

TOWARDS A MODEL OF CONTEXT-AWARE INFRASTRUCTURE IN MOBILE UBIQUITOUS COMPUTING

M. Beggas

Département Informatique, Centre Universitaire d'Eloued, Algérie

Received: 12 February 2011 / Accepted: 02 June 2010 / Published online: 30 June 2011

ABSTRACT

The aim of this paper is to propose the main components of a system that provide a model for context aware architecture in mobile ubiquitous computing as a layered architecture. The modeled system provide services should be adapted according to context for end user application, that can be executed in a mobile device as mobile web or mobile application

Keywords: Context, context awareness, ontology, ubiquitous computing, mobile computing

1. NTRODUCTION

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application [1]. Contexts can include information such as information about physical world (temperature, humidity, etc), location (e.g., of people or objects), time, execution state of applications, computational resources, network bandwidth, activity, user intentions and user emotions.

Author Correspondence, e-mail: mbeggas@gmail.com

[ICID: 1020777](#)

Context can be defined by specifying three categories of context [2]: (i) computing context: for example, network connectivity, communication bandwidth, nearby resources like printers, displays; (ii) user context: for example, user's profile, location, emotional state, people nearby, current activity; (iii) physical context: for example, lighting, noise level, traffic conditions, temperature.

The goal of context-aware applications is to respond to context changes to enhance the computing environment for the user by: presentation of information and services to a user; automatic execution of a service for a user; tagging of context to information to support later retrieval or adaptation of application's behavior and appearance.

Context information is captured by sensors, with sensor not only referring to sensing hardware, but to every source that can provide an application with context information. Sensors can be [2]:

Physical sensors are hardware sensors. Virtual sensors retrieve their context data from software applications or services. Logical sensors combine data from several sensors to gain inferred context information. For example, such a sensor can infer from the user's location.

2. ELEMENTS OF CONTEXT AWARE SYSTEMS

A context-aware system can be viewed as having three basic functionalities [3]: sensing, thinking, and acting.

- 1- Sensing: Sensors are used to acquire data or information about the physical world or some aspects of the physical world.
- 2- Thinking: Reasoning is employed to infer more knowledge from information perceived via the senses,.
- 3- Acting: Once context information has been gathered or situation recognized, actions are taken.

