Using Ontology Reasoning Over Location Based Services

Akansha A. Totewar
MTech(CSE), Department of Computer Science and Engineering, Government College of Engineering, Amravati 444 604, India.

Prashant N. Chatur
HOD (CSE), Department of Computer Science and Engineering, Government College of Engineering, Amravati 444 604, India.

AmitMeena
M.E (CSE), Persistent System Limited,Nagpur 440 022, India.

ABSTRACT
The motive of the current paper is to present process of enabling semantics framework as well as review the most commonly used Location based services. The paper determines how to implement intelligent system using ontology reasoning over location based services. The Location based services using ontology determines how semantics can be applied to make user context based reasoning framework. The objective is to develop framework (using ontology) which enables user based reasoning. Basically depending on the users context (date, time, longitude and latitude), the framework will infer overall useful information by considering contextual of user and inferring new knowledge on it. Even with knowledge of observed user locations, we can recognize mobility modes that are useful for several application domains.

Keywords
Location based services (LBS), Ontology Framework, Protégé OWL, Jess and Jena reasoning.

1. INTRODUCTION
The field of knowledge engineering is becoming an increasingly important area of computer science. Initiatives such as the Semantic Web [1] "… in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [2], will rely on ontologies to share data. Ontologies provide a shared conceptualization of a domain by defining the concepts in the domain and describing how those concepts are related to each other. However, most domains of discourse are not static, but evolve as the understanding of the domain grows. In order for ontologies to evolve successfully, there is a need for effective tool support. Representation standards for ontologies such as the W3C's Web Ontology Language (OWL) [3] and development tools like Protégé [4] are becoming prevalent, but the tools to support version control and difference comprehension are still lacking for ontology development. Of the ontology development tools currently available, the open source Protégé project developed by the medical informatics group at Stanford University is one of the most mature and best-adopted. The key feature that has contributed to Protégé's success is its open source plug-in architecture that allows it to be easily extended to better suit the needs of particular users.

This present paper establishes a contribution towards enabling Semantics over the Location Based Services using ontology reasoning. The ubiquitous computing develops an environment for human-computer interaction. The framework can be developed so that the user interaction can become easier.

Location based services only provide the attributes like latitude, longitude and altitude but cannot interpret the semantic of locations. Generally, there are two actions performed by the LBS system: one is positioning and the other is providing services based on the location of users. Through the analysis of user attributes, tag attributes and service attributes and the relationship among them, the system will use the rule of ontology reasoning to find the web services to meet user demand, enhancing services searching of precision and completion.

Through the analysis of user attributes, tag attributes and service attributes and the relationship among them, the system will use the rule of ontology reasoning to find the web services to meet user demand, enhancing services searching of precision and completion. The use of ontologies and explicit semantics enable performing logical reasoning to infer sufficient knowledge on the classification of processes that machines offer, and on how to execute and compose those processes to carry out manufacturing orchestration autonomously.

2. RELATED WORK
In ubiquitous computing environment, it is an important issue for context-awareness, which is aware of context and reasons appropriate service according to the context. Particularly, it is expected that home network adopt ubiquitous computing first of all and provides variable context aware services. Among context-aware services, location-aware service is one of the key context-aware computing [5].

The framework can be used for representing a user-centric view of usage contexts [6]. DL-reasoners can then be used for organizing context definitions, merging domain knowledge into these definitions, and performing recognition of contexts from sensor inputs.

OWLDL ontologies [7] represent non-trivial aspects of context, and to prefer forms of off-line ontological reasoning, while resorting to on-line ontological reasoning only when strictly required.

The motivation having local ontology reasoners is twofold. On one side, we believe that each entity, other than accessing shared ontologies, may hold private ones and may use values in ontological reasoning that should remain private. On the other side, this choice enables what we call on-demand ontological reasoning, which is described in [8]. In particular cases contextual data can be derived through ontological reasoning only populating the ontology with information provided by different entities. In this case, reasoning must be performed on-demand at the time of the service request.

A database supported approach, based on our "meta mapping" approach [9], successfully developed in a former project for huge ontologies on servers. It has the capability to be efficiently scaled down to mobile devices.

