2009 International Conference on Advanced Information Networking and Applications Workshops

A Service-Oriented Platform for Pervasive Awareness Systems

C. Goumopoulos[1], A. Kameas[1,2], E. Berg[3], I. Calemis[1]
[1]*Research Academic Computer Technology Institute, DAISy group, 26500 Rion, Patras, Hellas*
{goumop, kameas, calemis}@cti.gr
[2]*Hellenic Open University, 23 Sahtouri Str., 26222, Patras, Hellas*
kameas@eap.gr
[3]*Telenor Research and Innovation, Trondheim, Norway*
erik.berg@telenor.com

Abstract

Awareness systems are a class of computer mediated communication systems that help individuals or groups build and maintain a peripheral awareness of each other. In this paper a service-oriented platform is presented, that supports awareness and informal social communication between community members, distributed over multiple smart home environments. The ASTRA platform, developed in the context of an EU research project, provides a generalized solution to the development of awareness applications that are based on the concept of pervasive awareness, i.e., where awareness information is generated as a result of using personal and home devices and smart objects, which capture and exchange information about the user semi-autonomously.

1. Introduction

The European IST project ASTRA [1] aims to define a framework for supporting the conception and the design of pervasive awareness systems, specifically those that are intended to support social relationships. This framework consists of theories to guide the design and evaluation of pervasive awareness systems supporting social use and technological solutions in the form of a service oriented architecture, tools and applications that support communities to create, adapt and appropriate pervasive awareness systems.

Pervasive awareness systems are computer mediated communication systems whose purpose is to help connected individuals or groups to maintain awareness of the activities and the situation of each other. In the domain of group-work where awareness systems were first studied, awareness can be defined as "an understanding of activities of others that provides a context for your own activities" [2]. In a more social context, interpersonal awareness can be considered as an understanding of the activities and status of one's social relations, derived from social interactions and communications with them [3]. Awareness systems promise to address pressing social problems: elderly living alone, families living apart for large parts of time, monitoring the well being of an ill relative, etc.

There is a variety of different applications that try to mediate awareness and support informal collaboration. The systems range from text-based applications over audio/video-based systems to virtual reality systems. In addition to the single-media applications, several cross-media approaches exist.

The ASTRA platform, provides a generalized solution to the development of awareness applications that are based on the concept of pervasive awareness, i.e., where awareness information is generated as a result of using personal and home devices and smart objects, which capture and exchange information about the user semi-autonomously. The platform and the assorted end-user tools assist in establishing the feeling of connectedness between community members [4].

The remainder of the paper is organized as follows. In Section 2 the awareness management process is outlined along with the basic notions required to model the problem domain. Section 3 discusses the ASTRA SOA platform. We identify the basic services and briefly discuss the basic components of the platform. Then we shall present how smart objects in a person's environment can be used to capture and convey awareness information by giving an example awareness application. Finally, our conclusions are outlined.

978-0-7695-3639-2/09 $25.00 © 2009 IEEE
DOI 10.1109/WAINA.2009.197

2. Awareness management

2.1. Awareness model

Context and context awareness play a crucial role in modelling awareness systems. We follow the definitions given by Dey [5]: "*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, including the user and application themselves.*" and "*Context-awareness is a property of a system that uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task and situation.*"

In our approach we consider context from a user-centric perspective and emphasize its use to achieve *peripheral awareness* i.e., presenting information in unobtrusive ways, by means which Weiser and Brown referred to as "calm technologies" [6]. Such means engage both the center and the periphery of our attention, and in fact move back and forth between the two. A classification of awareness according to the awareness situation of a user or a community is as follows: *activity awareness,* revealing what one or more community members are doing; *presence awareness,* representing the knowledge of who is around in a community; *group awareness,* representing the knowledge of activity and presence awareness of a community; *situated awareness,* representing contextual awareness; *social awareness,* referring to the information that a person maintains about others in a social context. For any of the above categories an awareness system may handle, it needs to tackle the questions of what users should be made aware of and how they should be made aware of it.

