Design and Implementation of a Socially-Enhanced Pervasive Middleware

Gabriella Castelli, Alberto Rosi, Franco Zambonelli
Dipartimento di Scienze e Metodi dell'Ingegneria
Universitá di Modena e Reggio Emilia, Italy
Email: name.surname@unimore.it

***Abstract*—Middleware infrastructures for pervasive computing, in order to be able to support services and users activities, have to deal with both spatially-situated and socially-situated interactions. In this paper we present the solution adopted in the SAPERE middleware that exploits the graph of a social networks, and combines it with relations deriving from spatial proximity, to drive the topology of interactions among users, devices and services. This results in a middleware that facilitates the development and management of services that are adaptive to both spatial and social concerns, and can support effective service discovery and orchestration, and naturally tackles privacy issues.**

***Keywords*-pervasive middleware; social interaction; proximity**

I. Introduction

Portable pervasive devices such as smart phones and tabs are already pervading our life and leading the increasing diffusion of location-based services. Yet, once smart phones and their associated sensors and services will be made able to opportunistically connect with each other, and possibly with environmental sensors and public displays, the overall resulting distributed infrastructure will promote the deployment of a richer variety of innovative pervasive (i.e., mobile and context-aware) services. These will span from services for augmented interactions with the physical and social worlds [1] to socially-intelligent services for collective perception and action [2].

In this vision, the pervasive infrastructure will become as participatory and capable of value co-creation as the Web today, and the users will be expected to contribute with data, service components, sensing and actuating devices [3], other than with their specific human capabilities [4]. At the same time, though, the openness and dynamics of the scenario let a key general issue arising: according to which strategy should distributed service components and devices be allowed to connect and interact with each other towards the provisioning of distributed pervasive services?

Future pervasive services will likely deal with the spatially-situated activities of users in some environments, as well as location activities do now. Accordingly, several middleware infrastructures for pervasive scenarios propose exploiting spatial proximity between devices or services as the key enabler for interactions [5], [6]. This approach definitely suits the spatially-aware nature of several pervasive services, but misses in accounting for the social relations between components: why and to what extent should a user share services and devices with nearby foreigners?

To tackle the problem, we propose combining together the awareness of spatial relations between users, devices and services (as deriving from physical proximity) with the awareness of social relations as expressed in social networking platforms. In particular our proposal relies on the following design assumptions:

- Physical proximity between the nodes of the pervasive infrastructure (smart phones, environmental sensors, or ambient servers) defines the basic topology of the network, and enables communication among devices;
- The nodes of the infrastructure are associated to a profile in a social networking platform (e.g., Facebook), typically corresponding to the existing profiles of the owning users and institutions;
- For any two nodes, the mutual discovery and composition of their services and the sharing of data are enabled by physical proximity and subjected to the existence of a social relations between the twos.

In this paper we show how we implemented this approach within the EU project SAPERE (www.sapere-project.eu). However the proposed approach can be applied to any pervasive middleware infrastructure, and it is indeed very effective to harness the power of social networks in the context of spatially situated pervasive services. In fact: it makes possible to promote and control adaptive interactions taking into account both spatial and social concerns; it facilitates service discovery and composition relying on the social relations between services; it makes it possible to express complex spatial relations between nodes by acting on the social network graph; finally, it enables handling and controlling privacy issues very effectively.

II. The SAPERE Middleware Architectures

SAPERE tries to tackle the issues of the dynamics and decentralization of future pervasive networks modeling the overall world of services, data, and devices as a sort of distributed and spatially-situated computational *ecosystem*.

In SAPERE the pervasive service environment is modeled as a *spatial substrate*, laid above the actual pervasive network infrastructure. Such a spatial substrate embeds the so-called *eco-laws*, basic laws that rule how the components

