A Context-Aware Framework for Collaborative Activities in Pervasive Communities

Christopher Lima, Universidade de Aveiro, Instituto de Telecomunicações, Aveiro, Portugal

Mário Antunes, Universidade de Aveiro, Instituto de Telecomunicações, Aveiro, Portugal

Diogo Gomes, Universidade de Aveiro, Instituto de Telecomunicações, Aveiro, Portugal

Rui Aguiar, Universidade de Aveiro, Instituto de Telecomunicações, Aveiro, Portugal

Telma Mota, Portugal Telecom Inovação, Aveiro, Portugal

ABSTRACT

Pervasive environments involve the interaction of users with the objects that surround them and also other participants. In this way, pervasive communities can lead the user to participate beyond traditional pervasive spaces, enabling the cooperation among groups taking into account not only individual interests, but also the collective and social context. In this study, the authors explore the potential of using context-aware information in CSCW application in order to support collaboration in pervasive environments. In particular this paper describes the approach used in the design and development of a context-aware framework utilizing users' context information interpretation for behaviour adaptation of collaborative applications in pervasive communities.

Keywords: Computer Supported Cooperative Work (CSCW), Context-Aware, Design, Development, Pervasive Computing, Social Computing, Communities

INTRODUCTION

Pervasive environments have achieved a matured level, thanks to the advance of the technology, mainly through sensors and a growing variety of mobile devices. By combining these smart spaces with social computing it is possible to create a richer environment in which real world and virtual information can be utilized to meet the user needs. Smart spaces provide an environment aware of context information that enables users to interact electronically with other users and devices in their environment and to perform their activities.

Context information can be any relevant information available about location, users, services, devices and their relationships. In addition, the information available in social network sites can also be useful for pervasive systems as context information, where the

changes of environment are important to offer powerful ways of working and communicating.

Social computing has recently become a subject of study in a variety of areas related to context-adaptive systems. Context-adaptive systems are systems that can dynamically adapt their behaviour at run-time in response to changes in context information such as: users' preferences, operational environments, and underlying infrastructure (Hussein et al., 2012).

In the present scenario, pervasive systems lack the ability to deal with social group context, besides the individual. Recently, FP7 SOCIETIES project tries to tackle the creation of communities of smart spaces, investigating group context information, privacy and preferences. The communities of smart spaces envisage the integration of pervasive with social fields to provide personal smart spaces that can move along the users providing group capabilities following the users' interests.

Even though the pervasive environments can support large and small groups, usually collaborative activities happen often in small groups of up to 10-15 people (Farshchian & Divitini, 2010). Furthermore, several of these activities in smart spaces may involve users collaborating common tasks where collaborative applications take advantage as we state in (Lima et al., 2012).

In order to demonstrate the synergy of collaborative activities, pervasive communities and social computing, we present a context-aware framework for collaborative applications, which envisages cooperation in pervasive environments.

The goals of the framework are to:

- Provide adaptability for communication, coordination and relationships to CSCW.
- Facilitate collaborative work by providing pro-active behaviour for collaborative tools.
- Enhance collaboration work taking into account mobility of the users and physical distribution.
- Provide cooperation among users based on physical and social interaction.

This article is organized as follows: section "Background" presents a background regarding the framework which involves social computing, context-awareness in groups, CSCW systems and pervasive communities. Section "Related Work" explores the previous work in groups for pervasive environments and context aware applications for CSCW. Section "Framework Overview" describes the framework architecture, highlighting the new developed features and issues that have been addressed during the implementation. Finally, section "Conclusions" provides conclusions and directions for future work.

BACKGROUND

In this section we introduce with more details the background domain that formed the basis for the design and development of the context-aware framework for CSCW applications. The domain areas involve social computing, context awareness and pervasive computing with focus on communities.

Social Computing

Social computing has recently become a subject of study in a variety of areas related to information and communication technologies. Erickson (2013) defines social computing as systems that mediate social information across collectivities such as teams, communities, organizations, populations, and markets. Moreover, he states that social computing systems need to have mechanisms for managing information, identity, and their interrelationships.

In Wenger (1998), the author explains the importance of communities of practices. The "Communities of Practice" definition describes a group of people who share an interest, a craft, and/or a profession. Based on the work by Wenger, Gunawardena et al. (2010) presented a theoretical framework for building Online Communities of Practice in social networks. The framework attempts to develop a theoretical framework to understand learning among groups of individuals that utilize social

networking applications. People who participate in such communities learn accepted work practices that reinforce each other and become more productive.

With the advent of Web 2.0, it is possible to enable provision of content, goods and services in the areas of personal life and business by anyone. This new paradigm originated the crowd computing term, which targets the human interaction to share tasks and ideas making use of tools such blogs, wikis and SNS (Social Networking Sites).

