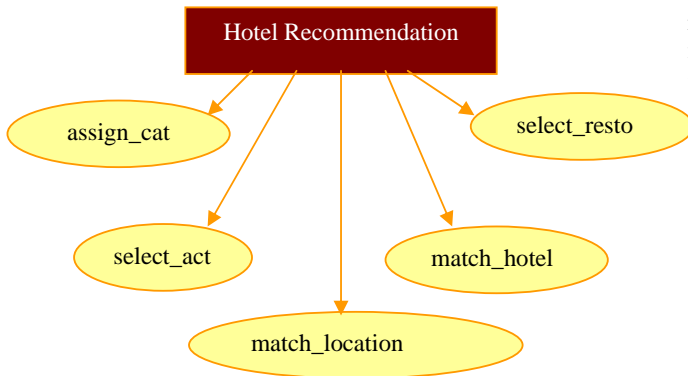Table II explains each inference in detail stating clearly its input and output conditions.

Fig. 6 Inferences to achieve the main task of Hotel Recommendation

TABLE II
INFERENCES NEEDED FOR HOTEL RECOMMENDATION

Inference	Description	Input	Output
assign _cat	infers traveller category	purpose of visit age nationality budget	traveller category
select _resto	infers the type of food that person would prefer and chooses restaurants among the list of hotels	nationality, traveller category	preferred restaurant ethnicity
select _act	infers the type of activities a traveller may be interested in	all answers with respect to activities are considered	preferred activities
match_ location	infers locations to be either coastal or centre	preference for the sea	hotel location
match_ hotel	infers the most suitable hotel	traveller category, traveller activities, preferred restaurant, hotel location	recommended hotel

VIII. VALIDATING THE KNOWLEDGE MODEL

Protégé 2000 has been chosen as development environment due to its user friendly and mature acquisition tool and its library of API that can be added. Instances created in Protégé can be easily imported in CLIPS, which can be used to build the prototype.

IX. CONCLUSION

To alleviate the problems faced by KBS developers due to the complex nature of some methodologies and also the lack of standards for the knowledge modelling, we have proposed steps to develop a knowledge model for a system

recommending Mauritian hotels. We have used UML for modelling the domain due to its common features between Object Oriented modelling and ontology modelling.

We are presently working on the prototype of the application that will be used to validate the knowledge model.

REFERENCES

[1] P. Speel, A. Th. Schreiber., W. van Joolingen., G.J. van Heijst, G. Beijer, *Conceptual Modelling for Knowledge Based Systems*, Encyclopedia of Computer Science and Technology, Marcel Dekker Inc. New York, 2001.
[2] M.S. Abdullah, A. Evans, I. Benest, R. Paige, C. ,Kimble *Modelling Knowledge Based Systems Using the eXecutable Modelling Framework(XMF),* Proceedings of the 2004 IEEE, Conference on Cybernetics and Intelligent Systems, Singapore, 2004.
[3] H. Knublauch , *An Agile Development Methodology for Knowledge-Based Systems Including a Java Framework for Knowledge Modeling and Appropriate Tool Support*, Dissertation (PhD thesis), University of Ulm, 2002.
[4] G.Booch, J.E. Rumbaugh and I.Jacobson, *The Unified Modelling Language User Guide,* Addison-Wesley, Reading, MA, 1999.
[5] A. Th. Schreiber, J. Akkermans., A. Anjewierden., R. De Hoog, N. Shadbolt, W. Van De Velde , B. Wielinga *Knowledge Engineering and Management: The CommonKADS Methodology,* MIT Press, 2000.
[6] J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubezy, H. Eriksson, N. F. Noy, and S. W. Tu. *The Evolution of Protege: An Environment for Knowledge-Based Systems Development*, International Journal of HumanComputer Studies, 2003.
[7] J. Angele and D. Fensel and D. Landes and R. Studer, *Developing knowledge based systems with MIKE*, Journal of Automated Software Engineering, 1998.
[8] MOKA homepage. Available: http://www.kbe.conventry.ac.uk (URL)
[9] H. Knublauch, T. Rose, *RoundTrip Engineering of Ontologies for Knowledge Based Systems,* Twelfth International Conference on Software Engineering and Knowledge Engineering (SEKE), (239-247), Chicago, IL, 2000.
[10] S. Cranefield and M. Purvis, *UML as an ontology modelling language*, In Proceedings of the Workshop on Intelligent Information Integration, 16th International Joint Conference on Artificial Intelligence (IJCAI-99), 1999.