Designing Mobile Data Logging Framework using Ubiquitous Computing Environment

Dr. S. P. Tripathi[1], Prof.(Dr.) J. B. Singh[2], and Vivek Kumar Singh[3]

[1] *I.E.T. Lucknow-India, spt@ietlucknow.edu*
[2]*Shobhit University, Meerut-India, jbs.tomar@shobhituniversity.ac.in*
[3]*BBDNITM, Lucknow -India, viveksinghbbd@gmail.com*

Abstract

The focus of this paper is to develop a Ubiquitous framework which can record the incoming calls and sort them according to the service provider. This information will provide subsequent traffic on a particular network and hence will be useful in deciding the bandwidth allocation. Second part of the paper deals with application of constraint on the ontology using OCL thereby providing a strict check on accessing and filtering of the information.

Keywords: *Object Constraint Language (OCL)*

1. Introduction

The focus of this paper is to provide a framework that can record the calls coming to a particular mobile with special characteristics of identifying the service provider of each call. In order to achieve this goal we have identified two basic steps, the first step deals with the development of Ubiquitous computing environment as the calls coming will be from various sources. Secondly, development of mapping scheme of OCL to Data Base Management System. Since OCL [6] is being applied on Object Oriented Language [2] we will be representing the entire communication framework into Object Oriented Language. In database application it is beneficial as a part of database schema.

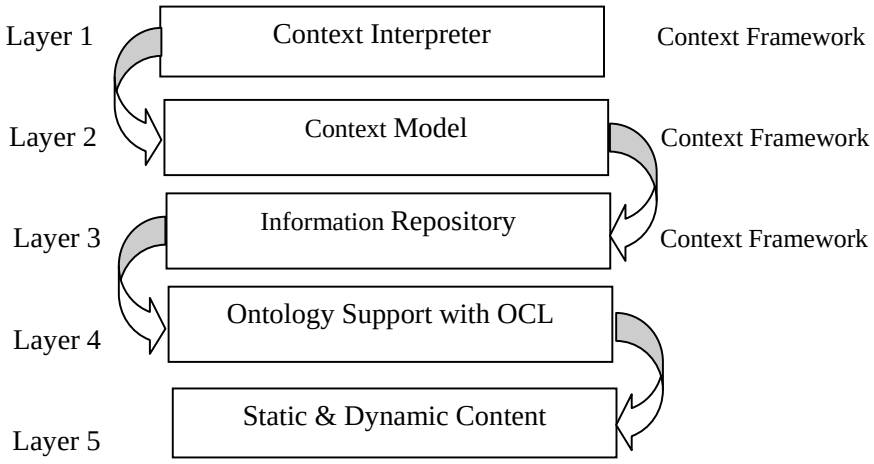

Figure 1. Framework for Ubiquitous Communication

The first three layers provide context information from different networks and delivering support of context model by ontology method. The Ontology framework is

supported by formal context model satisfying the need to facilitate context representation, context sharing and semantic interoperability of heterogeneous system.[4]

2. Framework for Ubiquitous Communication

Previous models like the model developed by Shekher Gupta which focuses on a system and a method for billing on incoming communication. A Code is received from a calling party and the incoming communications are allocated to an account of the calling party in response to receiving the code. Another method was developed by Saarenpaa Malti of Nokia Cooperation which provide a method for differentiated treatment of calling subscriber by introducing anew digit string depending on the subscriber associated parameter to the routing number of call. But both the framework locks exact separation of subscriber they only focus on the load on the system based on the in coming calls.

The focus of our model is to develop a framework which supports interoportability with various platforms.