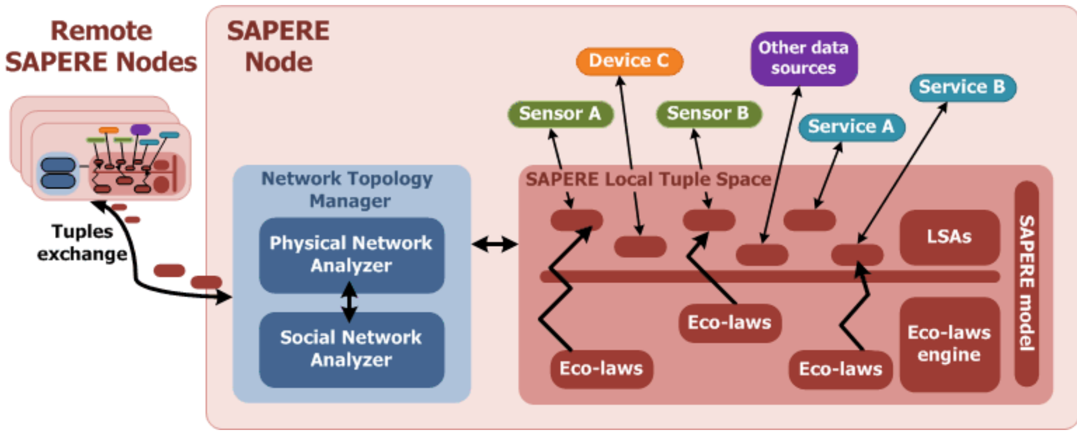

Figure 1. A local SAPERE node and its internal components.

of the ecosystem interact and combine with each other in a self-organized fashion. As a result the components are able to serve their own individual needs as well as assuring the the sustainability of the overall ecology. In order to realize such a spatial substrate, SAPERE relies on a distributed implementation made up of a number of distributed engines allocated on the nodes of the infrastructure (see Figure 1), each realized as a reactive tuple space [7], i.e., a tuple space with programmable pattern-marching mechanisms and supporting subscriptions to local events. Local application services on a node exploit such local engine for the support of their interactions. Semi-centralized implementations associate such a node to a server at a specific spatial location (e.g., in a smart room or by an interactive public display), letting mobile nodes and the application services on such nodes access and exploit such servers for their interactions. SAPERE considers that tuple spaces exist both on mobile nodes and on infrastructural nodes, with the idea of supporting both ad-hoc interactions and server-mediated ones.

For each component (whether services, devices, or data sources), SAPERE adopts a common modeling and treatment. Each of them has an associated self-descriptive semantic representation, called *LSA* (Live Semantics Annotations), which is a basic ingredient for enabling dynamic unsupervised interactions between components mediated by the tuple spaces. To account for the high dynamics of the components represented, such annotations are defined as living, active entities, tightly associated to the component they describe, and capable of reflecting their current situation and context. Local services and components refer to a specific node, where local interactions take place by storing, reading, and extracting *LSAs* from the local tuple space. In addition to the information space, on each node the tuple space engine includes also local eco-laws that are in charge, based on the local information and events, of triggering the necessary actions for the composition of services and for supporting their interactions indirectly with each other via tuple exchange and synchronization over tuple spaces events. According to the ecosystem vision, the eco-laws take the form of virtual *chemical reactions* among the *LSAs*. Such reactions will contribute establishing virtual chemical bonds between entities as well as producing new components.

A specific set of processes, named Network Topology Manager, takes care of managing the interactions with the other nodes of the network, e.g., for distributing information and events across "neighbor" nodes, or for enabling inter-node service and device discovery and interactions. For fully-distributed implementation, such distributed interactions occur in a sort of dynamically shaped ad-hoc network according to a field-based coordination model [5]. For semi-distributed implementations, the local servers typically interact with each other according to some logical or spatial relations.

III. Towards Socially-aware Networking

A key issue in the realization of the SAPERE middleware design is defining strategies for dynamically shaping the network of nodes, i.e., to determine which nodes are neighbors with each other at a given time and, consequently, which services and devices are allowed to interact.

To preserve generality, we have not hardwired any specific strategy for the shaping of such network. Rather, we have modularized the engine so as to make network topology managers configurable.

A. Spatial Strategies

A spatial policy that network topology managers can support is the *network-driven* one: shaping the neighborhood relations between nodes based on the existence of wireless connections (i.e., Bluetooth or Wifi) between them, quite reasonably assuming that wireless network proximity reflects spatial proximity.

This solution is hardly acceptable as a general approach due to privacy issues. For instance, a visitor of the exhibition center would be forced to share opinions or location with all close visitors, there included strangers. Furthermore, it requires some specific a priori configuration efforts to distinguish between mobile nodes (associated to users) and infrastructural nodes (to be associated to locations). For instance, while users may not wish to share current and past locations with other users indiscriminately, they may accept sharing them with pervasive displays, to enable crowd guidance services. As an additional problem, a mere network-driven can sometimes introduce mismatches between physical and logical proximity (e.g., two visitors can be physically very close but in different thematic areas of the exhibition).

B. Socially-Aware Strategies

To account for social, other than spatial interactions, several proposals have emerged in the past few years (see Section VII). On the one side, some proposals [8], [9] infer the social dimension from spatial interactions. On the opposite side, other proposals such as [10] suggest fully relying on social networks forgetting physical and logical proximity.

