Semantic-PlaceBrowser: Understanding Place for Place-Scale Context-Aware Computing

Anh-Tuan Nguyen, Seng Wai Loke, Torab Torabi, Hongen Lu
Department of Compute Science and Computer Engineering
La Trobe University, Victoria, 3086, Australia
Email: {t.nguyen, s.loke, t.torabi, h.lu}@latrobe.edu.au

***Abstract*—Our view of environment for context-aware applications is a place, which contains devices, digital services and people with their social contexts. Therefore, understanding place will contribute to the successful of applications. In this paper, we introduce two issues: (i) there is a need for sharing context knowledge about place; (ii) using a multiagent framework for modeling place will improve the system scalability and robustness. To support the idea, we have built a tool called Sematic-PlaceBrowser for discovering place context and contributing to the knowledge base. The knowledge base can be shared and contributed among different applications. The Semantic-PlaceBrowser can be deployed at different scales, such as from a place as small as a room or as big as a shopping complex.**

I. Introduction

Pervasive computing environments are becoming larger where there is a trend towards larger scale computing such as entire shopping complexes and urban environments. In this paper, we view the pervasive computing environment as a place, which contains computers and people with their relationships. Places can be different in scale; it can be a room or can be a suburb. We examine a place in which our activities and social relationships happen with the enhancement of computer-mediated services. That environment contains a group of computers, people and services in a physical place. We called that environment a Place-Based Virtual Community (PBVC). The Place-based Virtual Community (PlaceComm for short) is a kind of community that includes both place-based community and virtual community characteristics: common goals interest, share the same physical place, communications happen at a place are both face-to-face and mediated by computers.

The community is supported by both place-specific digital services, and online services, where environment awareness is supported by sensing devices. In other words, PlaceComm is a virtual community that is superimposed on the physical place, characterized by the community that occupied that place in a period of time. We do not model the world like Nexus' approach [1], but model the physical place in a PlaceComm with specific sensed contexts.

In addition, people may interact with multiple PBVCs. A PBVC may overlap with or be located within another. For example, when users go to a shopping center, they can be both in the shopping center community and the supermarket community within that shopping center at the same time. People can dynamically join communities to use services when they are inside the communities. Thus, effectively, the user is within a stack of PBVCs which changes with the user's movements into and out of communities. The stack structure is implied by the nesting of one community within another area, corresponding to simple geographical containment or overlap. In other words, communities can be stacked up [2].

At first look, things in a particular place are not related to each other. So why would we call them a community? Actually, even though they are not aware of each other, they share the same context of the environment. They physically belong to that place no matter they are aware of it or not. They are in the environment and contribute context to the environment. It would be very useful if there exists an affordable tool that can help people discover and understand the place as well as contribute back their context to the place for other people to use.

In order to build a system that facilitates the contribution and sharing of context information, we found that the multiagent system approach is a suitable solution, because multi-agent systems are considered a promising approach for building complex applications [3]. Using a multiagent framework for modeling the pervasive computing environments will improve the system scalability and robustness.

To illustrate this idea, we introduce an application called Semantic-PlaceBrowser, which is implemented as a client for exploiting our PlaceComm framework. The Semantic-PlaceBrowser is set of applications that runs on a Nokia N95 mobile phone and can discover hidden context of place by using the phone's built-in sensors. In addition, the Semantic-PlaceBrowser can send queries to the PlaceComm Framework to get information from the knowledge base. The PlaceComm framework is a multiagent system that facilitates multiple users and can be deployed in a variety of places.

Our contributions are to:

- develop Semantic-PlaceBrowser, a tool for sensing context of place and contributing context to the knowledge base;
- develop a knowledge base for context sharing and reusing amongst applications; and
- present our PlaceComm Framework for developing context-aware applications in the large scale using agents technology.

This paper is organized as follows. Section 2 introduces the PlaceComm framework with its ontology and knowledge base.

