A MOBILE DEVICE ORIENTED FRAMEWORK FOR CONTEXT INFORMATION MANAGEMENT

Yongkai Zhan [1], Shuangquan Wang [2], Zhuang Zhao [1], Canfeng Chen [2], Jian Ma [2]*

[1] College of Computing & Communication Engineering, Graduate University of the Chinese Academy of Science, Beijing, 100049, China
[2] Nokia Research Center, Beijing, 100176, China
{zhanyongkai07, zhaozh}@mails.gucas.ac.cn
{ext-shuangquan.wang, canfeng-david.chen, jian.j.ma}@nokia.com
** Corresponding Author*

ABSTRACT

Recent developments of information technologies are leading the advent of mobile device oriented context-aware applications，which means using context to realize some novel applications on mobile devices. The context-aware application needs a framework for context information management. In this paper, we propose a mobile device oriented framework for context information management to solve the problems on context shortage and communication inefficiency. It consists of four main functional components: data collector, context processor, context manager and local context consumer. By using the framework, data collector collects data from internal and external sensors, then the context processor and manager extract and manage context, finally the local context consumer develops context-aware applications and provides appropriate services to the mobile users. The propose framework realizes real time application with quick data access, power efficiency, personal privacy protection, data fusion of internal and external sensors and simplicity in usage. In order to show the feasibility of the proposed system, an example scenario is given, which is applied to a context-diary system.

Index Terms—Framework, Context Awareness, Mobile Services, Ontology Model

1. INTRODUCTION

With the progress of electronic technology and communication technology, the mobile phone, which is used universally as an effective equipment for communication, has a high-speed processor and a large memory which could apply to some complex applications. It can also obtain different kinds of sensor data from the sensors internal or external of the mobile phone. Thus we can extract user-centric context information and provide context-aware services to mobile users.

The current research spots are concentrating in how to manage the context information. However, most existing context oriented systems are designed for PC, only a few are designed for mobile phone. Two typical context-aware systems designed for mobile phone are the Kontti [1] and the Mobile Life [2]. The Kontti obtains raw data from the mobile's internal sensors while the Mobile Life obtains the data from both internal and external sensors, then they all extract contexts from those raw data. But they manage the context information in a centralized way, all the extracted context information stores on the backend server for applying context-aware services, which may lead to power consumption, time consumption and privacy leakage.

In this paper, we propose a mobile device oriented framework for context information management. It collects and fuses the internal and external data, and then extracts context from the fused data in a pre-defined way. In addition, we use a "publish and subscribe" mechanism [3] to inform the client the change of context but no data is sent to the client, which makes radically decreasing of the message traffic up to the application. Besides, the proposed framework manages all the extracted context information locally and provides context-aware services to the mobile user. Thus, the context information is managed in a decentralized way, which can reduce the energy consumption and time consumption for transferring data to the backend server and protect the user's privacy.

The remaining of this paper is organized as following. In Section 2, we propose the mobile device oriented framework for context information management and introduce each component in detail. Section 3 shows an example scenario based on the proposed framework. Section 4 concludes this paper and shows the future work.

2. A MOBILE DEVICE ORIENTED FRAMEWORK FOR CONTEXT INFORMATION MANAGEMENT

978-1-4244-5076-3/09/$26.00 ©2009 IEEE

To manage mobile device oriented context information effectively, the main tasks which the proposed framework should been fulfilled are [4, 5]:

- Concurrent context management in mobile device
- Collect data from the internal and external sensors
- Support data fusion
- Extract the data for context information
- Use ontology to standardize context
- Store the context based on the ontology model
- Support reasoning for high-level context
- Change detection
- Event-based communication of context to application

In order to achieve these goals under the restriction of mobile phone's storage and operational capability, we design a mobile device oriented framework for context information management, as shown in Figure 1. It includes four main functional components which all located on the mobile phone: 1) data collector, 2) context processor, 3) context manager and 4) local context consumer. In addition, the "publish and subscribe" mechanism is used in the framework. In the following part of this paper, we'll descript each component in detail.

