A Model for Context-Data Representation and Management for Pervasive Environment

S.G. Gollagi, Dr.M.M. Math and V.J. Patwardhan

Abstract--- Combining computing and/or computing applications into surroundings, instead of having computers as discrete object is the aim of pervasive computing. Applications must adjust their behavior to every change in surroundings. Adopting involves proper capturing, maintaining and reasoning of context. This paper proposes effective representation of context in a hierarchical form and storing of context data in an object relational database than an ordinary database. Semantic of the context is managed by Ontology and Context data is handled by object relational database. These two modeling elements are associated to each other by semantics relations built in the ontology. The separation of modeling elements loads only relevant context data into the reasoned, therefore improving the performance of the reasoning process.

Keywords--- Context, Context Awareness, Context Aware Middleware, Ontology, Modeling, Reasoning.

I. INTRODUCTION

PERVASIVE computing system targets at constantly adapting their behavior in order to meet the needs of users within every changing physical, computing, social and communication context. Pervasive devices make ad-hoc connections among them and may be connected to different types of sensors to capture changes in the environment. Figure 1 shows the flow in the evolution chain from centralized computing to pervasive/ubiquitous computing as presented by [1] [2].

Context awareness is at the heart of pervasive computing problems. "Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves [14].

Context can be defined as an operational term whose definition depends on the intension for which it is collected and the interpretation of the operations involved on an entity at a particular time and space rather than the inherent characteristics of the entities and the operations themselves according to Dye & Winogards [3, 4]. The complexity of such problems increases in multiplicative fashion rather than

additive with the addition of new components into the chain. Pervasive Context aware computing has three major basic components: pervasive environment, Context management modeling and context-aware service. Stored and presented to the reasoning module. Context aware service performs context reasoning and decisions about the actions to be triggered.

Pervasive environment is characterized by dynamicity, heterogeneity and ubiquity of users, devices and other computing resources, ad-hoc connection among the devices and existence of hardware and software sensors. Context management modeling deals with how context data is collected, organized, represented, stored and presented to the reasoning module. Context aware service performs context reasoning and decisions about the actions to be triggered.

Figure 1: Evolution Chain of Pervasive Computing

A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task [14].

In this paper special emphasize on context data management which can be further used in the development of collaborative context application is proposed. Proper modeling, specification and definition of context and its management are essential for efficient reasoning, interpretation, and utilization of context data.

II. RELATED WORK

Data required for modeling are obtained from the applications using sensors. Sensors can be physical, virtual or logical sensors. After collecting the data from the application; it has to be represented in a suitable way for processing. Various context management and modeling approaches are introduced to present context for reasoning in different application area. Data from the sensors are presented using any of the following modeling approaches like key-value-pair modeling, Graphical modeling, Object Oriented modeling, logic based modeling, Mark up scheme modeling and Ontology modeling. Among all the modeling approaches ontology based context model is more suitable for context aware computing [2]. Ontology is defined as explicit specification of a shared conceptualization [4]. Context is modeled as concepts and facts using ontology. Some context

S.G. Gollagi, Research Scholar, Dept. of Comp. Science, Gogte Institute of Technology, Belagavi, India. E-mail: shantesh1973@rediffmail.com

Dr.M.M. Math Professor, Dept. of Comp. Science, Gogte Institute of Technology, Belgavi, India. E-mail: mmmath@git.edu

V.J. Patwardhan, UG Student, Dept. of Comp. Science, Hirasugar Institute of Technology, Nidasoshi, Karnataka, India. Email: vm.pvaidhi@gmail.com