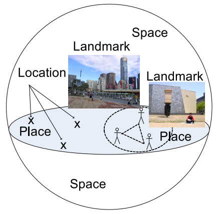

Fig. 1. Location, Space, and Place

Section 3 introduces the Semantic-PlaceBrowser with two usage scenarios: physical browsing and semantic browsing. The continuing section is related work. The final section is conclusion and future work.

II. The PlaceComm Framework

We view the pervasive computing environment as a place with all of these aspects: location, people present, activities at that place and relationships among them. There are three words we might find when speaking about place: location, space and place. Space is the 3D structure of the physical world that contains people and things. In contrast, place is a space with experiences due to human behaviors, cultural expectation and social meaning: we act in place [4]. Each place has a specific meaning that "shapes actions" or behaviors [5]. A general accepted definition for place is a meaningful location. However, place is not only a meaningful location. Harrison et al. claimed that "Space is the opportunity; place is the understood reality; we are located in space, but we act in place" [4]. Similarly, Cresswell looks at place in a way of understanding the world [6]. The author claimed that "place needs to be understood as an embodied relationship with the world. Places are constructed by people doing things and in this sense are never "finished" but are constantly being performed".

PlaceComm is a framework for developing context-aware applications with place as a key abstraction. PlaceComm is built using the JADE/LEAP framework [7]. PlaceComm takes the view of pervasive environments as a PBVC. The PlaceComm framework contains four main components:

- Sensing components: takes care of context information sensing and gathering.
- Knowledge Base: used for storing context knowledge as well as answering semantic queries about context.
- PlaceComm API: is the collection of library for building agents for modeling the Place-based Virtual Community.
- Service Enabler: is a collection of API and agents that enable user to build their own service with predefined format.

A. *PlaceComm System Architecture*

Figure 2 shows the system architecture of the PlaceComm framework. The architecture is divided into four layers: context gathering layer, context-processing layer, community layer and application layer.

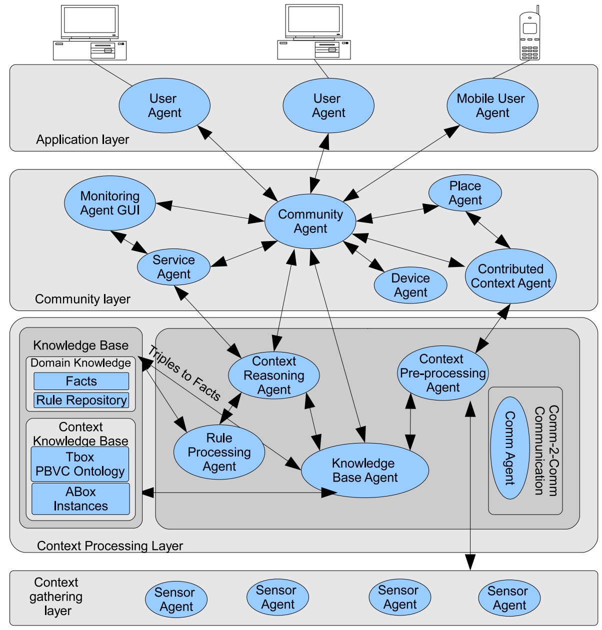

Fig. 2. PlaceComm Framework Architecture

The first layer is context gathering layer. It contains physical sensors managed by agents that can sense the environment to get context information.

The second layer is the Context-Processing Layer. This layer take cares of all activities related to context. The *Context-Preprocessing Agent* receives sensed context information from *Sensor Agents* and preprocesses them before passing information to the *Knowledge Base Agents* (KBA) to insert into the *Context Knowledge Base* (CKB). The core component in here is the CKB which is built on top of the PBVC ontology (see figure 3). The *Knowledge Base Agent* is responsible for knowledge insertion and query answering. It can also convert the query results to facts and put into rule repository for context reasoning. We use the Jessrules as a rule engine for our system [8]. The *Context-Reasoning Agent* will take advantage of the Jess rules engine and the *Knowledge base Agent* to answer complex queries as well as provide reasoning services for applications. The *Comm Agent* is in charge of community-to-community communications realizing the community stack concept [2].