Social Network Sites

Social network sites have gained much popularity nowadays mainly due to Web 2.0. This major trend is due in part to the development of applications by third parties, which led to differentiation in available services by existing social networks. These applications allow access to various types of information, such as profile, activities occurred, friends list. Furthermore, with recent advances in mobile technologies, mobile devices are increasingly becoming associated with their owners and an attractive platform for development of social and smart space applications.

Even though being a relative new area, social computing integration with other fields is rising due the social context in a few collaborative areas such as enterprise, eGov and computer supported collaborative learning (CSCL) (Erickson, 2013).

Computer Supported Cooperative Work Systems (CSCW)

The concept of a CSCW (Computer Supported Cooperative Work) system is to provide rich interaction supporting co-located groups in a common task or goal. Much effort has been dedicated to design applications that support individuals who are in a synchronous collaboration (e.g. co-browsing, audio/video conference), and who interact in asynchronous way (e.g. wiki, email). These efforts generally aim to create virtual spaces to promote collaboration

by providing mutual awareness, communication and coordination (Farshchian & Divitini, 2010).

Users in a collaborative environment are usually a group of people with similar preferences, tasks and needs. In this way, groupware is a very relevant subject of study in CSCW and Social Computing.

Groupware applications have become increasingly widespread in all branches of research and industry, for both work and learning (Temperini & Nanni, 2012). In particular, group awareness is considered a crucial point to allow all actors for a better grounded and productive collaboration (Dourish & Bellotti, 1992).

In order to enable groupware applications to adapt to discovered context information, the context-aware applications should be designed to consume and react to the contextual information that serves as resources for cooperative work. In the CSCW domain, context-aware applications presents challenges in designing and implementing, which involves ambient resources, privacy policies and infrastructure for the collaborative activities (Kulkarni et al., 2012).

Context-Awareness in Group of Smart Spaces

In Dey (2001), the author defines context as: "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 applications themselves".

A large part of smart spaces challenges reside on the use, interpretation and reasoning techniques for context information. Especially when these information needs to be used in real time by the system, services and third-party applications.

Usually pervasive platforms provide the means to receive context information and apply the information to build context-aware applications. Several context management systems are based on the context broker architecture similar to CoBrA (Taylor et al., 2011).

A context management component is responsible for collecting the context information from sources and sensors, modelling and maintaining the current and historic data in appropriate repositories. The context information describes the users' environment, where applications can retrieve it and decide how to behave based on this information. Below is presented some of the components usually found in a context management. It is important to stress that some of them can slightly change or even be optional like the history management:

- Context Broker: Responsible for providing data in in a synchronous and/or asynchronous way.
- Context DB Management: Comprises a Database Management System (DBMS) to store and maintain the information. The management is done by database queries (i.e. retrieval, update, addition, removal).
- Context Source Management: Responsible for managing the external sources, including for instance SNS data and individual's location.

The social interaction is an authentic activity and by itself pervasive, taking place in people's everyday professional and personal lives. Consequently, social networks provide an opportunity to access detailed contextual information that is hard to achieve through other means. The social context plays an essential role in describing the situation in which users interact. Information about social relations can serve as an example of social context for users. Hence, the social context can be considered as a specific type of context information.

The context information is structured as an entity and attribute. The properties of the person or the device are described by multiple attributes that are assigned to an entity. For instance, the name, the age and the location of a person can be represented as different attributes of an entity *person*. In the case of a community, members represent an attribute, while the community itself the entity.

However, in order to deal with communities of smart spaces, the approach in the SOCIETIES project was to extend the context management to address the needs of a distributed context. A hybrid approach was chosen in order to address scalability issues in terms of processing, storage, remote communication and power consumption (Liampotis et al., 2012).

Pervasive Communities

One of the most important issues for pervasive computing is to create an infrastructure to support multiple users in different environments. In these systems, it is important to effectively support development and run-time execution of services/applications in wide-scale of scenarios. Smart spaces are by nature context aware, where location (e.g. home, office), activities (e.g. walking, driving) and environment features (e.g. hot, cold) enable users to interact with computational resources to perform activities. These environments can provide better user experience by allowing a user to connect new devices flexibly and to access all the information in the multi device system from any of the devices (Smirnov et al., 2008).

A pervasive community is a term coined by the SOCIETIES (2014) project to designate a group of smart spaces. In the use-case of the project, a smart space not only includes physical locations, but users who can interact with the environment and other individuals. Thus, these personal smart spaces can communicate with other devices and users through traditional devices that range from notebooks to mobile devices.

The aim of the project is to explore aspects associated to group interaction through pervasive systems and social networking. The project introduces the notion of a Cooperating Smart Space (CSSs). Each CSS consists of multiple devices, both fixed and mobile interconnected in an ad-hoc manner, owned by a single user. The CSSs can interact with other individuals and communities whenever possible (Doolin et al., 2012). Each community can offer various characteristics to CSSs, such as a set of shared

resources and services enhanced by additional functionalities provided by others CSSs members of the same communities.