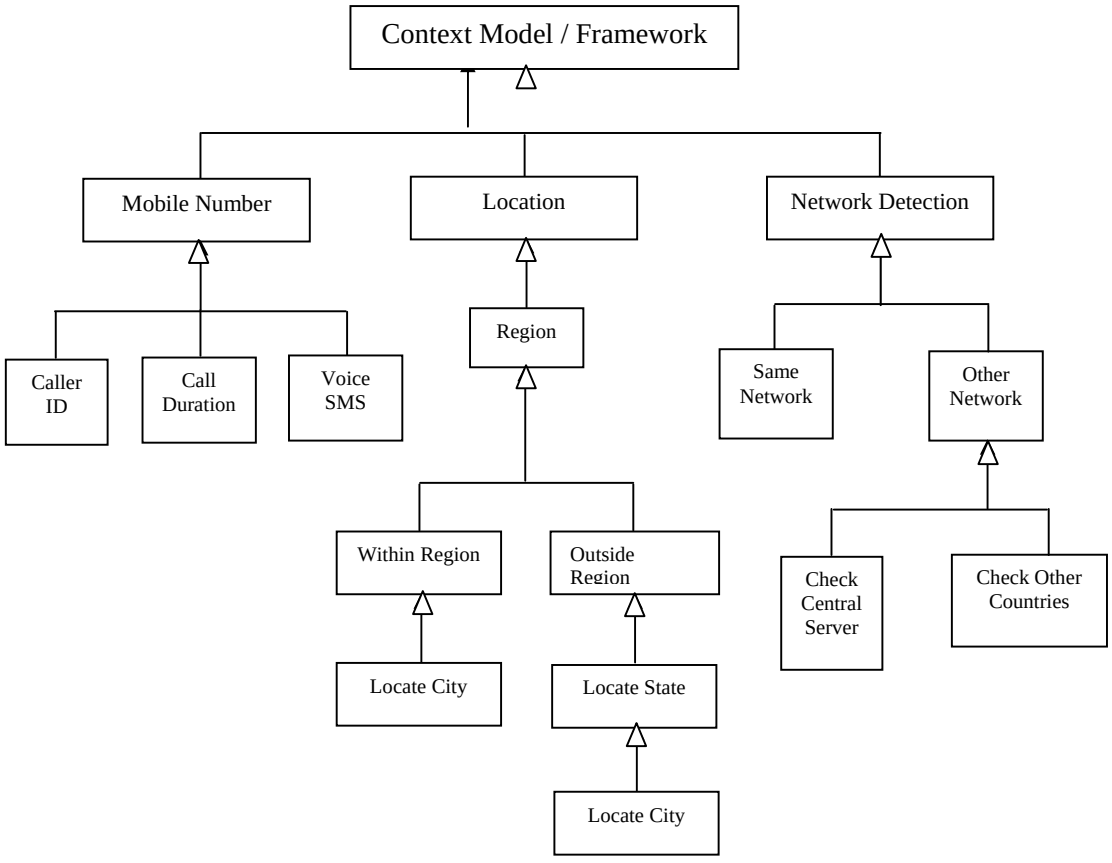

Figure 2. Class Representation of Ubiquitous Context Ontology

Based on Ontology contextaware computing can develop various existing logic reasoning mechanism to deduce high level context to lower level context. In the above framework we are able to represent the entities in a structured way describing the physical or conceptual object.

This ontology framework can be further extended to derive the relationship between various Class categories.

The Dynamic Ontology [1] aspects shows inherited relationship with the specific attributes of the class are represented.