The solution implemented in SAPERE, exportable to other middleware infrastructure, tries to get the best from such opposite approaches. In particular, it integrates the physical proximity concept of network-driven strategies with social relations extrapolated by social networks. That is, given the possibility for nodes to access the graph of some existing social networks: *"devices and services on a node can interact with those of another node only if the two nodes are close according to both the network and the social network viewpoint"*.

Comparing this strategy to the network-driven one:

- The spatial-awareness is preserved, but no longer requiring complex mining of spatial interactions to acquire some degrees of social-awareness;
- Social networks are be considered *by definition* up-to-date, since they are updated by users based on their willingness to share information, which is very important to effectively support adaptivity;
- Relations in a social network are typically peer-based (friendship between two users), which perfectly suits fully-distributed middleware solutions and ad-hoc interactions;
- Other interactions can be modeled associating the concept of group in a social network to logical and/or physical space by means of infrastructural servers that may be possibly associated to that place (e.g., a interacting public display in that room).

IV. Shaping Socially Aware Interactions

Let us now get into details about the solution actually implemented in SAPERE middleware, and about how interactions take place in it. The basic assumption is that all SAPERE nodes are associated to a profile in a social network, whether an individual user profile (for personal mobile phones) or a group one (for infrastructural nodes).

For the sake of simplicity, we will refer in the following to Facebook and to its specific terminology, as this is the social network we have considered in our first implementation. In any case, the concepts behind are of a more general nature and can be applied to other classes of social networks (e.g., Google+ or LinkedIn).

From an implementation viewpoint, to promote the integration of Facebook with the SAPERE middleware, all that is necessary is to let network topology managers access information about the Facebook social graph (via the Facebook Developers API), configure their actions (i.e., connecting with neighbor nodes and propagating tuples) accordingly to the retrieved information.

A. Smart phones and Ad-hoc Interactions

For smart phones, the many embedded sensing devices on the phone, the data entered by the user, as well as those software agents/services/applications executing on the phone, each has: *(i)* a self-descriptive tuple associated to the local tuple space, in which they can also publish the results of their computations and can query for tuples produced by other services/devices; *(ii)* a corresponding description in the "About me" section of the Facebook profile, associated with a set of sharing permissions.

Sharing permissions determine to which nodes the tuples of a given service/device should be propagated, so as to enable tuple-based distributed interactions. For instance, a user can decide to share with friends of friends its current location, so as to allow being being invited to lunch by them.

The network topology manager on a smart phone continuously verifies the state of the wireless connections and of the social network relations, along with the specified sharing permissions. As from Figure 2-left, when the topology manager on a node perceives it is in wireless range with some other node, it accesses Facebook to verify the existence of a social relationship between the owners of the two nodes. Depending on such relations, and on the sharing permissions, tuple propagation between these two nodes can start and lead to compose and orchestrate the activities of the distributed services and devices on such mobile phones.

B. Infrastructural Nodes and User-to-Infrastructure Interactions

Infrastructural nodes, as interactive public displays, are assumed to have a group profile on the social network (as in Figure 2-left). In social networks, the idea of group typically corresponds to some spatially-unrelated notion. However, for the kinds of pervasive services we are interested in, a group can represent the logical space in which the infrastructural node is located. Individual users (and their mobile phones)

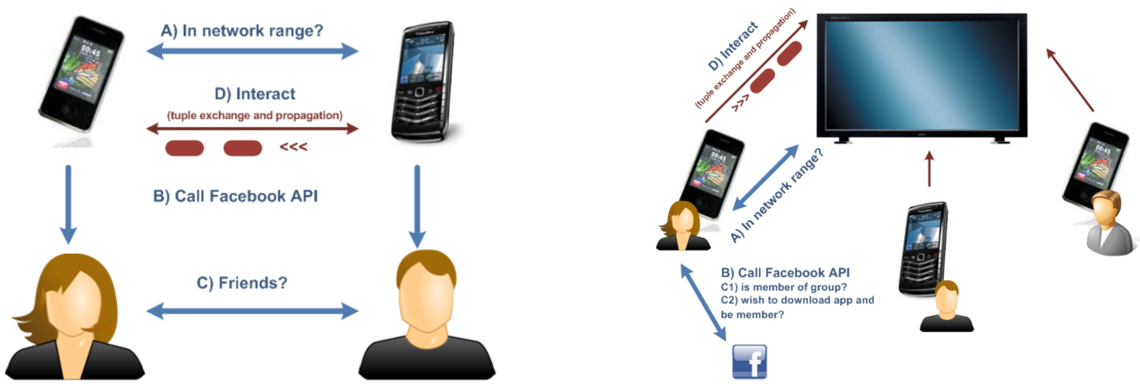

Figure 2. (up) Enabling ad-hoc interactions between friends and (bottom) with a group.

can subscribe to the group to participate to the local ambient services promoted by such infrastructural node.