An approach for the conceptualization of awareness systems proposes the description of the awareness in reference of the activities that a person is made aware of. Based on this approach, Metaxas and Markopoulos introduced an abstract formal model of awareness systems that incorporates related concepts and supports reasoning regarding social aspects of using awareness systems [7]. Their model draws the basic notions of *focus-nimbus* by the work of Benford-Fahlén [8], who applied them in a spatial model of group interaction, in order to address mutual levels of awareness within a virtual environment. Briefly, focus represents a sub-space within which a person focuses their attention. One is more aware of objects inside one's focus and less aware of objects outside of it. An entity's nimbus is a sub-space within which it makes some aspects of itself available to others. This could be its presence, its identity, its activity or some combination of these.

2.2. Awareness mediation

In a high level view an awareness management process is responsible to control and manage incoming and/or outcoming awareness information. Controlling incoming awareness information of others implies the user is able to control the level of awareness detail she will accept to prevent disturbance, whereas controlling outgoing awareness information about herself implies she is able to control privacy issues. Information can be filtered by introducing rules and constraints that can be configured by the user via a defined vocabulary (e.g. an Ontology) and the services of end-user tools.

Information flow from one actor to another mediated by computer-based systems consists of several stages. In [9] a three-staged process is given, which takes the following steps: "*First, information has to be collected. Second, the collected information has to be distributed. Third, the distributed information has to be presented*". Following this view the ASTRA awareness management process is depicted in Figure 1.

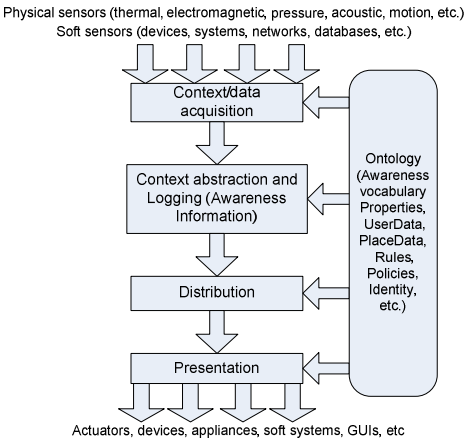

Figure 1. Awareness management process

Awareness information may be mediated in various ways. Synchronous awareness provides the user with up to date information (e.g., explicitly generated by the user or passively collected by sensors) while asynchronous information offers information about stored events (e.g., retrieved from a history or logging utility). The former are usually delivered by a mechanism called server-push (e.g., by notification services). The latter mostly relies on the client to retrieve the information when needed (client-pull).

Notification services are required for the realization of awareness management in terms of information distribution. A model that provides a loosely coupled form of interaction required for dynamic awareness systems is the *publish/subscribe* model [10].

Presentation does not only mean the display of information. Awareness information may be presented in other perceptible ways using physical artifacts and devices surrounding the user. According to HCI experts it is a visual (or sonic or other perceptual) representation of information from which the presence, activities, and other characteristics of members of a social collectivity may be inferred, and by extension, can provide the basis for making inferences about the activities and characteristics of the group as a whole.

3. ASTRA Service-Oriented platform

The ASTRA platform design principles and concepts follow the service-oriented architecture (SOA) paradigm [11]. SOA has been envisioned as an evolution of the component-based architectures centered on the concept of service. This can be applied in the design of distributed applications that are seen as a composition of services. In addition, the service concept can be applied recursively, since a system component can provide a service, but simultaneously it can encapsulate a composition of services from its service requesters.

In an SOA environment, resources on a network are made available as independent services that can be accessed without knowledge of their underlying platform implementation. A provided service is usually embodied in a set of interfaces, each of which represents an aspect of the service. In general this set contains the operations that a service supports, and some information on how to access these operations. Service interfaces can be published in registries, which also provide services themselves (publish and discovery services), allowing the potential service requesters to discover and access these services.

The use of a SOA approach in the development of the ASTRA platform implies the introduction of a middleware layer to support the interactions between the application components. This middleware provides important services as well in order to support a variety of awareness applications in a general way.