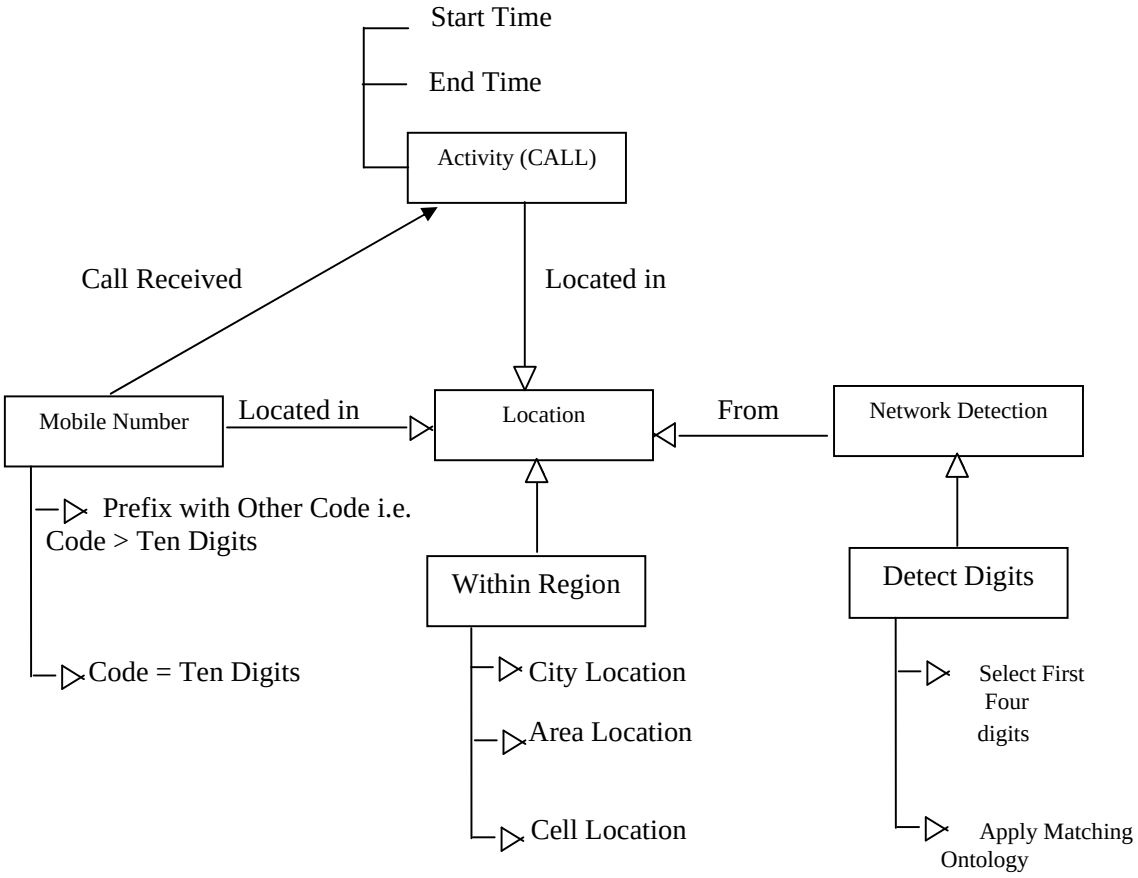

Figure 3. Dynamic Ontology Mapping

The system architecture for the above domain consists of Environment Manager, Context Manager and Task Manager [2]. The Environment Manager detects the region where the current user is and position of the user within network topology. When a user moves from one place to another the Ubiquitous environment provides an automatic movement of computing environment through Task Manager. The Communication Manager present in Environment Manager detects the entrance of the user end and the Task Manager brings the information related to the user. The job of the Task Manager is to perform three functions:

1. Detection of the mobile number whether it is more than ten digits or equal to ten digits.
2. If the location is within same region then detect city location, Area Location and Cell Location.

3. Lastly, detection of the digits, generally the first four digits are the key-id of the service provider like : 9415 - BSNL
   9918 – VODA PHONE
   9253 – TATA INDICOM
   9353 – RELIANCE COMMUNICATION…. etc

The number is picked up by the help of RMI mechanism in JAVA which exploits low level message passing. Number requester sends and receives messages to Number Provider on remote host, and Number Provider sends a number to Number Receiver. The entire message passing is done by the help of Simple Object Access Protocol (SOAP), which is an XML based light weight protocol for exchange of information in a decentralized, distributed environment. For every request a number that is being received by the tracer (Number Provider) present in network Detection a corresponding Apache SOAP deployment descriptor in XML is generated.

<Soap> <Phone List_Phone Number/></Soap>

As a primitive element Ubiquitous computing provides an intelligent mapping by the help of SOAP descriptors and RMI support.