Ontologies are seen as the key technology [10] used to describe the semantics of information at various sites, overcoming the problem of implicit and hidden knowledge and thus enabling exchange of semantic contents. As such, they have found applications in key growth areas, such as ecommerce, bio-informatics, Grid computing, and the Semantic Web. An ontology can be described as [11] a specific vocabulary referring to an abstract model of basic concepts of a problem domain. Ontologies are composed of classes describing basic concepts in a domain and relations between them, properties related to features and attributes of the concepts, restrictions of properties and individuals, i.e. instances of the predefined classes.

The Web Ontology Language (OWL) is a W3C recommendation standard that can be used for expressing ontologies which can be processed by software. OWL DL is a sub language of OWL, based on Description Logics and supports those users who need maximum expressiveness while retaining computational completeness which makes it ideal for the ebBP ontology [12].The semantics of an information source according to [13] may be described using an ontology defined as "an explicit specification of conceptualization". The task of integration using different ontologies is a classical problem in information science, and continues to be highly active research issue within many topics, including databases, interoperability, the semantic Web, knowledge representation, data warehousing, and geographical information integration.

3. RESEARCH AND METHODOLOGIES

3.1 Ontologies and their development

Ontologies provide a formal specification of a domain of discourse and are becoming increasingly prevalent in the high tech world.

3.1.1 What is an ontology?

There are many different definitions of an ontology and also some question of where an ontology ends and a knowledge base begins [14]; however, for our purposes, Gruber's short definition is suitable. "An **ontology**is an explicit specification of a conceptualization" [15]. The use of ontologies to construct knowledge base systems is growing rapidly. As already mentioned, they are widely used in the medical community and will provide the backbone of the Semantic Web. On the surface ontologies may appear to be like database schemas; however, ontologies are not a way of organizing a specific data set for efficient retrieval, but rather a reusable structure for data within a domain that is designed to capture all the inherent relationships and meta-data among the knowledge that will be stored in there. Ontologies are intended for both humans and computers to manipulate. In short, ontologies provide a common vocabulary for communication of knowledge within domains [16].

There are two primary methods that have been used to construct ontologies. Description Logic based systems and Frame based systems. The following is a description of the top level items of a frame-based knowledge mode:

- **Classes** are collections of objects that have similar properties. Classes are arranged into a subclass-superclass hierarchy using either single or multiple inheritance. Each class has slots (described next) attached to it. Slots can be inherited by the subclasses.
- **Slots** are named binary relations between a class and either another class or a primitive object (such as a string or a number). Slots attached to a class may be further constrained by facets.
- **Facets** are named ternary relations between a class, a slot and either another class or a primitive object. Facets may impose additional constraints on a slot attached to a class, such as the cardinality or value type of a slot.
- **Instances** are individual members of classes.

The primary difference between description logics (DL) and frame based knowledge systems is that as a subset of first order predicate logic, DL includes the ability to automatically classify new concept descriptions with respect to previously defined concepts and to check the consistency of declared statements. This frees knowledge engineers from having to explicitly enter all the information about a new concept because the system will automatically add any implied information (based on previously defined concepts). Thus, a description logic based system has both explicitly and implicitly defined information as opposed to a frame based system where all information must be explicitly defined.

3.1.2 Ontology development process

As ontologies become more complex their development becomes increasingly collaborative, requiring a group of domain experts and engineers to construct them [17]. This parallels the historical development of software systems where, as systems grew in size and complexity, more people were required to complete the project and *ad hoc* development procedures were not suitable. Formal models were required to define the software development process and workflow management tools were required to help engineers adhere to the model. Nowadays, a suite of tools is often used to support software development projects. Two key tools within such a suite are some sort of version control software to track the evolution of the system and a difference tool to compare versions of files. Likewise, ontology development is beginning to enter the stage where projects require a formal development process [17] and the support of tools to help engineers to adhere to a defined process. The set of tools is similar to those used in software engineering with ontology development sharing the need for versioning and difference detection tools.

3.1.3 Ontology Design

Designing comprehensive ontologies requires supporting the ontology designers by design methodologies, ontology editors, and automated reasoning tools to reflect the decision-making policies of the designers.