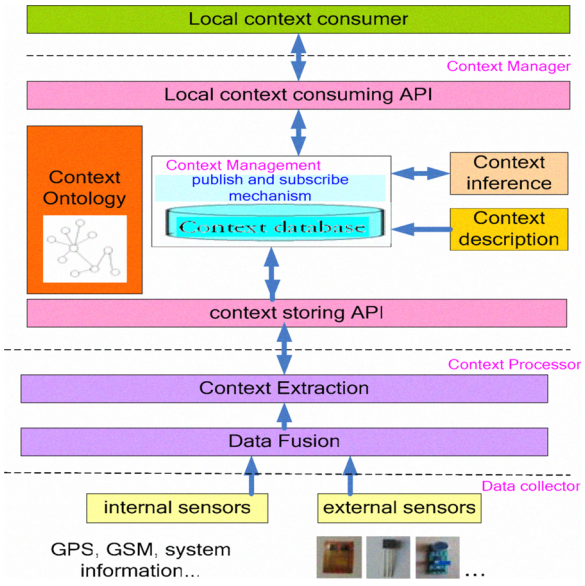

Figure 1: A mobile device oriented framework for context information management

2.1 Data collector

There are two kinds of sensors for mobile: the internal and external sensors. The internal ones which are embedded in the mobile phone can be divided into physical sensors (such as GPS, GSM, Bluetooth devices, accelerometer and WLAN) and virtual sensors (such as system information, call and SMS). The external ones which are placed outside the mobile phone sense the physical phenomena around (such as light, sound, temperature, pressure, acceleration, infrared Rays and ultraviolet Rays).

What the data collector dose is to collect the data from the internal and external sensors. The internal sensor date is collected through the application programming interfaces (APIs) provided by the mobile phone's operation system. The external sensor data is obtained through WLAN, Bluetooth or ZigBee. Universal sensor data (uSD) entry card, as shown in Figure 2, which contains a ZigBee transceiver, can communicate with the sensor node directly [6].

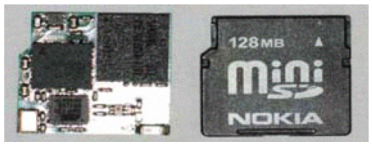

Figure 2: The universal sensor data (uSD) entry card (left) and the standard SD memory card (right)

2.2 Data Processor

Data Processor fuses the collected data and extracts the context information from the fused data.

Data fusion supplies more valuable and more accurate data. For example, the location data could be got from the internal sensors (GPS and GSM) and the external sensors (Beacon). GPS data is more accurate than GSM data and the Beacon data describes the precise position in a small area. Thus the data fusion provides the precise position data from nation, province to the precise position in a small area.

The fused data could not be used directly. For example, the pressure data: 0 or 1 doesn't have practical usage. So we must extract context from the raw data. Context is "any information that can be used to characterize the situation of entities (i.e., whether a person, place, or object) that are considered relevant to the interaction between user and an application, including the user and the application themselves."[7] Context extraction supplies the extracted context based on the fused data, such as the user's locations, the profile of the mobile phone, etc. Context extraction is task oriented. For different applications, the context information maybe extracted differently. The system designer should define all the necessary context information to extract beforehand. Normally, low-level context information can be extracted directly from the sensor data. Middle-level context information and high-level context information can be obtained from the processing and analysis of the sensor data. Besides, some context information can be inferred by using context inference which we introduce below.

2.3 Context Manager

Context manager provides APIs for context storing and local consuming, stores the context in the context database based on the description of the ontology model, deduces

high-level context and provides a 'publish and subscribe mechanism'. It divides into context ontology and context management.

