A Semantic and Adaptive Context Model for Ubiquitous Computing

Yunting Tang[1] and Qing Wu[2]

[1] Ningbo Institute of Technology, Zhejiang University,
Ningbo, Zhejiang, China, 315001
`tyt@nit.net.cn`
[2] College of Computer Science, Zhejiang University,
Hangzhou, Zhejiang, China, 310027
`wwwsin@cs.zju.edu.cn`

Abstract. Ubiquitous computing pursues naturalness and harmony. We think that semantic and adaptation are two important aspects. This paper presents a semantic and adaptive context model for ubiquitous computing. We emphasize the fusion of semantic information and context model in smart vehicle space to support ubiquitous computing. In our work, we use a web ontology language to model ontologies of context including the common essential ontology and the application domain-specific ontology. In addition, we present an application scenario in smart vehicle space. Also, we propose a prototype system of our model.

1 Introduction

A new computing model is coming into our life, which we called "ubiquitous computing" [1]. Environment, people, and smart devices are three elements of this novel computing model. In order to relate with each other harmoniously and naturally, we should synthesize multifarious techniques. As a result, ubiquitous computing integrates many research areas including software architecture, middleware platform, languages design, distributed systems, artificial intelligence, vision recognition, user interfaces, and biological authentication. Recently, many related academic research efforts and commercial reality are made for ubiquitous computing. As a whole, they focus on active and smart spaces such as intelligent home, easy meeting-room, and smart museum. In the smart spaces, in terms of changes of the people, environment and devices, this computing model automatically and continuously self-adjusts according to pre-defined strategies to new states in order to provide better cooperation and communication between entities. Achievement of this attractive goal poses a large number of new challenges for software architecture and middleware technology. The traditional computing models are no longer suitable [2]. Therefore, a novel middleware architecture is needed to support ubiquitous computing. We consider the keystone to be adaptation, which we use a semantic context model to realize. Context-aware mechanisms provide an infrastructure for adaptation. Since there are many different run-time environments, in order to achieve greatly improved cooperation,

they should understand and communicate with each other better than they do today. Thus we define common essential ontology for smart space. On the other hand, to deal with the specific application domains, we first find problem domains, and then build ontology of specific domains. Moreover, according to the context complexity, we deal with context according to hierarchy and priority. Emphatically, we argue that "semantic view" for context is the key enabler of smart space.

Vehicles have merged into our daily life, playing an important role. Because we need more comfort, facility and safety in vehicles, we select vehicle space as a representative scene of ubiquitous computing. In this paper, we propose a semantic context model integrating semantic web technology and ontology language for self-adaptation. Our work is related to other pervasive and context aware computing research such as CoBrA [3], Context Toolkit [4], One.World [5], TOTA [6]. Compared with the previous systems, our design focuses are twofold. Firstly we focus on smart vehicle space; secondly we emphasize the semantic view of a context-aware middleware model in smart vehicle space and we propose a semantic context model. Because context-driven and person-centric are the characteristics of ubiquitous computing, we consider the semantic context model is very important. We have defined the common essential ontology and application domain-specific ontology in smart vehicle space.

The structure of the paper is as follows. Section 2 presents the the semantic and adaptive context framework, which comprises smart vehicle space overview, semantic view for context and the formal specification of our context model. Section 3 introduces an application domain-based ontology in smart vehicle space. Next, section 4 presents an application scenario in smart vehicle space and a prototype system of our model. Finally, section 5 summarizes the discussion.

2   Semantic and Adaptive Context Framework

As Tim Berners-Lee [7] described, semantic web is an extension of the current web where information is given well-defined meaning, better enabling computers and people to cooperate better. Below, we present smart vehicle space and semantic view for context. In addition, we propose a semantic and adaptive context model.

2.1   Smart Vehicle Space

Smart vehicle space is an essential research field of smart spaces. Using security authentication, image processing and pattern recognition technology, in terms of relationships between people, vehicle and environments, the grade of security and comfort is greatly improved.

Smart vehicle space has four parts, which is defined as SVS=(CA, CR, AC, CP), where CA is a context acquisition system; CR is a context repository reasoning system; AC is an auto controlling system and CP is a centralized processing system. CA is defined as CA=(($\triangle$statpe,$\triangle$statdv,$\triangle$staten), (sen, cam,

soundrec)). CA comprises sensors, cameras and sound receivers, which aims at sensing the status change of people, devices and environments in the vehicle. CR is defined as CR=(context, ontology, domain, inference), and uses correlative contexts and application domain-based ontologies to make the manipulating strategy for adaptation. AC is defined as AC=(ste, com, ent, nav, sec), which includes steering, communication, entertainment, navigation and security subsystem. Particularly, CP is the kernel of smart vehicle space, which controls above third parts co-operating effectively and adaptively. We define CP as CP=f(CA, CR, AC), where f is a control function.

