An Intelligent Broker for Context-Aware Systems*

Harry Chen
University of Maryland
Baltimore County
hchen4@cs.umbc.edu

Tim Finin
University of Maryland
Baltimore County
finin@cs.umbc.edu

Anupam Joshi
University of Maryland
Baltimore County
joshi@cs.umbc.edu

ABSTRACT

We describe Context Broker Architecture (CoBrA) – a new architecture for supporting context-aware systems in smart spaces. Our architecture explores the use of Semantic Web languages for defining and publishing a context ontology, for sharing information about a context and for reasoning over such information. Central to our architecture is a broker agent that maintains a shared model of the context for all computing entities in the space and enforces the privacy policies defined by the users and devices. We also describe the use of CoBrA in prototyping an intelligent meeting room.

Keywords

Context-aware systems, smart spaces, semantic web, agent architecture

1. INTRODUCTION

Context-aware systems are computing systems that provide relevant services and information to users based their situational conditions [3]. Among the critical research issues in developing context-aware systems are context modeling, context reasoning, knowledge sharing, and user privacy protection. To address these issues, we are developing an agent-oriented architecture called Context Broker Architecture that aims to help devices, services and agents to become context aware in smart spaces such as an intelligent meeting room, a smart vehicle, and a smart house.

By context we mean a collection of information that characterizes the situation of a person or a computing entity [3]. In addition to the location information [6], an understanding of context should also include information that describes system capabilities, services offered and sought, the activities and tasks in which people and computing entities are engaged, and their situational roles, beliefs, desires, and intentions.

Research results show that building pervasive context-aware systems is difficult and costly without adequate support from a computing infrastructure [1]. We believe that to create such infrastructure requires the following: (i) a collection of ontologies for modeling context, (ii) a shared model of the current context and (iii) a declarative policy language that users and devices can use to define constraints on the sharing of private information and protection of resources.

The need for common ontologies. An ontology is a formal, explicit description of concepts in a domain of discourse (or classes), properties of each class describing various features and attributes of the class, and restrictions on properties [8]. In order to create computer systems that can "understand" and make full use of a context model, the contextual information must be explicitly represented so that they can be processed and reasoned by the computer systems. Furthermore, shared ontologies enable independently developed context-aware systems to shared their knowledge and beliefs about context, reducing the cost of and redundancy in context sensing.

The need for a shared context model. CoBrA maintains a model of the current context that can be shared by all devices, services and agents in the same smart space. The shared model is a repository of knowledge that describes the context associated with an environment. As this repository is always accessible within an associated space, resource-limited devices will be able to offload the burden of maintaining context knowledge. When this model is coupled with a reasoning facility, it can provide additional services, such as detecting and resolving inconsistent knowledge and reasoning with knowledge acquired from the space.

The need for a common policy language. CoBrA includes a policy language [5] that allows users and devices to define rules to control the use and the sharing of their private contextual information. Using this language, the users can protect their privacy by granting or denying the system permission to use or share their contextual information (e.g., don't share my location information with agents that are not in the CS building). Moreover, the system behavior can be partially augmented by requesting it to accept new obligations or dispensations, essentially giving it new rules of behavior (e.g., you should inform my personal agent whenever my location context has changed).

2. CONTEXT BROKER ARCHITECTURE

Our architecture differs from the previous systems [3, 7] in the following ways:

- We use Semantic Web languages such as RDF and the Web Ontology Language OWL [8] to define ontologies of context, which provide an explicit semantic representation of context that is suitable for reasoning and knowledge sharing. In the previous systems, context are often implemented as programming language objects (e.g., Java class objects) or informally described in documentation.
- CoBrA provides a resource-rich agent called the *context*

*This work was partially supported by DARPA contract F30602-97-1-0215, Hewlett Packard, NSF award 9875433, and NSF award 0209001.

broker to manage and maintain a shared model of context[1]. The context brokers can infer context knowledge (e.g., user intentions, roles and duties) that cannot be easily acquired from the physical sensors and can detect and resolve inconsistent knowledge that often occurs as the result of imperfect sensing. In the previous systems, individual entities are required to manage and maintain their own context knowledge.

- CoBrA provides a policy language that allows users to control their contextual information. Based on the user defined policies, a broker will dynamically control the granularity of a user's information that is to be shared and select appropriate recipients to receive notifications of a user's context change.