The main focus of the SOCIETIES platform is to facilitate creation, organisation and management of communities, providing better experience for individuals and the communities. Each platform component is responsible for a specific task, providing functionalities via APIs used by internal components and third party services. Some of the essentials components are: communication framework, device management, privacy and trust management, context management, community orchestration and personalization.

The concept of community of practices adopted by the project and associated with the social computing can offer a different impact in the CSCW area. The CSCW can take advantage of the pervasive communities in a number of scenarios as described in (Lima et al., 2012), where users with the same degree of similarities can be grouped in collaborative sessions. Two scenarios that we emphasized were: Computer-supported collaborative learning (CSCL) and enterprise.

RELATED WORK

Some studies regarding groups in pervasive environments are available in the literature. PICO (Pervasive Information Community Organization) (Kumar et al., 2003), presented a framework with the objective to create pervasive communities that can collaborate proactively in areas such as telemedicine, military and crisis management. This aims to achieve a sequence of events that can lead to the creation of communities (Kim et al., 2006). More recent approaches attempt to tackle the problem of pervasive communities supporting also mobile users. POPEYE project approached spontaneous virtual communities that can be formed in a peer-to-peer fashion for collaborative work (Meyer et al., 2011). Unfortunately, the project focus is in transport issues, lacking in services integration and social computing. In Foell et al., 2007 it

is proposed a framework to form communities according to users' interests and participation. It applies recommendation and personalization algorithms to perform self-organization based on online social sites. Nonetheless the work only encompasses the context originated from online social sites, not providing management for face-to-face communities and integration in several pervasive environments.

There are few related works that try to gap CSCW and context information. Intermezzo (Edwards, 2005) is a collaboration toolkit supporting coordinated information sharing, user awareness, session management and policy control. It addresses dynamic flexibility, by allowing applications to modify the behaviour of the toolkit based the situational context of the collaboration (Dourish et al., 2000). However, the drawback is the integration of social interaction aspects of the users and the group approach, which takes the virtual and physical world interactions into account in order to adapt it to the user needs.

The ECOSPACE Project (Prinz et al., 2006) proposes an integrated collaborative environment to support semantic description of collaborative services for orchestration and choreography. Besides a semantic description of each service, a semantic description of the composition of services is required to coordinate their orchestration. Nevertheless, this part of the project remains as a design aspect.

Recently Haake et al. (2010), proposed a generic framework for context modelling in collaboration environment providing adaptive rules to support collaborative activities. The conceptual approach proposes use of semantic languages to structure context information. However, the integration does not take into consideration the aspect of groups and social integration.

In summary, the proposals involving context-aware groups for collaborative work still lack on environment integration, mobility or social capital where synergy could enhance collaborative experiences.

FRAMEWORK OVERVIEW

This section presents the context-aware framework for CSCW applications that has been developed in order to enable context awareness in collaborative activities. The design of the framework is divided into three layers: modelling of context information, the context interpretation, and the runtime actions.

In order to build the framework prototype the architecture was modelled in UML and was developed in Java as an OSGi bundle (OSGi, 2014). Some concepts and description of the components have been presented in publications such as (Lima et al., 2013). For the sake of comprehensiveness, a brief description of the components is listed below. Fig. 1 illustrates the architecture.

Context Acquisition Component

The context acquisition is the first layer of the framework and is responsible for collecting relevant context information and prepares it to the subsequent layer. As it is shown in our earlier work (Lima et al., 2012), there are some relevant appropriate context categories depending on the scenario. Some of the categories that might be used include: geographic location, current availability, current computational (mobile/desktop) resources and professional information.

The context information is acquired via an external Context Broker, which communicates with the context sources. The Context Broker is responsible for gathering context information from the users through heterogeneous sources that can go from mobile sensors to SNS. The mobile devices provide location, availability and resources. On the other hand, SNSs provide information related to user profile such as interests, professional position and expertise.

The information, retrieved from the Context Broker, is classified as long-term or short-term, depending on temporal characteristics. The long-term information is comprised by data that does not change often such as: job position, areas of interests, skills. On the other

hand, short-term context comprises data that changes frequently such as location and availability (e.g. user busy, away).

Generally, pervasive systems have by default a specific component to manage context (context management) allowing it to capture and retrieve context information. Thus, the main focus of the framework resides in the analysis and adaptation, abstracting the concern to deal with context sources providers.

Concerning privacy, it is expected that the user provides only the information that he thinks appropriate. On pervasive systems such as SOCIETIES, it provides privacy policies rules related to shared context information. When the information is retrieved for the first time, the platform requests for each user if it is granted permission to fetch and receive updates about that specific information. The permission can be granted to a specific community or a third party service. The user also has the opportunity to agree/disagree of the privacy policy rules before being inserted in a community for instance. In this case, the members of the community and the third party services that belong to that community do not have access to the data or request in runtime.