3.1 Components and services

In ASTRA platform we identify the following basic categories of services: awareness support services, context provisioning and logging services, user and community management services, notification services, service life-cycle (e.g., registering a service), discovery services (service/context discovery) and third party services (e.g., communication, identity services). These services are distributed in a number of system components which collectively provide the ASTRA platform functionality (Figure 2). We briefly discuss below the basic components of the ASTRA platform.

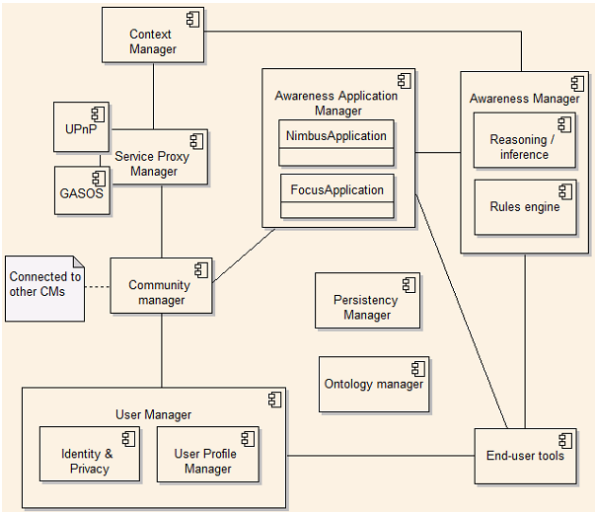

Figure 2. ASTRA component architecture

User Manager (*UM*): The UM manages users, their profiles, and identities. It is involved in the basic initial user authentication, and will then provide the necessary user profile data to services in the platform. The user profile contains personal data and user-specific application data like awareness states.

Community Manager (*CmM*): The CmM manages the communities in the ASTRA platform. To support community oriented interaction the CmM component is responsible for providing the possibility to define, share and connect virtual community representations in within which users can share awareness information.

Service Proxy Manager (*SPM*): SPM is responsible to discover local services and provide their data in a specific format in higher lever components. There should be an SPM for each of the different protocols that one can discover services in the local space, e.g., UPnP, Bluetooth or middleware based such as GAS-OS [12]. Each SPM has operations to discover devices/services, connect/disconnect from them, read service sensing information and enforce any actuating information to an appropriate service.

Context Manager (*CnM*): The CnM provides a unified repository of the collected local discovered services reflecting the hardware environment at each moment. It handles the runtime gathering of local context values and the incoming requests for activating an actuating device through the correct SPM.

Awareness Manager (*AM*): The AM is responsible for dealing with awareness information and in particular with awareness state inference, aggregation and logging. It is connected to the rule engine kernel

which is responsible for awareness assessment according to a set of rules (Figure 3).

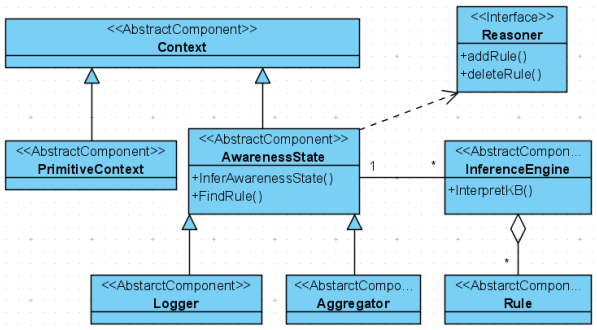

Figure 3. AM design

Awareness Application Manager (*AAM*): The AAM is responsible for managing the local awareness applications which can be either a nimbus or a focus. An awareness application is a service representation of a specific awareness state that can be made available to a community. The AAM is associated with a collaboration instance of the CmM, which in turn is the communication point with other nodes. The AAM is also associated with an AM. Whenever the awareness state is changed, the AM informs the AAM, which in turn checks to see if any communities need to be informed. The AAM at the same time monitors communities for such state changes that are of interest.

Ontology Manager (*OM*): The OM manages the ontology with core knowledge of the ASTRA domain. The ontology provides for example a common lexicon for identifying similar services. In that way, high-level descriptions of services and resources independent of the context of a specific application are possible, facilitating the exchange of information between heterogeneous entities and the discovery of services.