To ensure the quality of the designed ontologies especially the correctness of domain ontologies is particularly important. There are generally two strategies to guarantee the validity of ontologies: developing more sophisticated ontology modeling tools and verification by logical reasoning. Reasoning support of ontology has been an indispensable component in ontology management systems. With the advent of expressive ontology languages (such as OWL) and its close relationships with description logic, many important information (such as is-a relation of classes) and logical contradictions can be detected by description logic reasoners.

3.2 Protégé and OWL

Ontology editor Protégé is a free, open source ontology editor and knowledge-based framework which is developed by Stanford University and Manchester University (version 4.0 and above). Protégé is based on Java, is extensible, and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development. It is a desirable tool for editing, browsing ontologies, and some reasoning operations such as incoherence detection can also be performed in it. Protégé 4.0 and the editions above have embedded Pellet and FaCT++ reasoners into ontology editor, which makes reasoning more convenient. Protégé is supported by a strong community of developers and academic, government and corporate users, who are using Protégé for knowledge solutions in areas as diverse as biomedicine, intelligence gathering, and corporate modeling.

3.3 Jess and Jena Reasoning Agent

Please Agents are aware of context of itself, reason context using that information, act of itself and communicate with other agents. And agents react to user action or context, and they have reasoning and learning ability as well as pro-activity for knowledge. Moreover, they perform autonomously action that user needs and have social ability for cooperation among multiple agents.

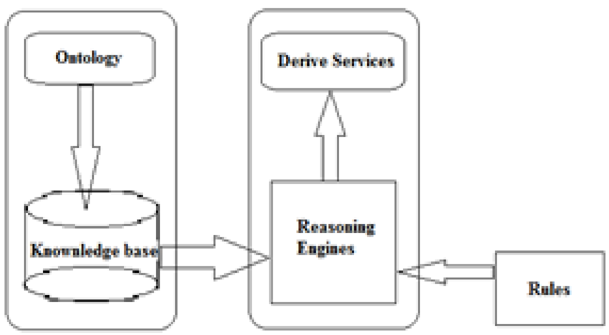

Fig 1: Architecture for agent

3.3.1. Jess

Jess, Java Expert System Shell, is a rule engine and scripting environment allowing building so-called intelligent software known as Expert Systems. Jess rules are similar in concept to "*if...then*" constructs. The *if*part of rules contains a series of patterns that are tried to be matched against existing facts. If it matches the "*then*" part of rules is executed. The rules are repeatedly applied to a set of facts. Jess uses Rete algorithm to perform the process. The rules are executed whenever their *if*parts are matched, which makes Jess less deterministic than typical procedural language.

3.3.2. JessTab

JessTab is a plug-in to Protégé. It acts as a bridge between Protégé ontology and Jess. It interacts with Protégé and Jess allowing, for instance, mapping Protégé instances to Jess facts. It is also possible to fire rules using JessTab console to directly manipulate the ontology by creating new classes or objects.

3.3.3. Jena

Jena is a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFs and OWL, SPARQL and includes a rule-based inference engine. Jena is open sources and grown out of work with the HP Semantics Web Programme HP Laps. The Jena Framework includes:

- A RDF API
- Reading and writing RDF in RDF/XML, N3 and N Triples
- An OWL API
- In-memory and persistent storage
- SPARQL query engine.

4. SEMANTIC FRAMEWORK ARCHITECTURE

The semantics architecture flow diagram for location based services is shown in the following figure, it consist of end user who is in want of the service he would apply for the user account which will be stored in the database of the browser. Then according to the address of the user or after identifying the location of the place form where the user is operating, the service provide will filter the content of the knowledge base wherein all nearby visits according to the user need is stored. And provide the same as an output to the user giving more information like distance of those places form the current location also the certain specification for the current preference.