Context Interpretation Component

The Context Interpretation component is responsible for modelling the knowledge and process the information collected from the previously layer. In order to translate the context to the framework this information is stored in a graph database, where each node represents a person. The user context information is used to assign edges representing relations existing among the users, based on the long and short-term context information.

The long-term information is associated as properties of the person node as illustrated by Figure 2. The short-term is associated to a new graph that is created according to demand. Each new node of short-term information stores the timestamp and the new information using the person node as root. This behaviour is very similar to an activity stream, typically used by

Figure 1. Framework architecture
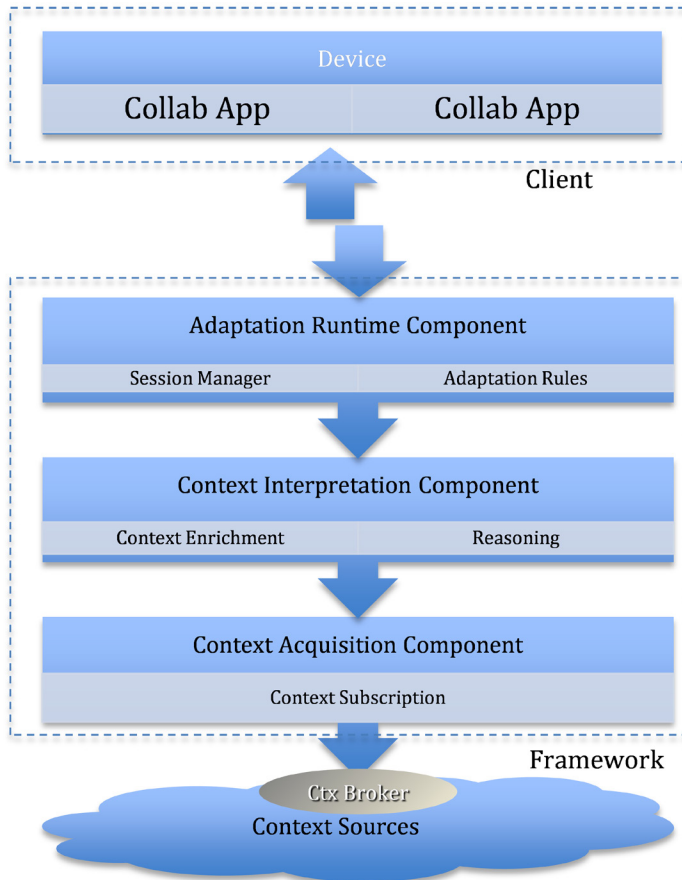
SNSs to maintain a list of recent activities performed by an individual. The framework also keeps the history of short-term information, but not for the long-term since the information in most cases is not relevant. This separation allows this layer to make decisions on how to link individuals from the same community and assigning weights through the context information available.

To create the graphs we choose the Neo4j graph database engine (Neo4j, 2014), which offers a graph-oriented model for data representation. In Neo4j graph structures consists of nodes, relationships and properties. In the framework, the nodes represent persons and also short-term context information for each individual as illustrated in Figure 2. The relationships among persons are associated with the degree of similarity and are expressed as weights, while the short-term information has a timestamp and is represented as a LIFO (Last-In-First-Out), where the last node is more recent update. The properties express the node information. Figure 2 demonstrates long-term context information and Figure 3 the short-term.

In order to extend the context information collected from the previous layer, the component performs an enrichment of context depending on the nature of the data. The information can be expressed in numeric or text values. The texts values are submitted to externals sources of Natural Language Processing (NLP) for

Figure 2. Short-term context information in the graph database
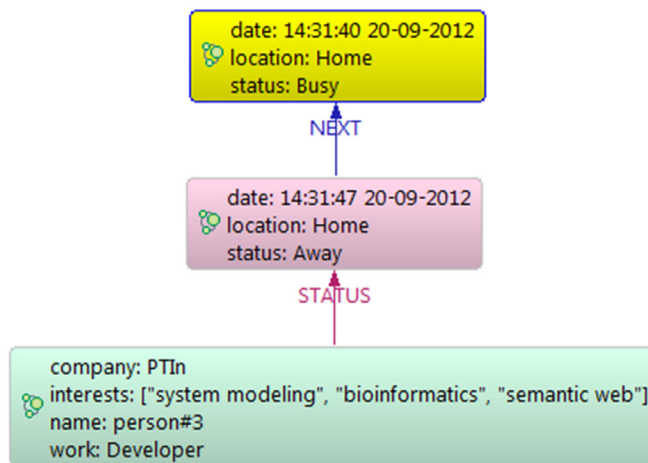
semantic analyse, returning synonymous words that are aggregated with the existing information. This enrichment provides knowledge to create rules that drive decision making in the adaption layer.