The ASTRA services have been developed based on the OSGi service platform (www.osgi.org). OSGi leverages the Java Virtual Machine with a dynamic component model based on services, suitable for a wide range of devices from mobile phones to servers.

3.2 ASTRA deployment

Figure 4 shows the 3-tier architecture defined for the deployment of pervasive awareness applications.

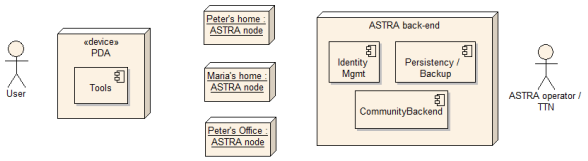

Figure 4. 3-tier deployment architecture

The *ASTRA node* represents the main deployment unit for supporting awareness application's business logic in terms of executing rules. This node can for instance be the local deployment in one's home, at the office, at a university or in the car. The nodes work together on a P2P approach and their common channel of communication is through the CmM; different users collaborate with each other (i.e., sharing information) through so-called collaboration instances of the CmM; this is where services and attributes are exposed to other users in the context of a shared community.

The *ASTRA backend* represents the resource management layer responsible for managing identities and providing a persistency solution to components. Identities are dependent on an identity provider, which will manage and authenticate identities across different nodes. For a user to trust the identity of another user, a trusted third party (TTP) is required to provide identity management. In addition, it also provides the back-end component of the CmM, which is responsible for synchronizing community states between the distributed CMs of the local ASTRA nodes, as well as providing persistent storage of this state for bootstrapping purposes of the local nodes. This state is comprised of: i) community members ii) shared awareness applications and iii) shared community space. The CmM mediates incoming and outgoing awareness information via publish/subscribe [10].

The *ASTRA devices*, which may be a computer, PDA or mobile phone, allow end-users to use the ASTRA tools for managing (editing, configuring, etc.) awareness applications. The ASTRA tools use a web interface and can interface to the ASTRA platform services (Figure 5).

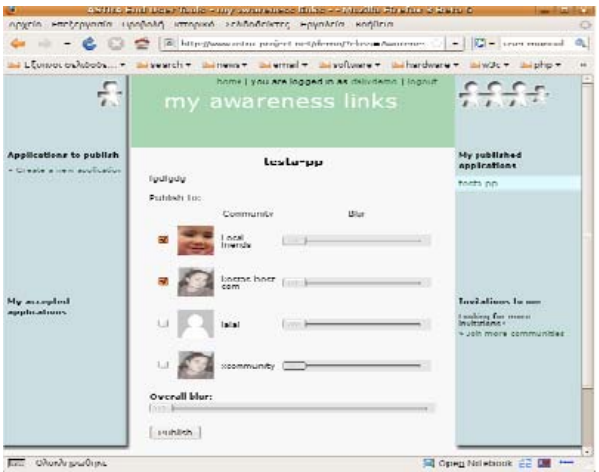

Figure 5. Sample ASTRA end user tool

4. An example awareness application

We illustrate our example based on the following simple scenario. "*Martha and Amanda have met at the retired professional engineers club and they are now living alone; sometimes they invite each other for a walk. They like to do this easily and without social pressure on each other so they recently, installed ASTRA system that helps them convey their wish for a walk. When they feel like walking, they can turn a 'switch' installed in their living room; the system indicates their intentions to the other side by lighting a small lamp in a visible position in the living room.*"

An awareness application that provides sharing of awareness information between two or more users is a distributed application requiring the following actions: i) controlling awareness information definition ii) controlling awareness information presentation and iii) controlling the distribution policy of the awareness information either as incoming, or as outcoming.

Artifacts are used as service providers and have an active role in the first two parts. Artifacts are physical objects that may be augmented with sensing, computing, communication, and actuation capabilities. Using a notification service, clients (e.g., Amanda) may subscribe to receive information about multiple and/or particular awareness information that publishers (e.g., Martha) have previously published.