2.2   Semantic View for Context

Context [8] is any information that can be used to characterize the situation of an entity. An entity may be a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves. Commonly used contexts consist of location, identity, time, temperature and activity. We consider specific objects in environments are all context. The semantic information of context is essential for dealing with complex tasks in ubiquitous computing environments.

Context-aware is an ability to sense and use different context. Any application that takes advantage of context is a context-aware application. Context-aware computing is the ability of computing devices to detect, interpret and respond to the change of environment and system. The W3C organization has specified a language OWL that is based on DAML and OIL for semantic web, which is more expressive than RDF and RDF-S. Because context is so complicated and hard to understand and use, we have introduced semantic web technology into the context framework in smart space. We share common vocabularies and use OWL as our standard syntax to present information and inference. In this way, we model the different context hierarchies more effectively and adaptively.

2.3   Semantic and Adaptive Context Model

In this part, we give a detailed formal specification for the semantic and adaptive context model, including three definitions and one algorithm.

Definition 1 Context model Context model CON=(S, P, V) is a 3-tuple, where S=$\{s_1, s_2, ..., s_n\}$ is a set of context semantic information; P=$\{p_1, p_2, ..., p_n\}$ is a set of attributes of S; and V=$\{v1, v_2, ..., v_n\}$ is set of values of S.

Definition 2 Semantic Context model Semantic Context model MSC=(K, T, S) is a 3-tuple, where K is context-driven kernel including basic context services; T is a set of context base and process tools; T is defined as T=(B, O), where B=(SB, TB) is a set of context bases comprising context shared base and transcendent base; O=(SR, CF) is a set of context tools such as smart reasoning and context fusion algorithm. S, defined as S=$(St_{pe}, St_{en}, St_{de})$, is a set of states of people, environments and smart devices.

Definition 3 Semantic and adaptive context-aware processing model
Semantic and adaptive context-aware processing model SACM=(O, A, S, G, N, T), where O is the application domain-based ontology; A is a process of context acquisition, which aims to gain raw data from sensors and transform it to ontology entities or attributes. The data they export are of different type and structure, which is abstracted so that it can be used by an application; S is a process of context storage, which stores the acquired context in a repository for access when necessary; G is a process of context aggregation, which integrates correlative context for a specific entity; N is a process of context analysis, which infers current status of entities or the intention of users; T is a process of context-aware actions. We specify appropriate action rules for a number of context scenarios. At run-time, whenever the system state matches a particular scenario, the associated action will be automatically executed according to pre-defined strategies. Emphatically, T process is not serial and static, because of the execution of actions will cause the system to enter a new state, so bringing the system to a new context scenario. In addition, there may be multiple processes in progress interacting with each other. As a result, T is a stochastic and recursive process.

Algorithm SAC To elaborate the semantic and adaptive context process, we present the SAC algorithm.

Algorithm SAC(msc, sacm, △con): According to semantic of the context-aware tasks, it implements the adaptive context process.

```
Input:
  Semantic context set msc (a subset of MSC);
  Semantic and adaptive context-aware processing sacm (a subset of
  SACM);
  A set of change of context ccon;
Output:
  A new semantic context set msc' (a subset of MSC);
Begin
  Decompose ccon into several basic atom-context sets (atcs);
  Foreach atc in atcs do
    Do sacm.a according to sacm.o, msc.s;
    Do sacm.s, sacm.g, sacm.n according to msc.b.sb and msc.b.tb
    Until sacm.t;
    Update(msc.b.sb);
    Update(msc.b.tb);
  Endfor
End.
```

3   Application domain-based Ontology

An ontology [9] is an agreement about shared conceptualization, which includes conceptual frameworks for modelling application domain knowledge, content-

specific protocols for communication among interacting agents, and agreements about the representation of specific domain theories. Ontology has such characteristics as definitions of representational vocabulary, a well-defined syntax, an easily understood semantics, efficient reasoning supports, sufficient expressive capabilities, and convenience of expressions. We understand ontology at two different levels. (a) It is a vocabulary, which uses appropriate terms to describe entities and relationships between entities. (b) It is a knowledge base for a specific domain. In our view, ontology describes domain knowledge in a general way and provides consistent understanding of one application domain.

3.1   Common Ontology

For smart spaces, we have built some common essential ontology EO=(SS, PL, TI, PI, DC, SP), where SS is a set of smart space characteristics, which describes names, types, important locations and devices; PL is a set of physical location of person and devices; TL is a set of time, presenting the period of time or instant time; PI is a set of person identities, on which the system confirms one person different to others; DC is a set of devices characteristics, including names, types and attributes; SP is a set of security and privacy policy, denoting a person's capabilities.