After the enrichment of the text values it is possible to assign similarities among the persons in graph. This similarity enables to associate weights between person nodes and is calculated by dividing the matched information by the total available. This is applied for both and the result is divided by two as represented below:

$$W = (matched/node1) + (matched/node2) / 2$$

This component can also be prompted to report a threshold, to select relevant persons for a given weight. This method is called automatic thresholding and it is often used in image processing for segmentation (Sahoo, Soltani, & Wong, 1988). The thresholding mean value is calculated based on the weights assigned previously. The weights values can be verified in the arrows among persons as illustrated in Figure 2.

The numeric values can be enriched using historical information available, as an example it can be used to generate temporal series. This is helpful in case certain events to not occur regularly. Lastly, all the context information can be checked automatically when an update event happens or periodically through the awareness monitor framework. This procedure will depend on the context management behaviour. If the context management, which the framework is connected to, supports subscription of context changes events the framework will react to these notifications. In case the context management does not provide such functionality; the framework will fetch the context information periodically for changes in the context broker component.

In the current implementation, the framework uses the Alchemy API (AlchemyAPI, 2014). The content to be analysed is submitted to the respective cloud and return results in JSON or XML format. Nowadays, there are several possibilities of extracting semantic meta-data such as: concept, sentimental, text categorization, etc. Other similar cloud services that can be integrated in the framework for content analyses are Synonym (Synonym, 2014) and Coginov (Coginov, 2014). The connector with external sources needs to be implemented by the developer. However, it is intended to implement other connectors for future tests.

Figure 3. Long-term context information in the graph database
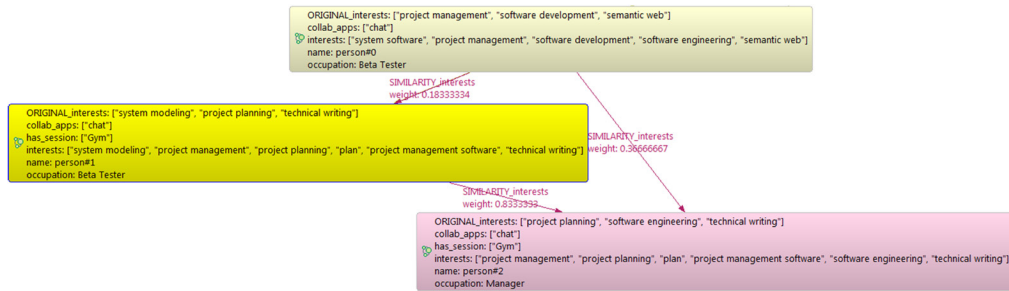
The enrichment of context information is optional step by the framework, although the enrichment plays an important role to calculate the similarity of the individuals.

Adaptation Runtime Component

This layer is responsible for managing the collaboration sessions and performs actions on the collaboration tools via adaptation rules. The session manager coordinates the session members and their integrated applications. Each session is composed by: members invited, context information matched by the rules and the users who actually accepted the invitation.

The session's history follows the same short-term approach used for the individuals, storing new nodes with a timestamp with the matched context information, invitations and participants.

Figure 4 exemplifies a session associated to one location and the participants in real-time.

The adaptation rules are expressed by IF-THEN clauses and are defined with: if "condition" then "action", e.g.: *if at least two individuals are in the same location and work at the same department and have the same interests THEN start a session*. The adaptations rules can also perform a status checking to choose which communication channel is more appropriate for collaboration or which available communication channel is common to all parties. If the person is idle for a long time the user can be kicked off the session or if busy, not invited to the session.

For each stipulated rule it is possible to assign a weight or/and a priority which will be used by the engine rule in the framework. The weight value is used to assign a score to each rule in percentage format. The rules and weights are formally represented below:

$$P = \omega 1 \times Rule1 + \omega 2 \times Rule2 + \dots \omega\text{-}n \times Rule\text{-}n$$

For instance, if the first rule is stipulated in 30% for location and 70% for interests a user may be invited for a session even if it is not present in the same location.

For priority format, the relevance is given for order designated for the rules. The rules and priorities are formally represented as:

$$P = Rule1_{(i)} + Rule2_{(i+1)} + \dots Rule\text{-}n_{(i+n)}$$

Following the previous example, if location is the first rule and the interests is the second rule; the engine should first check who is at the same location and then verify the interests.

In order to use the collaborative tools in the framework the user needs to integrate the server application with the adaptation layer, which is in charge of receiving and performing actions on the collaborative application. Thus the adaptation layer can execute actions directly in the server such as: invite user, kick user, create new session, etc.