3. REFERENCE MODEL FOR CONTEXT AWARE ARCHITECTURE IN MOBILE UBIQUITOUS COMPUTING

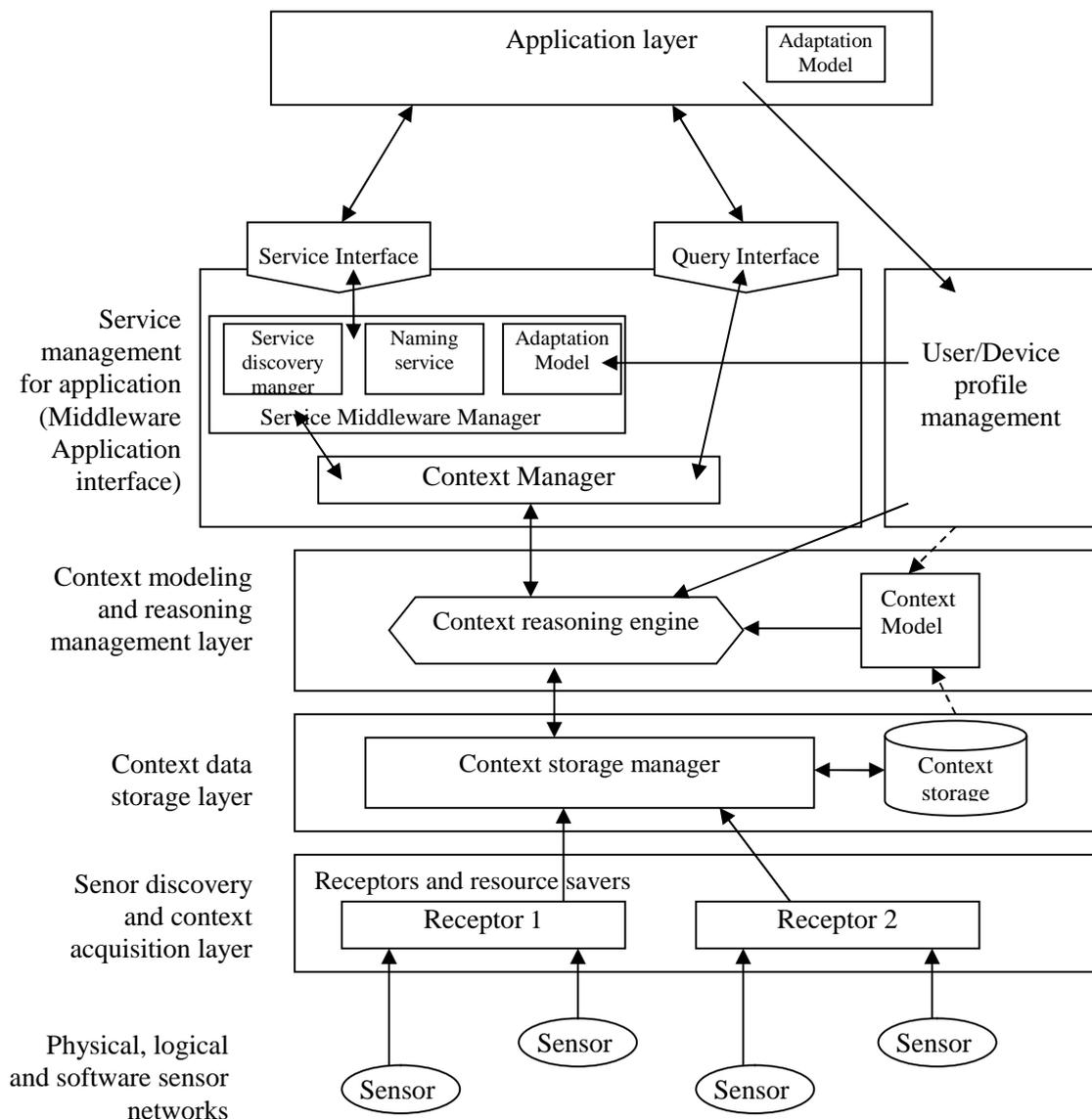


Fig.1. Reference model for context aware architecture in mobile ubiquitous computing

Context aware infrastructure provides to application high level context information. End user application don't carry in how discover and communicate with low level sensors. Context aware infrastructure provides context-aware services by applications. Services should be adapted to physical environment, user, and terminal contexts.

Context aware infrastructure assures the following functionalities:

- Context acquisition from sensors
- Context storage and management of raw context data
- Context modeling and reasoning
- User and environment profile management
- Service management for the end user application, middleware interface layer

Figure 1 illustrates a reference model defined as multilayered architecture, adapted from [4] [5] [3].

3.1. Context acquisition

A wide selection of sensors and sensing technologies can be applied to collect contextual information. Some of the sensing technologies can be further categorized into: Temperature and humidity, Vision and light, Location, orientation, and presence, Etc.

To be context-aware, it has to be further able to communicate with context sources. Hence, we arrive to the second obstacle: context sources are highly heterogeneous. As with the case of mobility, Heterogeneity appears in two senses, architectural and behavioral. Architectural heterogeneity refers to the different device types, sources. Behavioral heterogeneity relates to the different interfaces provided by context sources. The hardware sensor interface should be able to communicate with heterogeneous sensors that have different communication protocols. For example, ZigBee/802.15.4, UPnP, Salutation, SLP and JINI (for software interfaces).

We can use LivePlatform developed in LIMOS1 laboratory that contain hardware and software platform (sensor network, communication protocols and real time core).

3.2. Context Storage

The context storage can provide a context history for later retrieval. Moreover, it can be used to establish trends, predict future context values and identify interrelations between context data. A context-based storage system consists of a logical context data model and a physical data storage space. This repository could be layered on top of a database management system. [5].

3.3. Context modeling

Context reasoning infers high-level contexts from basic sensed contexts, resolves context conflicts, and maintains knowledge base consistency. To reason on concepts, it is necessary to have a robust model.

¹ LIMOS, Laboratoire d Informatique, de Modélisation et d'Optimisation des Systèmes, www.isima.fr/limos/

[6] describe different models for context modeling: Key-Value, Markup Scheme Models, Object Oriented Models, Logic Based Models and Ontology Based Models. This later is the most suitable and used model.

Ontologies are a promising instrument to specify concepts and interrelations. They are particularly suitable to project parts of the information describing and being used in our daily life onto a data structure utilizable by computers. Using ontologies provides a uniform way for specifying the model's core concepts as well as an arbitrary amount of subconcepts and facts, altogether enabling contextual knowledge sharing and reuse in an ubiquitous computing system. This contextual knowledge is evaluated using ontology reasoners. The model has been implemented applying selected ontology languages. These implementations build up the core of a non monolithic Context Ontology Language (CoOL), which is supplemented by integration elements such as scheme extensions for Web Services and others. Beyond determination of service interoperability in terms of contextual compatibility and substitutability, this language is used to support context-awareness in distributed service frameworks for various applications.