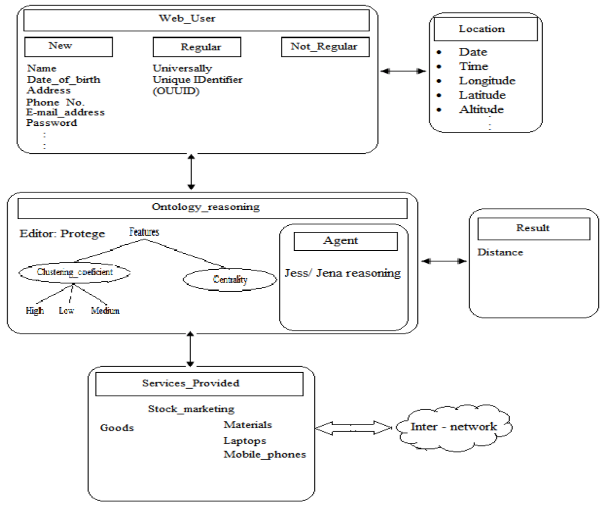

Fig 2: Semantic Framework Architecture.

5. IMPLEMENTATION AND MECHANISM

The implementation of the current paper can be done according to the resources available in the current situation. For defining or designing the ontology reasoner we can use protégé as an editor while we can also use the agent like Jess or Jena reasoners which provide the environment related to the frames, also pellet is a new reasoner which provide the helpful approach towards finding a complete OWL-DL reasoner with extensive support for reasoning with individuals, user-defined data types, and debugging support for ontologies.

Mechanism which can be followed during the implementation of the semantics framework is more complicated as compared to normal framing in Java. As according to the above mentioned semantics architecture we can make a structure by the help of factory design pattern.

6. EXPERIMENTAL RESULTS

Figure 3 shows the sample protégé page which elaborate the owl classes and there hierarchy that to be followed during the collaboration of the different ontology.

As we know that there are three types of ontology classification that is,

- Single Ontology.
- Multiple Ontology and
- Hybrid Ontology

Here we can use multiple ontology as it requires ontology to be accessed from the different sections.

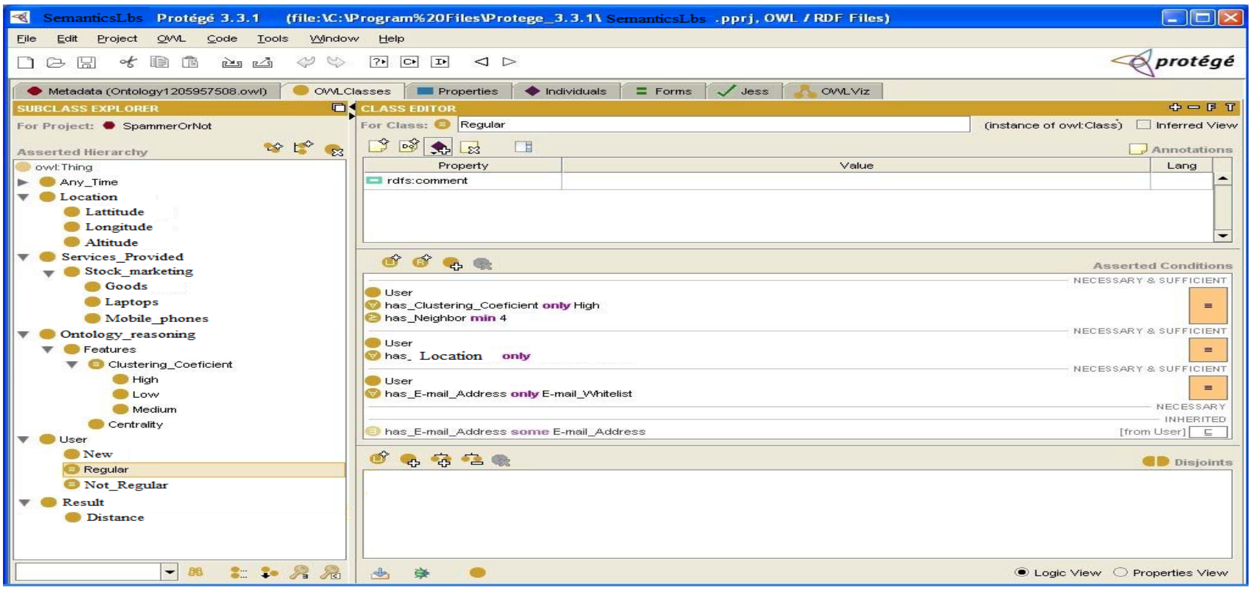

Fig 3: Sample Protégé Page.