3. Applying Constraint on Data Base :

A constraint as given by Kleppe is defined as restriction on one or more values of an object – oriented model or system. A restriction of the multiplicity of an association can be expressed either directly by using syntactical construct of Object Oriented Notation. Basically three types of constraints can be classified in Database Management System:

1. **Implicit Constraint:** An Implicit Constraint, which represents an integrity rule applied on data models.
2. **Explicit Constraint:** An Explicit Constraint, which represents the Business Rules.
3. **Inherent Constraint:** An Inherent Constraint, which are specified in a schema but are assumed to hold by the definition of relational model.

The order to apply constraint on the database we have used OCL invariant on SQL Table consisting of details of the Caller, its numbers, location. We have to develop a transformation pattern.

OCL Invariant :

NAME : OCL INVARIANT CREATION

Description : OCL for information retrieval of Caller Number detail.

Context < class name > inv < constraint name > : < OCL expression (self) >
is transmitted to :
< class name > all Instance → for All (< OCL expression (Self)>)

Now applying the invariant in our Ontology Schema

OCL Expression:

Context Call inv Number type :

Self.Number type → for all (first four digit = false / true)
Self. Service provider → Select (c: Company / C.location = within region.cellarea)

The OCL expression covers all the collection operations that result in a boolean value, such OCL expression can be basically mapped to SQL predicate with or without nested sub queries. The goal of our approach is to provide a procedural SQL Code [3] with full OCL language.

4. Conclusion:

This paper focuses on the development of Unified Ubiquitous computing environment using context modeling. As a primitive element of Ubiquitous Computing, the context framework provides an intelligent context of process of entering date by capturing general concepts of basic context and extending it for domain specific Ontology in a hierarchical manner. The heterogeneous network is also taken into account and detection of number plays an important role. In the second stage we are able to apply OCL for the specification of database integrity constraints. In future we will try to in corporate "pre" and "post" aspects of executable SQL Code.

References

[1] Bruce, K "A Pattern Language for Object RDBMS Integration, Knowledge System Group."
[2] Dresden UML Tool Set, http://www.st.inf.tu.dresden.de
[3] Melton, J "SQL's Stored procedures " A complete guide, Morgan Kaufman, 1998
[4] M. Weiser, "The Computer for the Twenty – First century" Scientific Aug 1991, pp 94 -101.
[5] N. Roy & S. K. Das, "Enhancing Availability of Grid Computational Services to Ubiquitous Computing Application ", proceeding's 19th IEEE International Symposia on parallel and Distributed Computing, 2005.
[6] Wanner; J & Kleppe, A: "The Object Constraint Language", Addison Wesley, 1999.

Authors

[1]**Dr. S. P. Tripathi** is currently working as a Assistant Professor & Head in Department of Computer Science & Engineering at I.E.T. Luckniw. He has total twenty two years teaching experience in computer science & engineering field. He has published number of papers in referred National journals.

His teaching areas are : Software Engineering, Database Management , Operating Systems , Data Mining, Computer Network

[2]**Prof. J. B. Singh** is currently working as a Director in School Computer Engineering & Information Technology at Shobhit University, Merrut. He has total Thirty Eight year teaching experience in computer science & engineering field. He has published number of papers in referred National journals.

His teaching areas are : Bio-Statistics, Statistics, Mathematics, Computer Science, Mathematical Modeling, Statistical Computing, Crop Yield Estimation, Operation Research, Neural Network and Computer application

[3]**Vivek Kumar Singh** is currently working as a Assistant Professor in Department of Information Technology at B.B.D.N.I.T.M. (Babu Banarasi Das National Institute of Technology & Management), Lucknow. He has total Eight year teaching experience in Computer Science & Engineering field. He has published number of papers in referred National journals.

His teaching areas are : Theory of Automata & Formal Language, Design & Analysis of Algorithm & Computer Architecture

* **Corresponding Author**