2.3.1 *Context ontology*

To manage the context on mobile phone effectively, an ontology model should be set up. The ontology is "A specification of a representational vocabulary for a shared domain of discourse — definitions of classes, relations, functions, and other objects". [8] Through it the context can be managed more effective. The two typical languages describing the ontology model are the Resource Description Framework (RDF) and the Web Ontology Language (OWL). The RDF is "a foundation for processing metadata" [9]. OWL is a language "builds on RDF and RDF Schema and adds more vocabulary for describing properties and classes" [10]. The difference is that, OWL with more powerful reasoning capability does reasoning easily, however it is more complex. Considering the restriction of mobile phone's storage and operational capability; RDF is used in the proposed framework to define the user-centric ontology.

The ontology includes some component ontology. The mobile phone can communicate through WLAN, Infrared Rays net, GSM and Blue tooth, so the network description ontology is designed as Figure 3. The contact ontology shows the contact relation for mobile user contact with person or object in Figure 4. The mobile phone's location is changing. As most user pays close attention to his\her location, the location description ontology (Figure 5) supplies the location of the user which described by the GPS, Address and other way. The environment ontology (Figure 6) describes the environment which gets information through the internal sensors and the external sensors around the mobile user and shows the weather forecast. User, Contact Profile, Network description, Device and Sensor description ontology are also described in the ontology model.

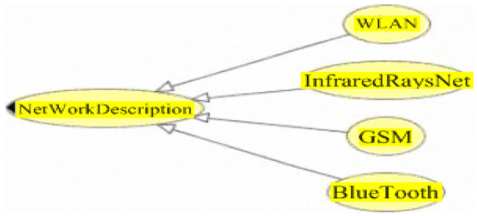

Figure 3: The Network Description Ontology

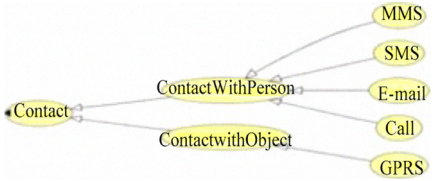

Figure 4: The Contact Ontology

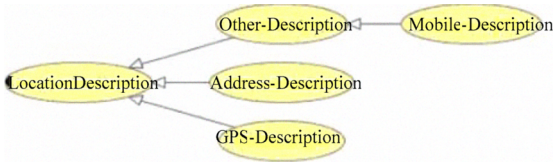

Figure 5: The Location Description Ontology

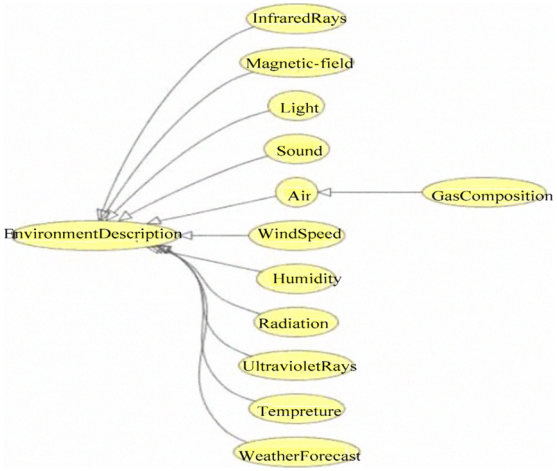

Figure 6: The Environment Ontology

2.3.2 *Context management and inference*

The Context management stores the context in the context database based on the description of the context ontology model; uses the stored context doing inference for the high-level context and adds new context in the context database; provides a 'publish and subscribe mechanism' which is introduced in the blackboard architecture model [3]. The mechanism informs the emergence of context or change of context or end of context but sends no data to the client.

Through storing and processing context locally, the proposed framework realizes real time application with quick data access; furthermore since it transfers no data to the server, the proposed framework's advantages in context manager are: 1) real time application with quick data access, 2) power efficiency and 3) personal privacy protection. Through the 'publish and subscribe mechanism', the proposed framework also makes radically decreasing of the message traffic up to the application

Context inference processes the low-level context for high-level context, and stores the high-level context in the context management. Relative to the low-level context, the high-level context is the context which produced by reasoning through one or more low-level context types. For example, we can use the GPS location context to inference which vehicle the user uses, when for a long time the GPS location context is null, and from beginning time to end time it changes, and both the beginning location and the end location are near the subway station, then we can inference that the user took subway at that time.