The third layer is the community layer. It includes six different types of agents that represent a PBVC: *Community Agent*, *Place Agent*, *Contributed-Context Agent*, *Device Agent*, *Service Agent* and *Monitoring Agent*. The *Community Agent* acts as a "receptionist" of the community. It knows other agents in the community. Whenever a request is sent to the community, this agent receives and finds a suitable agent to that task and introduces them together. The roles of *Community Agent* are similar to the roles of inetd super-server in the unix operating system.

The top layer is the application layer, it contains *User Agent* (UA) and *Mobile User Agent* (MUA). UA and MUA are special kind of agents that can allow the user to communicate

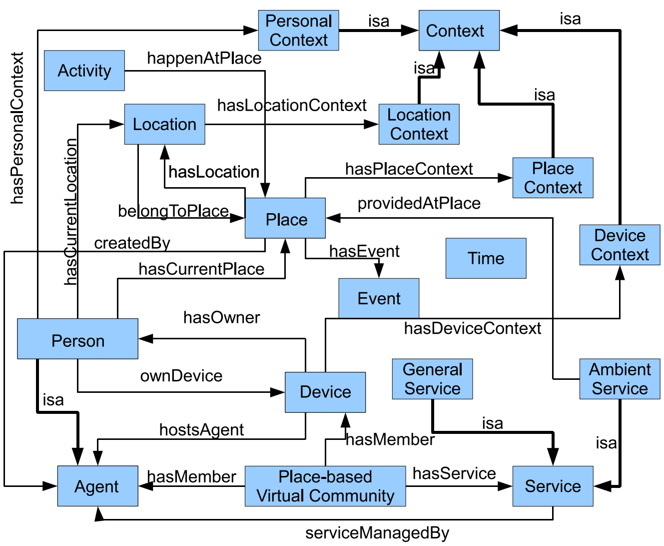

Fig. 3. The PBVC Ontology

with other software agents in the lower layer. It also has the ability to communicate and to collaborate with other user agents in different applications. This is the most powerful feature of the multiagent system. The MUA is built for running on mobile devices such as Nokia N95 which is strong on mobility and sensing devices. The UA is built for running on resource rich device, such as laptop or desktop which is strong on the resouce rich and powerful computational ability.

B. PBVC Ontology

The PBVC ontology stores core concepts and vocabulary for implementing the knowledge base for the PlaceComm framework. The PBVC ontology is illustrated in Figure 3. These relationships are predicates that help the reasoner traverse through nodes to obtain knowledge. In order to keep track of context history, we provide a Context class which acts like a context repository for place,location, people, and devices.

III. THE SEMANTIC-PLACEBROWSER

We are consend that capturing and understanding the context of place is crucial for context-aware applications. Therefore we build an application called: Semantic-PlaceBrowser which is not only can sense the environment but also can discover hidden context using semantic browsing technique based on ontology reasoning and context reasoning.

Figure 4(a) shows the system architecture for the Semantic-Place Browser. The architecture is divided into three layers depended on technology is used. The lowest layer is separate program implemented using Python and C++ to get the low level sensors information such as the accelerometor and Wireless LAN information. This layer communicate with second layer using TCP/IP socket. The second layer which we improve on our previous work on PlaseSense [9], we make use of J2ME technologies to sense Bluetooth devices (JSR82), get cell towers (JSR120) and GPS coordinates (JSR179). .The sematic-browsing layer is implemented using the JADE/LEAP agent [7]. The UserAgent here only sends SPARQL queries and receive answers to the Community Agent as a string data. It leaves the computing consuming task to the server side which comprises the community of agents (See figure 2)

(a) Semantic-PlaceBrowser Architecture (b) Low level browsing (c) SPARQL query

Fig. 4. MUA and running screen shots

Figure 4(b) shows the physical browsing results for low-level browsing. Figure 4(c) shows the SPARQL query on mobile phone will be sent to the Community Agent.