In SAPERE, the local tuple space of an infrastructural node acts indeed as a sort of shared wall for all the members of the group. Once a user connects with some infrastructural node, the network topology manager verifies her membership to the group. Moreover, it is also possible for a user to be asked to join the group dynamically, and to dynamically download – if one is not already installed – the specific service components/apps that will enable the user to exploit at the best the public services provided in that location. It is generally up to users to decide which tuples (and thus which services and devices) to share with whom. However, the settings of a group can require users to share specific tuples in order to be part of the group and thus take advantage of the provided services.

C. Infrastructure-to-Infrastructure Interactions

To promote the exploitation of logical spatial concepts in pervasive services, we consider that groups, at their turn, can subscribe to each other. That is, a group can become member of another group. This makes it possible to define relations between infrastructural SAPERE nodes, and to reflect spatial relations between the location they represent (e.g., for two confining rooms in the exhibition center, the respective SAPERE nodes will belong to each others' group to express such logical neighborhood concept).

As for interactions between infrastructural nodes, these again rely on tuples propagation and inter-node coordination for those nodes whose groups are members of each other.

V. The implementation

The current implementation of the SAPERE middleware supports Android being lightweight enough to be hosted even in smart phones and tablets. As shown in Figure 1, the SAPERE middleware is composed by two main parts: the Local Tuple Space and the Network Topology Manager.

A. The Local Tuple Space

The Local SAPERE Space is the core node and implements the SAPERE model basic elements: *LSAs* and eco-laws. It is made up of:

- The local *LSAs* collection that stores the *LSAs* generated by the local node or injected by remote SAPERE nodes, among which the eco-laws are executed;
- The model definition that implements the basic interfaces for the SAPERE model: *LSAs*, eco-laws and their inner components;
- The Eco-laws engine, based on TUCSON [7] at the moment. The current engine enables the execution of eco-laws in terms of programmable TUCSON reactions, and the distribution of *LSAs* to remote SAPERE nodes. In order to keep the node lightweight, *LSAs* distribution is implemented using Java sockets. However, a new tuple space engine, specifically conceived for the SAPERE model, is under implementation.

The Local SAPERE Space has been designed in modular way, which makes it easy to add custom extensions changing the tuple space engine, or the underlying model. A Local Space is sufficient to implement applications that run on the single device that hosts the space and exploits only local *LSAs* and eco-laws. A network of SAPERE nodes must be established to allow distributed applications to be executed.

B. Network Topology Manager

The Network Topology Manager enables the networking between SAPERE nodes. It is a process, configurable at launch time, that launches two main activities:

- The Network Analyzer that is in charge of periodically accessing the physical network and finding close devices;
- The Social Network Analyzer that is in charge of periodically accessing the social network and downloading the social graph;

Upon changes in the representation of the physical or social context of the device, the Network Topology Manager com-

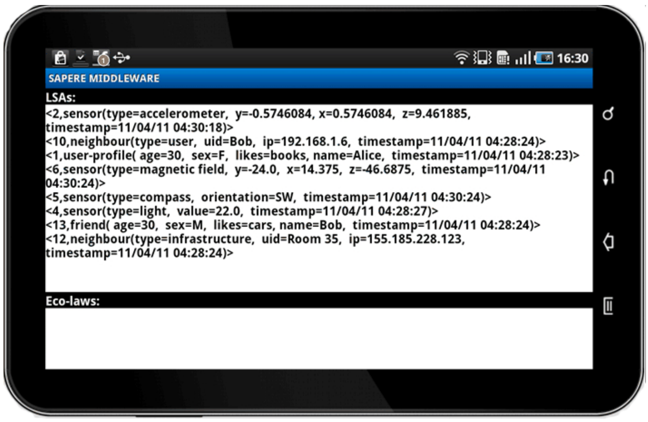

Figure 3. Screen shot of the middleware output in a Galaxy Tab Android device.

bines the results from the Network Analyzer with the output of the Social Network Analyzer, and makes eventually the local node aware of other nodes triggering the injection of proper *LSAs* in their remote tuple spaces. Network manager processes currently support Wi-Fi and Bluetooth for the physical network, while for the social network we currently adopt the Facebook API (developers.facebook.com).