7. CONCLUSION

We have studied and understood that ontology reasoning can be used to enable the semantic framework. Using the location based services we can contribute to the web services provided by any of the stock markets as well as the user based attributes can used to find out the nearby available sources to the user helping it all the way. This technology is growing day by day and is helping to introduce advance services.

Location Based Services plays an important role in realizing enhancing the usability of the Semantics framework design, the improvement of customers' relations and improving the requirement of system performance and so on. Ontology reasoning provides the support for the Semantic Framework design, and other business making decision, etc.

8. ACKNOWLEDGMENTS

The Location Based Services has provided many reviews and approach towards the new beginning of advance technology which can be achieved by enabling semantics over it. Also, Protégé is supported by a strong community of developers and academic, government and corporate users, who are using Protégé for knowledge solutions in areas as diverse as biomedicine, intelligence gathering, and corporate modelling.

Hence, this presented topic of dissertation has been supported and sponsored by Persistent System Limited (PSL), Nagpur Maharashtra, India. Our thanks to the experts who have contributed towards developing the project. PSL is known for its innovative approach towards the technical development and encouraging candidature to implement new ideas.

9. REFERENCES

[1] W3C Semantic Web: http://www.w3.org/2001/sw/

[2] Berners-Lee T., Heldler J., and Lassila O., 2001, "The Semantic Web," in Scientific American, vol. 284, pp. 34-43.

[3] W3C Web Ontology (WebOnt) Working Group (OWL): http://www.w3.org/2001/sw/ WebOnt/

[4] The Protégé Project: http://protege.stanford.edu

[5] Kong I., Kim J., Lee S., Jung-Tae Lee and Hwang W., 2005, "Design and Implementation of Ontology-based Context Reasoning Engine for Adaptive Multimedia Service Migration", IEEE digital library.

[6] Lassila O. and Khushraj D., 2005, "Contextualizing Applications via Semantic Middleware", IEEE proceedings of the Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'05).

[7] Alessandra Agostini, Claudio Bettini and Daniele Riboni, 2005, "Loosely Coupling Ontological Reasoning with an Efficient Middleware for Context-awareness", IEEE digital library.

[8] Ioanna Roussaki, Maria Strimpakou, Carsten Pils, Nikos Kalatzis, Miltos Anagnostou, 2006, "Hybrid context modeling: A location-based scheme using ontologies", Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06).

[9] Gunther Specht and Timo Weithoner, 2006, "Context-Aware Processing of Ontologies in Mobile

Environments", Proceedings of the 7th IEEE International Conference on Mobile Data Management (MDM'06).

[10] Dong LIU, Junliang CHEN, and Jie GUO, 2008, "An Ontology-based LBS System", IEEE.

[11] Radoslaw Brendel, Henryk Krawczyk, 2008, "E-mail User Roles Identification Using OWL-based Ontology Approach", Proceedings of the 1st IEEE International Conference on Information Technology, IT 2008.

[12] Heravi B. R., David Bell, Mark Lycett, 2010, "Towards an Ontology for Automating Collaborative Business Processes", 14th IEEE International Enterprise Distributed Object Computing Conference Workshops.

[13] Cheong Y. G., Kim Y., Yoo S. Y., Lee H., Lee S., Chae S. C. and Choi H., 2011, "An Ontology-Based Reasoning Approach Towards Energy-Aware Smart Homes", The 8th Annual IEEE Consumer Communications and Networking Conference - Emerging and Innovative Consumer Technologies and Applications.

[14] N. F. Noy, and D. L. McGuinness, 2001, "Ontology Development 101: A Guide to Creating Your First Ontology," Stanford University, Stanford, CA 2001.

[15] T. R. Gruber, 1995, "Toward Principles for the Design of Ontologies Used for Knowledge Sharing.," *International Journal of Human-Computer Studies*, vol. 43, pp. 907-928.

[16] A. Farquhar, R. Fikes, W. Pratt, and J. Rice, 1995, "Collaborative Ontology Construction for Information Integration," Stanford University, Stanford, CA.

[17] Y. Lu, 2003, "Roadmap for Tool Support for Collaborative Ontology Engineering," Computer Science, University of Victoria, Victoria.