3.2   Specific Ontology

Because smart vehicle space is a specific environment, we examine its characteristics from the view of context. (a) Space -inside vehicle- is relatively confined. The devices inside the vehicle may have limited capabilities and be fixed to the vehicle. Also, the space for user to move is restricted. (b) Though environments -outside of vehicle- continually change, we need not consider the whole range of dynamics, just those can influence our system.

The above characteristics determine the definitions of context in smart vehicle space. We use the idea of ontology to describe the context information. The context in smart vehicle space is defined as CONsvs=(VC, EC, DC), where VC is a set of vehicle context, concerning the statuses and attributes of devices inside the vehicle, such as air-condition, wiper, light, engine, ABS, and seat; EC is a set of environment context, comprising the environmental elements which may influence driving, such as weather, road status, fingerposts, and signal lamps; DC is a set of driver context, including (a) the status of a driver, such as the suitability and ability to driving; (b) physiological parameters, such as alcohol levels and pupil diameter.

We have used Protégé [10] tool to build context and create instances in smart vehicle space. Protégé is an ontology editor tool, providing a GUI for the user to create and manage the ontology architecture. Using Protégé, we export files in OWL format for inference. Figure 1 shows the ontology of smart vehicle space.
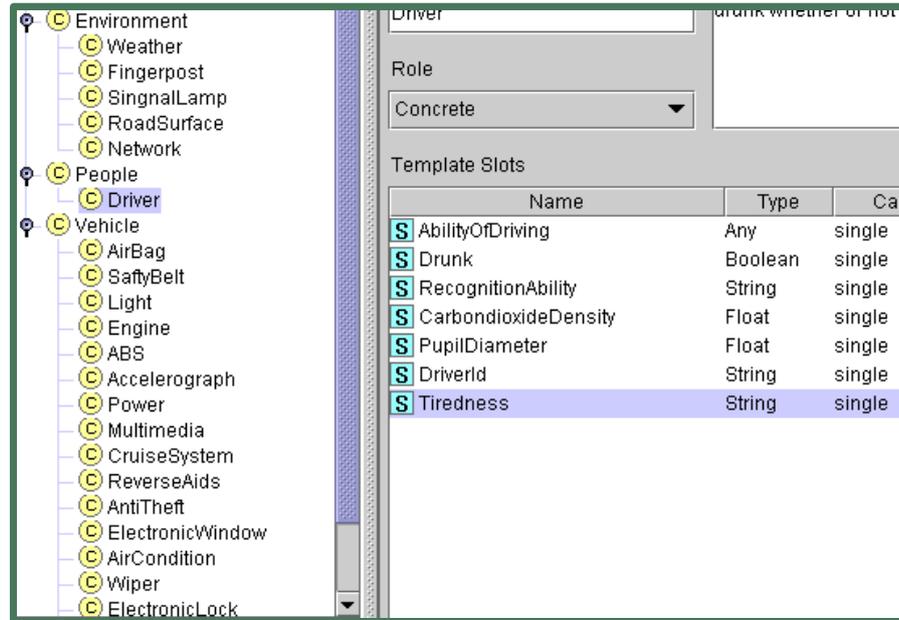
Fig. 1. Ontology Built in Protégé

3.3   Ontology Usage

We have developed an ontology repository and defined three base classes, which represent the context inside the smart vehicle, the outside environment of the vehicle, and the driver respectively. In order to be able to rely on the knowledge base, we filter the context scenario and specify fixed conditions(context) that trigger specific actions. Once the system meets the condition we have defined, the context reasoning system will be triggered and will perform the associated actions.

The context scenarios that we are interested in comprise three parts. We define IS=(SD, OI, DR), where SD is a set of security driving, including the scenario of driving at high speed in case of emergency; OI is a set of influences of outside environment, such as the vehicle can follow the signpost and turn to the right , or the ABS will engage in the case of loss of traction; DR is a set of the driver status. Importantly, we specify several driver conditions. Each one corresponds to different physiologic parameters. According to different danger level, system takes various actions. For example, if the driver is unable to continue driving when the danger level is high, the system will force a controlled stop and call for help. If the condition is not very serious, the system may simply park at the nearest convenient location. As a result, in terms of the ontology, the system can deal with different problems adaptively.