The CONON context modeling approach by Wang et al. [6] is based on the same idea of the CoOL approach, namely to develop a context model based on ontologies because of its knowledge sharing, logic inferencing and knowledge reuse capabilities. Wang et al. created an upper ontology which captures general features of basic contextual entities and a collection of domain specific ontologies and their features in each subdomain. The CANON ontologies are serialized in OWL-DL which has a semantic equivalence to well researched description logics. This allows for consistency checking and contextual reasoning using inference engines developed for description languages. A promising emerging context modeling approach based on ontologies is the CoBrA system. This system provides a set of ontological concepts to characterize entities such as persons, places or several other kinds of objects within their contexts. The CoBrA system uses a broker-centric agent architecture to provide runtime support for context-aware systems, particularly in Intelligent Meeting Rooms, a prevalent scenario of an ubiquitous computing environment.

3.4. Context reasoning

We consider context reasoning as a means to deduce new and relevant information to an application or user from the various sources of context data.

Context reasoning can provide additional information that does not exist in the context storage, but care must be taken to maintain data consistency before adding such information to the context storage, as the underlying raw context information may change continually.

3.5. User, terminal and environment profile

Context awareness is emerging as an essential feature for the next generation of mobile services.

The context defining a mobile service request as being described by a set of parameter values possibly belonging to different profiles. A profile is intended as a structured set of parameters describing an entity. Most common examples of profiles are user profiles, device profiles and profile parameters of the provisioning environment. . The first type usually contains data about user preferences, interests, and demographics,. The second type usually contains technical data describing device capabilities such as installed memory, screen resolution, computing power, available user interfaces, and installed software, and the battery level.

The profile parameters of the provisioning environment include the availability, type, and status of the network connection between the user and the service provider.

The acquisition from different sources of all the profile parameters defines the context of a service request and their aggregation into a consistent uniform description. The distribution of profile sources imposes two main requirements: (1) a common formalism and a shared vocabulary to be used by the different profile sources to represent the data, and (2) a mechanism to deal with possibly conflicting parameter values provided by different sources.

3.5.1. Profile Representation of Devices.

HTTP Headers. The device that makes the request involves identifying the device by means of HTTP request headers. It is worth noting that this technique is applicable only

to HTTP-based services. the information conveyed by HTTP/1.1 headers that can be useful for representing device capabilities is quite limited and includes only the user agent (i.e., browser) and media types (MIME types) accepted by the user agent, screen size and resolution, and color capabilities.

Composite Capability / Preference Profiles and UAProf. World Wide Web Consortium (W3C) defined the structure and vocabularies of Composite Capability/Preference Profiles (CC/PP) [7]. CC/PP uses the XML serialization of RDF graphs to create profiles that describe the capabilities of the device and, possibly, the preferences of the user. CC/PP profiles are structured as sets of components that contain various attributes with associated values. Components and attributes are defined in CC/PP vocabularies (i.e., RDF schemas). Data-type support in CC/PP is quite limited; in fact, attribute values can be either simple (string, integer, or rational number) or complex (set or sequence of values, represented as `rdf:Bag` and `rdf:Seq`, respectively). Currently, CC/PP is used primarily for representing device capabilities and network characteristics.

UAProf [8] has been proposed by the Open Mobile Alliance for representing the hardware, software, and network capabilities of mobile devices. In particular, UAProf defines seven components:

- `HardwarePlatform` provides a detailed description of the hardware capabilities of the terminal, including input/output capabilities, CPU, memory, battery status, and available expansion slots.
- `SoftwarePlatform` describes the device operating system, its Java support, supported video and audio encoders, and the user's preferred language.
- `BrowserUA` describes in detail the browser features, providing not only the browser name and version but also information regarding its support for applets, JavaScript, voiceXML, and text-to-speech and speech-recognition capabilities, as well as the user's preference regarding frames.
- `NetworkCharacteristics` provides information about the network capabilities and environment (such as the supported Bluetooth version), support of security protocols, and the current bearer signal strength and bit rate.
- `WAPCharacteristics` contains a set of attributes regarding the device Wireless Application Protocol (WAP) capabilities, including the supported WAP, Wireless

Markup Language (WML), and WMLScript versions, as well as the WML deck size.

- PushCharacteristics and MMSCharacteristics provide information regarding the device WAP push capabilities and multimedia messaging service (MMS) support, respectively.

Many hardware vendors make publicly available on their Web sites the UAProf profiles of their devices;

3.5.2. User Profiling. User profile is a structured set of parameters, it represents many relevant user characteristics; however, it contains, among other things, the complete user model.

With respect to our needs, we consider primary dimensions: (1) the adopted method for modeling users and (2) the richness and generality of user data modeled. We also consider, as secondary dimensions: (3) the kind of user data acquisition (e.g., explicit or derived data collection) and (4) the type of user adaptation (e.g., content or presentational adaptation).