DOI:10.9756/BJJSESC.8276

aware systems that use this approach are discussed. CONON (CONtext ONtology) [5] is based on treatment of high-level implicit contexts that are derived from low-level explicit context. It supports interoperability of different devices. It defines generic concepts regarding context and provides extensibility for adding domain specific concepts. Context reasoning in pervasive environment is time-consuming but is feasible for non time-critical applications. For time-critical applications the data size and rule complexity must be reduced. This is an infrastructure based environment. CoBrA-ONT [6] is a context management model that enables distributed agents to control the access to their personal information in context-aware environments. It is designed to overcome the interoperability of different devices. It is central part of CoBrA, broker-centric agent architecture in smart spaces. CoBrA-ONT assumes that there always exists a context broker server known by all the participants. It is infrastructure-centric and is not for pervasive computing. SOUPA (Standard Ontology for Ubiquitous and Pervasive Applications) [7] includes modular component vocabularies to represent intelligent agents with associated beliefs, desires and intention, time, space, events, user profiles, actions and policies for security and privacy. SOUPA is more comprehensive than CoBrA-ONT because it deals with more areas of pervasive computing and ontology can be reused. GAS Ontology [8] is ontology designed for collaboration among ubiquitous computing devices. The basic goal of this ontology is to provide a common language for the communication and collaboration among the heterogeneous devices that constitute these environments.

III. PROPOSED WORK

This section describes the working model of the proposed system. The context aware system has context acquisition layer, context middleware (representation layer, context management layer and decision layer) and application layer. Context acquisition layer gathers the context from the environment using sensors. Context representation layer represents context as entity relation hierarchy form. In the context management layer context is further classified as inactive context and active context. Predicates are used to decide the inactive context. For example context defined using predicates like ownedby are inactive and context defined using predicates like locatedIn are active. Inactive context are stored in Object relational database and active context are stored in Ontology. In the middleware, rules learned or rules derived from other rules are also maintained. Using the rules, relevant context from database and ontology are forwarded to the reasoning component. From reasoning layer appropriate context is sent to user in the application layer.

IV. CONTEXT REPRESENTATION

Context in terms of a statement that is made about the characteristics of the entities, their relationships and properties of the relationships [4]. Context can be personal, device, physical, activity, network, location etc. Personal entity provides contexts like person's identity, address, activity, location etc. Context can be represented as Entity, Hierarchy, Relation, Axiom and Metadata [9]. Hierarchy is a set of binary

relations that form an inversely directed acyclic graph. Relation is union of set of binary relations. Relation can be either entity relation or attribute relation. Entity relation has a set of binary relations having either its domain or range from the set of entity. Sub Entity of relation, Sub Property of, domain, range etc. is some relations used to link entities. Root of the hierarchy is a global entity called Context Entity. Entities and relations are sources of context data. Relation can be generic or domain based. The flow of context is shown in Figure 2.

Figure 2: Data Flow Diagram

For example in a generic level, relation is defined as person is Located In Location and in a domain level, relation is defined as Student is Located In Class. Attribute relation is the set of binary relations defined from set of entities to set of literals. Axiom is the axiomatic relations. Few generic level axiomatic relations are same As, inverse, symmetric and transitive. Meta data are information about a defined relation instance. Information like time of occurrence, precision, source of data can be a part of Meta information. Consider an example, Student is Located In Class at a given time t.

V. STORING CONTEXT DATA IN AN RELATIONAL DATABASE

Context represented in entity relation hierarchy form is stored in a relational database using the algorithm steps mentioned below. The attributes C Entity stores name of context entities. Attribute that stores name of the entity one step above in the hierarchy is-a relation. Layer stores whether an entity is in the generic or domain layer. Relation stores name of relations. Persistence of the relation stores whether an entity can be inactive or active [11]. Values of relations with static persistence are stored in the persistent context repository and values with dynamic persistence are stored temporarily for immediate use in the field Persistence. Value Form stores

source of value as a context entity or as a Literal. An attribute that stores name of instances is EInstance. Value is an attribute that stores value of the relation after applied to the instance. Timestamp stores context time. Source stores source of context. Precision stores precision of the context

VI. ADVANTAGES OF ORDBMS AND RDBMS

Relational models provide standard interfaces and query optimization tools for managing large and distributed context database or receive and send notification on context changes. Relational models are not designed for semantic interpretation of data. Relational database alone cannot be used to represent context in a pervasive environment. For semantic interpretations, ontology is used along with relational database. The Table 1 below summarizes the appropriateness of the approaches in relation to the necessary features. All approaches have strong and weak sides with respect to features for context management modeling [12]. Best of three worlds are combined to form a hybrid context management model. ORDBMS approach is more suitable than RDBMS approach. It ensures large storage capacity, quick access speed. Object relational database supports several storage units like collection list, arrays, types and UDTs (User defined data types) and most of them are represented as objects arrays. ORDBMS have a massive scalability compared to relational approach and excellent manipulation power of object databases. It supports rich data types by adding a new object-oriented layer. The systems are initially implemented by storing the inactive context to a relational database and active context to an ontology. Then the response time to get the relevant time is noted. Further system is implemented by replacing the storage of inactive context to relational database by object relational database. Then appropriate service can be provided to the user using service discovery [10].