Figure 4. History of collaborative sessions in the graph database
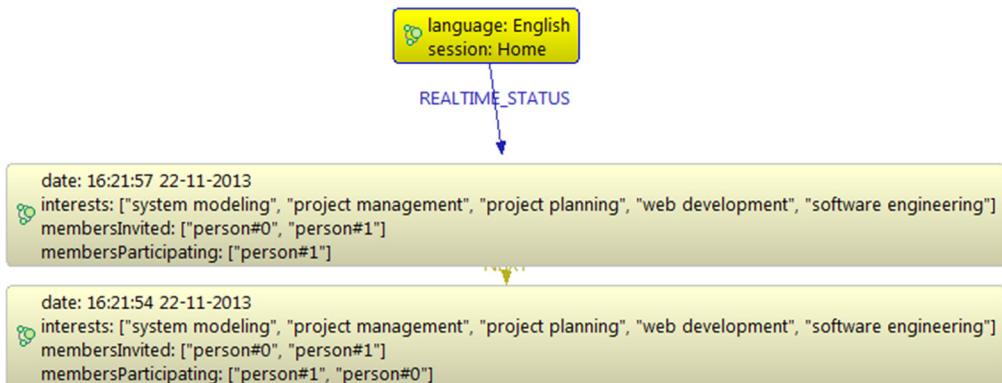
The applications can be integrated depending on its nature. Synchronous applications need to implement an abstract class, which inform how to join and leave a conference and the events related to the user such as join, leave and availability. For asynchronous applications the user needs to implement an abstract class to receive updates on which users that matches the rules assigned by him.

The applications integrated are not responsible to manage sessions and not need to be aware of the behaviour of others applications that are working at same time.

EXPERIMENTAL RESULTS

As part of preliminary tests, we created an environment consisting of the deployment of the SOCIETIES Platform and the proposed Framework as a third-party service on the same host using virtual machines.

The Framework service works on top of the platform, which provides other essentials components, but for this proof of concept it was just necessary to integrate it with the community manager and context management components. The community manager provides the members available in a specific community. The context management provides information from each member of that community. For context in-

formation subscription it was implemented a connector to receive updates from the Context Broker.

After the creation of a community, it is possible to select one to start populating the Framework mapping each user to one-person node. The first step starts with long-term context information and is structured as depicted in Figure 3. The second step is to acquire the first short-term information available. For the experience we adopted long-term information: occupation and interests. For short-term: location and status.

With all initial information collected the framework starts the context monitor. The monitor checks for update changes for the two types of context in real-time. In order to simulate the behaviour of a real scenario, we manually changed the location and status for each person.

Regarding the rules, we stipulated 2 rules and assign weights for: location and interests. Thus, depending on location and interests, the monitor checks whether it is appropriate start a collaborative session or remove a participant from a session. The rules were expressed as follows:

- First rule - Location: same location with weight 60%.
- Second rule - Interests: interests > thresholding with weight 40%.

We performed the test exclusively with mobile devices using a chat client in Android (Xabber) to verify invitations and the orchestration of participants among the collaborative sessions. It was also necessary for the SOCIETIES Android client to fill the user profile information and change the location. The preliminary test evaluated the behaviour and predictability of the actions taken by the framework.

CONCLUSION

In this article we presented a context-aware adaptive framework for collaborative activities in pervasive environments. The approach takes into consideration specifically pervasive communities of users, enabling runtime adaptation of collaborative tools in response to the context information changes. This work extends our previous work on a specification model level, presenting updates in the Adaptation layer in terms of the use of collaborative applications and session management.

The current approach aims to minimize the interaction of pervasive groups in collaboration environments by avoiding implicit interaction of the participants. The prototype was implemented demonstrating its potentials and is available online as a third party service of the SOCIETIES platform (SOCIETIES Third party services, 2014).

Many challenges remain for interpretation of existing context related to individuals and groups for CSCW. As a next step it is our intention to investigate other relevant methods for interpretation and analysis of context. We aim in the future to focus in floor control permitting to provide coordination of activities occurring amongst the collaborators.

Finally, we will perform validations to assess the applicability and practicality of our approach to verify performance, usage and usability of the framework in real situations.

ACKNOWLEDGMENT

REFERENCES

Alchemy API. (2014). Retrieved January 27th, 2014, from http://www.alchemyapi.com/

Coginov. (2014). Retrieved January 27th, 2014, from http://www. coginov.com/

Dey, A. (2001). Understanding and using context. *Personal and Ubiquitous Computing*, *4–7*. doi: doi:10.1007/s007790170019

Doolin, K., Roussaki, I., Roddy, M., Kalatzis, N., Papadopoulou, E., Taylor, N., … Kosmides, P. (2012). *SOCIETIES: Where pervasive meets social*, 30–41.