4   One Scenario

To demonstrate the application of the semantic and adaptive context model, we present the following scenario. It is time for Mr. Wu to go to work. He approaches his vehicle and puts his palm on the lock authentication machine. Next, his fingerprint information is sent to the in-vehicle computer that receives the data, analyzes them, confirms him as a legal driver, and then orders the door to open. After the door opens, he gets into the vehicle, sits down, and puts his ID card on the ID machine. At the same time, a sensor measures his weight and a camera records his appearance. These data are sent to the in-vehicle computer, which recognizes his identity. If allowed, the system sends a welcome command to the audio device in the entertainment system. A voice "Welcome, Mr. Wu" comes from the speaker. At the same time, the in-vehicle computer orders the entertainment system to play his favourite music and the climate system to adjust the air quality according to his preferences. Mr. Wu inputs his destination into the in-vehicle computer. The optimal router and some alternative routers are given. He selects the optimal one and the in-vehicle computer accepts his selection. He may say: "start". His command is received by voice sensor and sends it to the in-vehicle computer. Then the in-vehicle computer sends a command to the control system and the motor starts. The vehicle monitoring system detects the vehicle's status is good, and reports the status to the in-vehicle computer. If the control system detects water in the road, it will send a message to the in-vehicle computer. Therefore, Mr. Wu is warned to pay attention to the road condition and the control system sets the vehicle parameters to avoid skids. On arrival at the destination, the in-vehicle computer stops the motor and opens the door. Mr. Wu gets out. The in-vehicle computer orders the door to close after his departure is detected by the lock sensor.

According to the above scenario, we have developed a prototype system for smart vehicle space based on the semantic and adaptive context model. Figure 2 shows one screen shot of the system. According to our context model, we have defined an ontology related to smart vehicle space. With this prototype system, we can capture the required information about drivers, environments and vehicles. The system adapts to different conditions automatically.

5   Conclusions

In this paper, we have proposed a semantic and adaptive context model for smart vehicle space to explore ubiquitous computing. We focus on synchronization and adaptability aspects of semantic context, using OWL to build an ontology of smart vehicle space. Further, we argue that it is crucial to focus on the ontology-based context-aware aspect of interaction and communication. Moreover, from a practical point of view, we give a scenario in smart vehicle space and present a prototype system of our context model.
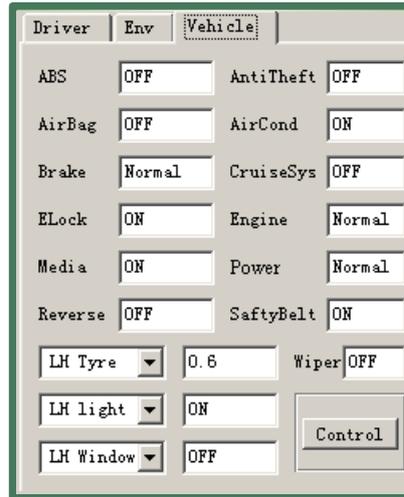
Fig. 2. A Screen Shot of the Prototype System

References

1. Weiser M: The Computer for the 21st Century. Scientific American, pp.94-100 (1991)
2. Anand Tripathi: Next-Generation Middleware Systems Challenges Designing. Communications of the ACM, 45(6), pp. 39-42 (2002)
3. Harry Chen, Tim Finin and Anupam Joshi: A Context Broker for Building Smart Meeting Rooms. American Association for Artificail Intelligence (2004)
4. Daniel Salber, Anind K. Dey and Gregory D. Abowd: The Context Toolkit : Aiding the Development of Context-Enabled Applications. Proceedings of CHI'99. ACM Press (1999)
5. Robert Grimm, Janet Davis, Eric Lemar, Adam MacBeth, Steven Swanson, Thomas Anderson, Brian Bershad, Gaetano Borriello, Steven Gribble, and David Wetherall: System support for pervasive applications. ACM Transactions on Computer Systems, 22(4), pp. 421-486 (2004)
6. Marco Mamei and Franco Zambonelli: Programming Pervasive and Mobile Computing Applications with the TOTA Middleware. In proceedings of the 2nd IEEE International Conference on Pervasive Computing and Communications, (2004)
7. Tim Berners-Lee and Mark Fischetti: Weaving the web: The original design and ultimate destiny of the world wide web by its inventor. (2001)
8. Anind K. Dey: Providing Architectural Support for Building Context-Aware Applications. PhD thesis, Georgia Institute of Technology (2000)
9. B. Chandrasekaran, John. R Josephson, and Richard V. Benjamins: What Are Ontologies, and Why do We Need Them? IEEE Transactions on Intelligent Systems, pp. 20C26 (1999)
10. Natalya, F. N., Michael, S., Stefan, D., Monica, C., Ray, W. F., Mark, A. M. : Creating Semantic Web Contents with Protege-2000. IEEE Intelligent Systems, Vol. 16, No. 2. IEEE Computer Society, pp. 60-71 (2001)