User data acquisition is both explicit and dynamically computed taking into account the user's behavior during the current session.

3.5.3. Profiling Provisioning Environments. In this section, we present some profiling methods for gathering information regarding the network status, the position of the user and of people and objects in the user's surroundings, and the user's environment.

Bandwidth Estimation. An estimate of the data rate that can be transmitted by the network link that connects the service provider to the user is important for determining the adaptation parameters. Some techniques estimate the end-to-end available bandwidth by means of streams of probing packets that the source (server) sends to the receiver (client).

Location. Currently, a number of mobile computing applications provide services targeted to the user's location. Navigation systems, emergency services, mobile tourist guides, and proximity marketing are only few examples of location-aware applications. To support such applications, many different location systems and technologies have

been developed to provide users and devices with information about their physical location and other people and items located in their surroundings.

The most renowned outdoor positioning technology is the global position system (GPS). One of the first indoor positioning infrastructures. Radiofrequency identification (RFID) systems utilize a set of readers that can read data through Electromagnetic transmission from RFID tags. RFID tags can be either active or passive. Active tags have radio capabilities and ranges of up to hundreds of meters. Passive tags only reflect signals received from readers; thus, their communication range is smaller

4. SERVICE MANAGEMENT FOR THE END USER APPLICATION

A service is context aware if it can transparently adapt its behavior according to the requirements of the entities for which this service works the execution of a service is adjusted when details on resources are known. These details concern the status of a resource, which acts as a context source for the service.. Before a service is invoked, the user's preferences are handled. These preferences regard the user's current location, order of activities, and current time and are stored in the user's context, which acts as a context source for the service.

4.1. Adaptation techniques

The service adaptation is defined by changing services behaviors to be used in the current context. Adaptation rule can be used to define adaptation actions to apply. A rule is modeled by a pair (Expressions, Actions). Expressions describe a contextual situation of the application, Action define the transformation to be done for modifying the behavior of application or service. The adaptation technique can be applied in content, presentation and/or appearance.

4.2. Content adaptation. Application services provide a set of multimedia data (images, videos, sound, text...). These services should be adapted according to the context where they will be used. For example, if the application terminal doesn't support the image format PNG, the image format should be converted. Many other types of adaptation can be used to provide data directly usable by end user application.

Data adaptation tools used in [9] are web services that execute operations for content adaptations (such as image compression, voice synthesis; text translation, etc.). Each

adaptation service take as input a data to adapt and its description, then provide as output the adapted data and its new description.

4.3. Presentation adaptation. Interaction interfaces assure data exchange between the end user and different service of the application. These interfaces should adapt data to application terminal and user preferences.

5. CONCLUSION

This paper described a reference model that should be validated with context aware system. This system should contain context aware services that can be adapted according to context acquired from a sensors (we will use the LivePlatform to communicate with sensors) and to user and device context.

6. REFERENCES

- [1] Dey, A.K., *Understanding and using context*, Personal and Ubiquitous Computing Journal, 5(1), 5–7, 2001.
- [2] Melanie H and Gerhard A *Context Models and Context Awareness*, chapter 11, Handbook of Research on Ubiquitous Computing Technology for Real Time Enterprises, IGI Global, 2008.
- [3] Seng L. *Context-Aware Pervasive Systems Architectures for a New Breed of Applications*". Auerbach Publications 2007
- [4] Yih-Farn C and Rittwik J *Mobile Middleware: Processing Context-Related Data in Mobile Environments*. The Handbook of Mobile Middleware pp 877-900. Auerbach Publications 2007.
- [5] Kanter T, Jansson C-G, Jonsson M, Kilander F, Wei Li, P L, and Gerald Q. M. *Context Middleware for Adaptive Mobile Services*. The Handbook of Mobile Middleware pp 1105-1136. Auerbach Publications 2007.
- [6] Strang T and Linnhoff Popien, C *A Context Modeling Survey*. Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004
- [7] Klyne, G., Reynolds, F., Woodrow, C., Ohto, H., Hjelm, J., Butler, M.H., and Tran, L., *Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0*, W3C Recommendation, World Wide Web Consortium (W3C), 2004,

[8] Open Mobile Alliance, *User Agent Profile Specification*, Technical Report WAP-248-UAPProf20011020-a, Wireless Application Protocol Forum, 2001, <http://www.openmobilealliance.org/>.

[9] Chaari T, *Adaptation d'applications pervasives dans des environnements multi-contextes* These de doctorat, INSA-Lyon. 2007.

How to cite this article

Beggas M. Towards a model of context-aware infrastructure in mobile ubiquitous computing. J Fundam Appl Sci. 2011, 3(1), 47-57.