On Martha's home we need to define an awareness state named "`wish for walk`" shortly `W4W` (i.e., a nimbus) by using an awareness coordination object (ACO). The ACO used in this example is a mathmos tumbler[1] light artifact. This artifact can sense its position, changing throughout a full spectrum of colours as it's position changes. The rule for announcing the particular nimbus to Amanda is:

```
if mathmos color is purple, then notify
   my W4W nimbus to subscribed users
```

Amanda must link Martha's awareness state with a visualization object in her living space (an object in Amanda's focus). On Amanda's side, five artifacts are involved for controlling visualization of the `W4W` state. The logic for producing the proper focus according to the received nimbus is outlined graphically in Figure 6.

Amanda uses her ASTRA editing tool and she associates visualization services of a Nabaztag[2] bunny artifact activated when Martha expresses her `W4W` from her house. Amanda does a second configuration by setting up her ASTRA application to send an SMS in her mobile phone, if she is not in the house.

[1] http://www.paramountzone.com/mathmos_tumbler.htm

[2] http://www.nabaztag.com/index.html

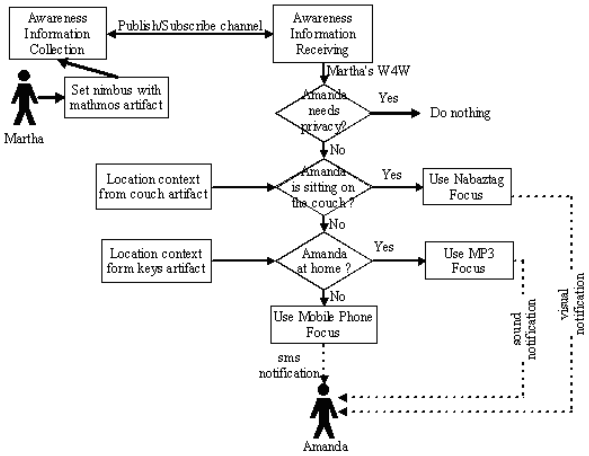

Figure 6. Controlling awareness presentation

The sequence diagrams in Figure 7 show interaction between the users, the interfacing tools, the ASTRA system components and artifacts for part of the operations executed in the scenario of our example.

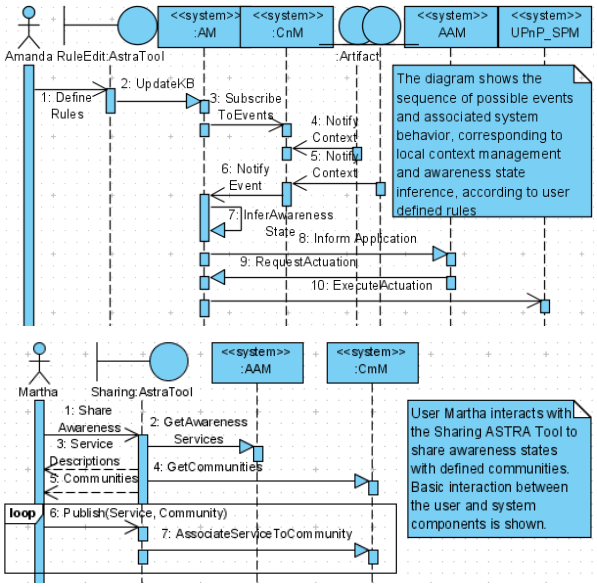

Figure 7. Sequence diagrams

Table 1 lists all the artifacts and summarizes their use, role and service provision in the awareness application example. This example prototype shows also a context-aware service composition approach. We use context information (e.g. user location) in the service association phase (we associate an awareness service with a presentation service) to invoke different actions given the user's current context.

5. Conclusions

In this paper, we have presented a SOA platform that supports the composition and management of pervasive awareness applications. A 3- tier system has been developed to support this service. The architecture presented uses smart objects in the person's space as conveyors of this person's awareness state. The person can configure these objects using tools, which support the discovery of artifact services and their combination in a simple way, as well as the definition of awareness applications using first-order logic rules on these services.