Table 1: Comparison of RDBMS, ORDBMS and Ontology

Necessary Feature	Approach		
	Relational Approach	Ontology Approach	Object Relational Approach
Semantic Support	No	Yes	No
Ease of transaction	Yes	No	Yes
Query optimization	Yes	No	Yes
Reasoning Support	No	Yes	No
Formality	Yes	Yes	Yes
Scalability	Yes	No	Yes

VII. PARAMETERS

Basic metrics used to evaluate the relational database and object relational database are throughput or response time and storage usage. Throughput is the number of queries per unit time. Evaluation time can be wall-clock (real), Central Processing Unit (user), Input/output (system), server side versus client side. To compare the relational database and object relational database with respect to ontology, complexity of complex columns is considered. In the experimentation, it has been found that number of queries per unit time (ie. response Time) improved with the proposed work.

VIII. CONCLUSIONS

Context is represented using layered and directed graph. Layered organization helps to classify and tag context data as generic domain independent or as domain dependent. A combination of context model using ontology and object relational database is proposed. This paper focuses on context representation and storage of context. Reasoning and decision making of the context obtained from the context management are the future work.

REFERENCES

- [1] T. Strang and C. LinnhoPopien, "A Context Modeling Survey", In: UbiComp, LNCS, Vol. 3205, 2004.
- [2] M. Satyanarayanan, "Pervasive Computing Vision and Challenges", IEEE Personal Communications, Pp. 10–17, 2000.
- [3] A.K. Dey and G.D. Abowd, "Towards a Better Understanding of Context and Context Awareness", In: Proceedings of the CHI Workshop on the What, Who, Where and How of Context-Awareness, The Hague, The Netherlands, 2000.
- [4] T. Winograd, "Architectures for Context", Human-Computer Interaction Vol. 16, No. 2-4, Pp. 401–419, 2001.
- [5] X. Wang, D. Zhang, T. Gu and H.K. Pung, "Ontology Based Context Modeling and Reasoning using OWL", workshop on context modeling and reasoning. In: IEEE International Conference on Pervasive Computing and Communication, 2004.
- [6] H. Chen, "An Intelligent Broker Architecture for Pervasive Context-Aware Systems", PhD Thesis University of Maryland, Baltimore Country, USA, 2004.
- [7] E. Christopoulou and A. Kameas, "GAS Ontology: ontology for Collaboration Among ubiquitous Computing devices", International Journal of Human-Computer Studies, Vol. 62, No. 5, Pp. 664–685, 2005.
- [8] H. Chen, F. Perich and T. Finin, "SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications", In: International Conference on Mobile and Ubiquitous Systems: Networking and Services, Boston, USA, 2004.
- [9] B. Vanathi, U. Rhymend and V thariaraj, "Ontology based service discovery for context aware Computing. In: 1st IEEE International Conference on Advanced Computing.
- [10] D. Ejigu, M. Scuturici and L. Brunie, "An ontology-based approach to context modeling and reasoning in pervasive computing", In Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, Pp. 14-19, 2007.
- [11] M. Mamei and R. Nagpal, "Macro Programming through Bayesian Networks", Distributed Inference and Anomaly Detection; Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and communications, Pp. 87–96, 2007.
- [12] O. Yurur, L. Chi and W. Moreno, "A survey of context-aware middleware designs for human activity recognition", Vol. 52, Pp. 24–31, 2014.
- [13] G. Okeyo, L. Chen and H. Wang, "Combining ontological and temporal Formalisms for composite activity modelling and recognition in smart homes", Vol. 39, Pp. 29-43, 2014.
- [14] G.D. Abowd, A.K. Dey, P.J. Brown, N. Davies, M. Smith and P. Steggles, "Towards a better understanding of context and context-awareness," Pp. 304– 307, 1999