Dourish, P., & Bellotti, V. (1992). Awareness and coordination in shared workspaces. In *Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work (CSCW '92)* (pp. 107–114). New York, NY: ACM Press. doi:10.1145/143457.143468

Dourish, P., Edwards, W. K., & Paul Dourish, W. K. E. (2000). A tale of two toolkits: Relating Infrastructure and use in flexible CSCW toolkits. [CSCW]. *Computer Supported Cooperative Work*, *9*(1), 33–51. doi:10.1023/A:1008709725729

Edwards, W. K. (2005). Putting computing in context. *ACM Transactions on Computer-Human Interaction*, *12*(4), 446–474. doi:10.1145/1121112.1121117

Erickson, T. (2013). Social computing. *The encyclopedia of human-computer interaction* (2nd ed.). Retrieved from http://www.interaction-design.org/encyclopedia/social_computing.html

Farshchian, B. A., & Divitini, M. (2010). *Collaboration support for mobile users in ubiquitous environments*, 1–28. doi:10.1007/978-0-387-93808-0

Foell, S., Boessling, P., Linner, D., Radusch, I., & Steglich, S. (2007). Self-organizing pervasive online communities. In *Proceedings of the Eighth International Symposium on Autonomous Decentralized Systems (ISADS'07)* (pp. 403–410). doi:10.1109/ISADS.2007.69

Gunawardena, C. N., Hermans, M. B., Sanchez, D., Richmond, C., Bohley, M., Tuttle, R., & Beth, M. (2010). A theoretical framework for building online communities of practice with social networking tools A theoretical framework for building online communities of practice with social networking tools. *Educational Media International*, *46*(919435511), 3–16. doi:10.1080/09523980802588626

Haake, J. M., Hussein, T., Joop, B., Lukosch, S., Veiel, D., & Ziegler, J. (2010). Modeling and exploiting context for adaptive collaboration. *International Journal of Cooperative Information Systems, 19*(01/02), 71–120. doi:10.1142/S0218843010002115

Hussein, M., Han, J., Colman, A., & Yu, J. (2012). An architecture-based approach to developing context-aware adaptive systems. In *Proceedings of the 2012 IEEE 19th International Conference and Workshops on Engineering of Computer-Based Systems* (pp. 154–163). IEEE. doi:10.1109/ECBS.2012.13

IBM Presence Zones. (2014). Retrieved January 27th, 2014, from http://www.research.ibm.com/haifa/projects/imt/zones/

Kim, M., Jung, Y., & Lee, J. (2006). Context-based cooperation architecture for ubiquitous environment. *Ubiquitous Computing Systems*, 171–182. doi:10.1007/11890348_14

Kulkarni, D., Ahmed, T., & Tripathi, A. (2012). A generative programming framework for context-aware CSCW applications. *ACM Transactions on Software Engineering and Methodology*, *21*(2), 1–35. doi:10.1145/2089116.2089121

Kumar, M., Shirazi, B. A., Das, S. K., Sung, B. Y., Levine, D., & Singhal, M. (2003). PICO: A middleware framework for pervasive computing. *IEEE Pervasive Computing / IEEE Computer Society [and] IEEE Communications Society*, *2*(3), 72–79. doi:10.1109/MPRV.2003.1228529

Liampotis, N., Kalatzis, N., Roussaki, I., Kosmides, P., Papaioannou, I., & Sykas, E., … Xynogalas, S. (2012). Addressing the context-awareness requirements in personal smart spaces. In *Proceedings of the 2012 IEEE Asia-Pacific Services Computing Conference* (pp. 281–285). IEEE. doi:doi:10.1109/APSCC.2012.65 doi:10.1109/APSCC.2012.65

Lima, C., Gomes, D., & Aguiar, R. (2012). Pervasive CSCW for smart spaces communities. In *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, (pp. 118–123). IEEE. doi:10.1109/PerComW.2012.6197462

Lima, C. V., Antunes, M., Gomes, D., & Aguiar, R. (2013). A context-aware framework for CSCW applications in enterprise environments. In *Proceedings of the IADIS International Conference Web Based Communities and Social Media 2013, Proceedings of the IADIS International Conference Collaborative Technologies 2013* (pp. 11–18).

Meyer, E. M., Wichmann, D., Büsch, H., & Boll, S. (2011). Supporting mobile collaboration in spatially distributed workgroups with digital interactive maps. *Mobile Networks and Applications*, *17*(3), 365–375. doi:10.1007/s11036-010-0288-9

Neo4j. (2014). Retrieved January 27th, 2014, from http://www. neo4j.org/

Openfire. (2014). Retrieved January 27th, 2014, from http://www.igniterealtime.org/projects/openfire/

OSGi. (2014). Retrieved January 27th, 2014, from http://www. osgi.org/

Prinz, W., Loh, H., Pallot, M., Schaffers, H., Skarmeta, A., & Decker, S. (2006). ECOSPACE -- Towards an integrated collaboration space for eProfessionals. In *Proceedings of the 2006 International Conference on Collaborative Computing: Networking, Applications and Worksharing* (pp. 1–7). IEEE. doi:10.1109/COLCOM.2006.361868