2.4 Local context consumer

Based on the available context information stored in the context database, the context-aware applications can be developed and the appropriate services can be provided to the mobile users in local context consumer. The context-aware applications and the appropriate services can be activated automatically or manually.

In order to use the automatic applications, the emergence of context or change of context or end of context can be informed to local context consumer first through the 'publish and subscribe mechanism'. Then the automatic context-aware applications and appropriate services may commence. For example, the local context consumer informs the change of environmental light intensity, and then the context-aware applications which could turn on or off the mobile's background light may active automatically.

User uses the manual context-aware applications to see the context or do customized applications. For example, user uses the manual applications through searching suitable stored context to see the situation around.

3. EXAMPLE SCENARIO

In order to show the feasibility of the proposed framework, an application example is shown in this part. Diary records one's daily events, appointments, observations, etc. The context-diary system using the proposed framework supplies more abundant content than normal diary.

The context-diary gets data from internal and external sensors, fuses the data such as the location data, and then extracts the needed context whose type is pre-defined, stores the extracted context in a database based on the description of the context ontology model, uses the stored context to do inference for the high-level context and adds new context to the context database. Then the context-diary applications and the appropriate services can be activated automatically or manually. The automatic applications create a file named of the date which is used to record the user's information in multiple ways. User uses the manual applications to see the context in the database. Through the internal and external sensors, the system can get more abundant content. It shows the user's daily route, the environment around, the contact, the photo and video with the location, time and environment information, etc.

4. CONCLUSION

In this paper, we propose a mobile device oriented framework for context information management and introduce each component in detail. It collects and fuses the internal and external context, manages extracted user-centric context information in a centralized way, then provides context-aware services for the mobile user. To show the feasibility of the proposed system, we presented an example of context-diary system by using the proposed framework.

Compared with other existing frameworks, the proposed framework has the following advantages: 1) real time application with quick data access, 2) power efficiency, 3) personal privacy protection, 4) fused data of internal and external,5) simplicity in usage.

5. ACKNOWLEDGMENT

This work was supported in part by the Knowledge Innovation Program of the Chinese Academy of Sciences under Grant no. KGCX2-YW-120, NSFC Grant no. 60702074, and open fund of Key Laboratory of Wireless Sensor Network and Communications, Chinese Academy of Sciences. The conference participation is supported by Nokia bridging the world program.

6. REFERENCES

[1] Juha Kolari, Timo Laakko, *et. al.*, "Context-Aware Services for Mobile Users," VTT, Finland, 2004.

[2] Patrik. Floréen, Matthias. Wagner, *et. al.*, IST-2004-511607 MobiLife, D32 (D4.6) Version 1.0, 2006

[3] Panu. Korpipää, "Blackboard-based software framework and tool for mobile device context awareness," VTT, Finland, 2005.

[4] Dey. A, *Providing architectural support for building context-aware applications*, Ph.D. dissertation, Georgia Institute of Technology, pp. 170 ,2000.

[5] Mitchell. K, *Supporting the development of mobile context-aware Systems*, Ph.D dissertation, Lancaster University, pp. 224, 2002

[6] X. Zhang, X. Wang, Y. Ren, C. Chen and J. Ma, "A Novel Compatible Hardware Expansion Method Based on General Memory Interface", in Proceeding of 2009 International Conference on Communications and Mobile Computing, Kunming, China, 2009.

[7]Dey, A. & Abowd, G. 2000. "Towards a better understanding of context and context awareness." Proceedings of What, Who Where, When and How of Context-Awareness Workshop in CHI2000 ACM Conference on human factors in computer systems.

[8]: Lars Marius Garshol (2004). "Metadata? Thesauri? Taxonomies? Topic Maps! Making sense of it," www.ontopia.net. 13 October 2008.

[9]: Resource Description Framework (RDF) Schema Specification. http://www.w3.org/TR/2000/CR-rdf-schema-20000327/

[10]: Web Ontology Language (OWL). http://www.w3.org/2004/OWL/