The semantic browsing functionality includes two parts: ontology reasoning and context reasoning based on rules. The ontology reasoning makes use of Jena API to check consistency, classification type of context. The context reasoning is a combination of two steps: extract relevant context information from knowledge base then pass the result to jess rule engine via a triples-to-facts conversion procedure inside Knowledge Base Agent to provide a deducted context. We do not convert the whole knowledge base into facts in Jess engine like OWL2Jess method [10]. By this approach, we can improve the computational speed by working in a smaller set of triples acquired from the query. The context information extraction is implemented using SPARQL queries. We have implemented 32 predefined queries for getting knowledge of the community. Due to the page limit, we only introduce a SPARQL query that can be sent from MUA to find where are Tuan's friends from timestamp T1 to timestamp T2 in listing 1

Listing 1. Where are Tuan's friends?

```
SELECT    ?location   ?person
WHERE {?location   rdf:type  plc:Location.
?location   pbvc:hasTimeInstant ?timestamp.
?timestamp  pbvc:timestamp  ?value.
?location   pbvc:belongtoPersonalContext
            ?PersonalContext.
?PersonalContext  pbvc:createdByPerson
                  ?person.
?person  foaf:friendOf  ?personTuan.
?personTuan  foaf:name  ?nameOfTuan
FILTER  ((?value  >=T1  &&  ?value  <=  T2)  &&
(?nameOfTuan=  "TuanNguyen"^^xsd:string))}
```

IV. RELATED WORK

The idea of browsing the environment interests many researchers [11], [12], [13]. However, little research focused on sharing the captured context information amongst different applications.

We are inspired by the idea of people centric sensing [14], where the community knowledge is contributed and enriched by members using Semantic-PlaceBrowser. Further more, we also provide a framework which includes the knowledge base and the mechanism for reusing and contributing amongst applications. Therefore, when building a new application for the same environment, we do not have to worry about the context sensing and capturing part. We just focus on using the sensed context more efficiently.

Castelli et al. describe the term "Browsing the world" which is the vision that in the near future, we can browse not only the web but also the physical world around us by using sensors to discover objects tagged with RFID in the environment[15]. In contrast, Semantic-PlaceBrowser browse real devices in the world: Bluetooth and WLAN. In the future, the RFID tags discovery can be implemented via using Bluetooth RFID reader.[1]

Nakamura et al. in [12] introduced a AmbientBrowser application that helps users browse web pages in daily activities without touching the computer. The AmbientBrowser's approaching looks similar to the Semantic-PlaceBrowser at first look. The AmbientBrowser focuses on loading web pages from the Internet that have content topics relevant to the keywords stored inside the RFID tag wearing by the users, while Semantic-PlaceBrowser is a "magnifier" to discover the surrounding physical environment by using sensing technology and semantic technology.

Another project called PlaceBrowser has been discussed in [13]. PlaceBrowser is a PDA based application that allows the user of the application to navigate around an area of geographical interest, such as a city, using a zoomable, panable hierarchy of aerial images, in a fashion similar to Google Maps. The novel aspect to the work is that an area of precise interest within the map can be pin-pointed by the user by directly dragging out a rectangular area on the map.

The novel idea of the Semantic-PlaceBrowser is the ability of sensing physical environments and discovering the new context that cannot be directly sensed by sensors but obtained by inference. Furthermore, the knowledge base facilitates context storing for sharing amongst applications.

V. Conclusion and Future Work

Currently, the Semantic-PlaceBrowser can sense bluetooth devices, wifi ID, cell towers and detect the movement and acceleration. In mobile phone there are still more technologies that can be used for sensing such as microphone, camera, light sensor and radio. In the future, we will deploy in many users mobile device for large scale performance measurement. Furthermore, the knowledge base with rules will strengthen the ability to acquire reasoned context. These approaches require much effort for implementation; therefore we leave this for future work.