Sahoo, P., Soltani, S., & Wong, A. K. (1988). A survey of thresholding techniques. *Computer Vision Graphics and Image Processing*, *41*(2), 233–260. doi:10.1016/0734-189X(88)90022-9

Smirnov, A., Shilov, N., Krizhanovsky, A., Lappetelainen, A., Oliver, I., & Boldyrev, S. (2008). Efficient distributed information management in smart spaces. In *Proceedings of the 2008 Third International Conference on Digital Information Management* (pp. 483–488). IEEE. doi:10.1109/ICDIM.2008.4746699

SOCIETIES. (2014). *Self-orchestrating community ambient intelligence spaces*. Retrieved January 27th, 2014, from http://www.ict-societies.eu/

SOCIETIES Third party services. (2014). Retrieved January 27th, 2014, https://github.com/societies/SOCIETIES-SCE-Services

Synonyms. (2014). Retrieved January 27th, 2014, from http://www.synomys.net/

Taylor, N., Robertson, P., Farshchian, B., Doolin, K., Roussaki, I., & Marshall, L. et al. (2011). Pervasive computing in daidalos. *IEEE Pervasive Computing / IEEE Computer Society [and] IEEE Communications Society*, *10*(1), 74–81. doi:10.1109/MPRV.2010.24

Temperini, M., & Nanni, U. (2012). eLearning for knowledge management in collaborative architectural design. *International Journal of Design Sciences and Technology, 19*(2), 105–122. ISSN 1630-7267.

Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity*. Cambridge University Press. doi:10.1017/CBO9780511803932

Christopher Lima obtained his MSc degree in Computer Science from the Federal University of Santa Catarina, Brazil, in 2007. Currently he is pursuing a PhD in the Department of Electronics, Telecommunications and Informatics at the University of Aveiro, Portugal. He is a researcher at the Institute of Telecommunications in European projects related to pervasive computing and collaborative applications such as FP7 PERSIST (PERsonal Self-Improving SmarT spaces) and FP7 SOCIETIES (Self Orchestrating AmbiEnT InteligEnce Spaces).

Mário Antunes obtained his MSc degree in Computer and Telematics Engineering from University of Aveiro. Currently he is pursuing a PhD in computer science under a joint program organised by Universities from Minho, Aveiro and Porto. In the past his work focused on robotic vision, specifically semantic vision, object detection and object recognition. As a result of this research, he participated in two major robotic competitions, SRVC 2009 and Robocup 2011. Currently, he is a researcher at the Institute of Telecommunications. His present research focuses on Machine-2-Machine communication, Internet of Things context-aware platforms, context inference and information retrieval.

Diogo Gomes graduated in Computers and Telematics Engineering from the University of Aveiro in 2003 with first class honours, and concluded his PhD by the same University on Resource Optimization for Broadcast Networks in 2009. He's currently an Invited Auxiliar Professor at the University of Aveiro. Since his graduation year, he has participated in several IST Projects such as IST-Mobydick, IST-Daidalos, IST-Akogrimo, IST-C-MOBILE and ICT-C-Cast where besides conducting research on QoS, IP Mobility and Multicast/Broadcast has always been deeply involved in the deployment prototypes demonstrations. He was one of the founders of the University Linux User Group and has contributed to several OSS projects. Recently he participates in FP7 ICT-MEDIEVAL and in Industrial Partnership with Portugal Telecom Inovação on Machine-2-Machine Communication and Information retrieval from sensor networks.

Rui Aguiar received a PhD degree in electrical engineering in 2001 from the University of Aveiro. He is currently an associate professor with "Agregação" at the University of Aveiro and an adjunct professor at the INI, Carnegie Mellon University. He is leading a research team at the Institute of Telecommunications, Aveiro, on next-generation network architectures and protocols. His current research interests are centered on the implementation of advanced wireless networks, systems, and circuits, with special emphasis on QoS and mobility aspects. He has more than 300 published papers in those areas. He has served as Technical and General Chair of several conferences, such as ICNS '05, ICT '06, and ISCC '07, and is Associate Editor of several journals. He is a senior member of IEEE and member of ACM.

Telma Mota, has a degree in Electrical and Computing Engineer and a master degree in Telecommunications both from the Engineer University of Porto. In 1994, she joined Portugal Telecom Inovação where she got expertise in the most relevant service and network architectures such as IN, IN evolution, Parlay, 3GPP IMS, TISPAN, MBMS and more recently OMA, SOA, Web 2.0. Since then, she has been responsible for PTIn participation in several ACTS/IST and EURESCOM projects. Currently, she is the head of the "Platforms and Multiservice Networks" group at PTIN and her main area of interest is service architectures, smart and pervasive environments, social networks and context awareness.