Although the concept of using rules to govern the behavior of a system is widely used in society and should not be hard to understand, their application to configure the awareness data publishing and privacy could be complex and possibly too difficult for everyday people. Depending on the different levels of the technical competency of end users, and their willingness to appropriate the system, different levels of end user involvement is expected. Besides the typical application configuration with web-based user interfaces we seek to reduce complexity by designing tangible user interfaces through smart configuration objects. These configuration objects will enable, for example, an everyday user to select and activate a pre-fabricated awareness application or to configure the recipients of the awareness data in a tangible way.

Acknowledgement

This work has been funded by EU FP6 IST STREP project ASTRA, under Grant agreement No. 29266.

References

[1] IST FET Open project ASTRA (Awareness Services and Systems – Towards theory and ReAlization), website at: http://www.astra-project.net/

[2] P. Dourish, and V. Bellotti, "Awareness and coordination in shared workspaces" Proc. ACM Conference on Computer-Supported Cooperative Work (CSCW' 92), ACM, New York, pp. 107-114, 1992.

[3] B.A. Farshchian, "Presence Technologies for Informal Collaboration", In: G. Riva et al. (Eds.) *Being There: Concepts, Effects and Measurement of User Presence in Synthetic Environments*, IOS press, Amsterdam, The Netherlands, pp. 209-222, 2003.

[4] N. Romero, P. Markopoulos, J. Baren, B. Ruyter, I. Wijnand, and B. Farshchian, "Connecting the Family with Awareness Systems", *Personal and Ubiquitous Computing*, Springer-Verlag, pp. 299-312, 2007.

[5] A.K. Dey, D. Salber, and G.D. Abowd, "A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications", Human-Computer Interaction, pp. 97-166, 2001.

[6] M. Weiser, and J.S. Brown, Designing Calm Technology, Xerox PARC, October 5, 1996, Available at: http://www.johnseelybrown.com/calmtech.pdf

[7] G. Metaxas, and P. Markopoulos, "Aware of what? A formal model of Awareness Systems that extends the focus-nimbus model", Proc. Engineering Interactive Systems, 2007.

[8] S. Benford, and L. Fahlén, "A spatial model of interaction in large virtual environments", Proc. European Conference on Computer-Supported Cooperative Work, Kluwer, pp. 109-124, 1993.

[9] M.E. Sohlenkamp, Supporting Group Awareness in Multi-User Environments Through Perceptualization, Technical Report, European Research Consortium for Informatics and Mathematics at FHG, 1999.

[10] P. Eugster, P. Felber, R. Guerraouli, and A. Kermarrec, "The many faces of publish/subscribe", *ACM Computing Surveys*, pp. 114–131, 2003.

[11] T. Erl, "Service-Oriented Architecture: Concepts, Technology, and Design", Prentice Hall, 2005.

[12] N. Drossos, C. Goumopoulos, and A. Kameas, "A conceptual model and the supporting middleware for composing ubiquitous computing applications", *Journal of Ubiquitous Computing and Intelligence*, pp. 174-186, 2007.

Table 1. Artifacts and their role in the awareness application prototype

Artifact	Role	Context/Media	Service Description
Mathmos Lamp	NP (Nimbus Provider)	Wish for walk awareness state	By using this artifact the user defines an outgoing awareness information
Keys	LCP (Location context provider)	User location is 'home'	This artifact is used to determine user's location. The user presence at home can be determined by using mote technology (e.g., Crossbow MICA2dot) placed in the user's key-fold. When the user is present at home, any signal from the mote can be detected by the home base station.
Couch	LCP	User location is 'living room'	This artifact is used to determine user's location. The pressure plates indicate the user presence at the room the couch is placed in.
Nabaztag	FP (Focus Provider)	Nabaztag messages	This artifact is used to present (e.g., by blinking and moving ears) incoming awareness information.
MP3 Player	FP	Vocal messages	This artifact is used to present incoming awareness information.
Mobile Phone	FP	SMS messages	This artifact is used to present incoming awareness information.