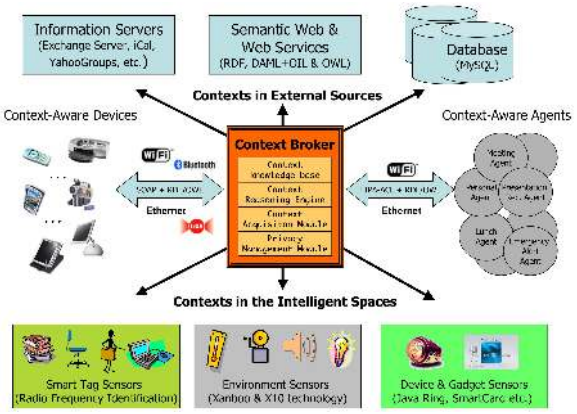

Figure 1: A context broker acquires contextual information from heterogeneous sources and fuses it into a coherent model that is then shared with computing entities in the space.

Figure 1 shows the architecture design of CoBrA. The context broker is a specialized server entity that runs on a resource-rich stationary computer in the space. In our preliminary work, all computing entities in a smart space are presumed to have priori knowledge about the presence of a context broker, and the high-level agents are presumed to communicate with the broker using the standard FIPA Agent Communication Language [4].

3. EASYMEETING: AN INTELLIGENT MEETING ROOM

To demonstrate the feasibility of our architecture, we are prototyping an intelligent meeting room system called **EasyMeeting**, which uses CoBrA as the foundation for building context-aware systems in a meeting room. This system will provide different services to assist meeting speakers, audiences and organizers based on their situational needs.

We have created an ontology called COBRA-ONT [2] for modeling context in an intelligent meeting room. This ontology, expressed in the OWL language, defines typical concepts (classes, properties, and constraints) for describing places, agents (both human and software agents), devices, events, and time. We have also prototyped a context broker in JADE[2] that can reason about the presence of a user in a meeting room. In our demonstration system, as a user enters the meeting room, his/her Bluetooth device (e.g., a SonyEricsson T68i cellphone or a Palm TungstenT PDA) sends an URL of his/her policy to the broker in the room[3]. The broker then retrieves the policy and reasons about the user's context using the available ontologies. Knowing the device owned by the user is in the room and having no evidence to the contrary, the broker concludes the user is also in the room.

4. FUTURE WORK AND REMARKS

We believe an infrastructure for building context-aware systems should provide adequate support for context modeling, context reasoning, knowledge sharing, and user privacy protection. The development of CoBrA and the EasyMeeting system are still at an early stage of research. Our short-term objective is to define an ontology for expressing privacy policy and to enhance a broker's reasoning with users and activities by including temporal and spatial relations. A part of our long-term objective is to deploy an intelligent meeting room in the newly constructed Information Technology and Engineering Building on the UMBC main campus.

REFERENCES

[1] CHEN, G., AND KOTZ, D. A survey of context-aware mobile computing research. Tech. Rep. TR2000-381, Dartmouth College, Computer Science, Hanover, NH, November 2000.

[2] CHEN, H., FININ, T., AND JOSHI, A. An ontology for context-aware pervasive computing environments. *Special Issue on Ontologies for Distributed Systems, Knowledge Engineering Review* (2003).

[3] DEY, A. K. *Providing Architectural Support for Building Context-Aware Applications*. PhD thesis, Georgia Institute of Technology, 2000.

[4] FIPA. *FIPA ACL Message Structure Specification*, December 2002.

[5] KAGAL, L., FININ, T., AND JOSHI, A. A policy language for a pervasive computing environment. In *Proceedings of the IEEE 4th International Workshop on Policies for Distributed Systems and Networks* (2003).

[6] PRIYANTHA, N. B., CHAKRABORTY, A., AND BALAKRISHNAN, H. The cricket location-support system. In *Proceedings of MobiCom 2000* (2000), pp. 32–43.

[7] SCHILIT, B., ADAMS, N., AND WANT, R. Context-aware computing applications. In *Proceedings of the 1st IEEE WMCSA* (Santa Cruz, CA, US, 1994).

[8] SMITH, M. K., WELTY, C., AND MCGUINNESS, D. Owl web ontology language guide. `http://www.w3.org/TR/owl-guide/`, 2003.

[1]Notice that we have a broker associated with a given space, which can be subdivided into small granularities with individual brokers. This hierarchical approach with collaboration fostered by shared ontologies helps us avoid the bottlenecks associated with a single centralized broker.

[2]Java Agent DEvelopment Framework: `http://sharon.cselt.it/projects/jade/`

[3]The description of the URL is sent to the broker in a vNote via the Bluetooth OBEX object push service