In conclusion, for developing context-aware computing applications on a large scale where many applications are acquiring context, a shared context knowledge base for the environment is extremely useful. The Semantic-PlaceBrowser with the PlaceComm framework is a proof of concept application that facilitates the context gathering and contribution to the knowledge base with multi-user support.

Acknowledgment

The first author is sponsored by the Project 300, of Ho Chi Minh City, Vietnam, Contract No 2377/QD-UBND,25/5/2006

References

[1] D. Nicklas and B. Mitschang, "The nexus augmented world model: An extensible approach for mobile, spatially-aware applications," in *Proceedings of the 7th International Conference on Object-Oriented Information Systems : OOIS '01 ; Calgary, Canada, August 27-29, 2001*, Y. Wang, S. Patel, and R. Johnston, Eds. London: Springer-Verlag, Januar 2001, Konferenz-Beitrag, pp. 392–401.

[2] T. Nguyen, S. W. Loke, and T. Torabi, "The community stack: Concept and prototype," in *AINAW '07: The 21st Int. Conf. on Advanced Information Networking and Applications Workshops*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 52–58.

[3] M. Luck, P. McBurney, and C. Preist, "A manifesto for agent technology: Towards next generation computing," *Autonomous Agents and Multi-Agent Systems*, vol. 9, pp. 203–252, 2004.

[4] S. Harrison and P. Dourish, "Re-place-ing space: the roles of place and space in collaborative systems," in *CSCW '96: Proceedings of the 1996 ACM conference on Computer supported cooperative work*. New York, NY, USA: ACM, 1996, pp. 67–76.

[5] O. Brown, "Place as a practical concern of mobile workers," *Environment and Planning A*, vol. 35, no. 9, pp. 1565–1587, 2003. [Online]. Available: http://www.envplan.com/abstract.cgi?id=a34231

[6] T. Cresswell, *Place: A Short Introduction*. Wiley-Blackwell, 2004.

[7] F. Bergenti and A. Poggi, "Leap: A fipa platform for handheld and mobile devices," in *ATAL '01: Revised Papers from the 8th International Workshop on Intelligent Agents VIII*. London, UK: Springer-Verlag, 2002, pp. 436–446.

[8] E. F. Hill, *Jess in Action: Java Rule-Based Systems*. Greenwich, CT, USA: Manning Publications Co., 2003.

[9] T. Nguyen, S. W. Loke, T. Torabi, and H. Lu, "Placesense: A tool for sensing communities," *Wireless Pervasive Computing, 2009. ISWPC 2009. 4th International Symposium on*, pp. 1–5, Feb. 2009.

[10] J. Mei, E. P. Bontas, and Z. Lin, "Owl2jess: A transformational implementation of the owl semantics," in *In Proceedings of International Workshops on ISPA, LNCS 3759*, 2005, pp. 599–608.

[11] G. Castelli, M. Mamei, A. Rosi, and F. Zambonelli, "Browsing the world: Bridging pervasive computing and the web," in *Int. Workshop on Ubiquitous Information Systems. Münster, Germany.*, 2006, p. 7.

[12] S. Nakamura, M. Minakuchi, and K. Tanaka, "Ambientbrowser: Web browser in everyday life," in *Ambient Intelligence in Everyday Life*, vol. 3864/2006. Springer Berlin/Heidelberg, 2006, pp. 157–177.

[13] D. Bainbridge, M. Jones, and D. Arter, "A map-based placebrowser for a pda," University of Waikato, Department of Computer Science, New Zealand, Tech. Rep., 2007. [Online]. Available: http://hdl.handle.net/10289/759

[14] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn, "The rise of people-centric sensing," *IEEE Int. Comp.*, vol. 12, no. 4, pp. 12–21, 2008.

[15] G. Castelli, A. Rosi, M. Mamei, and F. Zambonelli, "A simple model and infrastructure for context-aware browsing of the world," in *Fifth Annual IEEE International Conference on Pervasive Computing and Communications, 2007. PerCom '07.*, March 2007, pp. 229–238.

[1] http://www.rfid-in-china.com/2008-11-09/products_detail_2162.html