The SAPERE middleware is an Android app that can be launched by app launcher. In Figure 3 a screen-shot of the output of the middleware running on a Samsung Galaxy Tab is depicted. The picture shows the *LSAs* stored in the device. Each *LSA* has a unique id that is automatically associated by the local tuple space engine during the injection, a qualifier that describes the type of the entity represented (e.g., user-profile, sensor, friend, etc.), and possibly meta-data managed directly by the middleware, such as the timestamp. The *user-profile LSA* contains the profile of the owner of the device. The *sensor LSAs* contain sensorial data coming from sensors embedded in the device, they are periodically updated to reflect the current reading of the sensor. *Neighbor LSAs* are injected into the local tuple space by the Network Topology Manager when a device that is close in the physical network, and is also part of social network, is detected. As shown, neighbors are typed either as "user" or "infrastructure" depending on their profile in the social network. Once the presence of a neighbor user is notified by the proper *LSA*, the node will spread the node-owner profile information to the neighbor in order to possibly enable further interactions. In the picture we can see that a *friend LSA* has been spread to the device and contains the profile of the user Bob.

VI. Discussion

The proposed solution exhibits a number of advantages. First, it allows to develop services that are at the same time adaptive to both the concerns. Also, by acting on social graphs, those interactions can be controlled and shaped on-the-fly, without requiring re-configuration of the services.

Second, service discovery and composition is made more effective due to the possibility of selecting services depending on their social relations, other than simply in terms of physical proximity. This is very important in densely populated environment.

Third, the approach makes it possible to express higher-level spatial relations than simply metric ones (e.g., spatial proximity), by exploiting the social network graph (and in particular relations between groups) to represent logical spatial concepts and by having spatial coordination activities being shaped by such logical spatial relations.

Fourth, by acting on the social network profile, the handling of privacy issues by users and ambient administrators can be notably facilitated, made more transparent, and more controllable.

However, the proposed approach has also some current limitations that we are currently trying to address. First, other than being quite slow, existing social networks miss the possibility to effectively associate devices and services to the user profile.

Second, they will have to account for much higher-dynamics in social relations introduced by pervasive services.

VII. Related Work

The idea that social aspects and social awareness are highly relevant for the effective realization of pervasive computing services is not new [8], [11]. However, several proposals attack the problem only from the perspective of exploiting pervasive computing devices to infer social relationships extrapolating from proximity interactions [12], [9], [6]. Rather, we think that pervasive services can be notably improved by exploiting the large body of social awareness already embedded in social networking tools.

In the area of service-oriented computing, it is getting recognized that social networking at the level of services can notably facilitate service discovery and composition [13], [14]. For instance, in the LinkedWS proposal [15], a social network of services is dynamically built by analyzing the patterns of co-invocation and similarity, and defining the social relationships between services accordingly. Our proposal commits to this idea, but instantiates it to the specific scenario of pervasive computing services and devices.

An interesting propolas is presented in [16], where a framework to model social context and exploit it for smart home environment is presented. We share with the authors the central role of social aspects in everyday pervasive world, but we bring it at the very core of our approach exploiting social graphs to actually shape pervasive networks.

The proposal that more closely relates to our is that of "Social Computing Network" (SCN) [17]. In SCN, the idea is that the awareness hidden in social networks about "who trust who" can be effectively exploited to promote discovery and sharing among personal sensors. SCN defines an extended social network concept that, for each users, includes a "circle" of personal devices and services of the user. On this basis, a user can specify which of its own resources to share with friends or groups. Our proposal shares many of the key assumptions of SCN, but goes further by suggesting that the social network has not only to promote discovery and composition but also to define the very structure of space in which interaction takes place.

VIII. Conclusions and Open Challenges

In this paper we have presented the approach that we have implemented in the SAPERE middleware to promote socially-enhanced spatial interactions in pervasive computing services. The basic idea is to integrate the physical space of interactions with a social networking graph, so as to facilitate the autonomous and adaptive activities of pervasive services while accounting for social and spatial issues at the same time.

Our current work includes performing tests related to the overhead of our early implementation and developing several applications to challenge the idea in the real-world. In addition, we are planning to extend our approach to include social recommendations in service composition [18] and to leverage users' posts as sensing devices [19].

Acknowledgements: Work supported by the SAPERE project (EU FP7-FET, Contract No. 256873).

References

[1] B. Coleman, "Using sensor inputs to affect virtual and real environments," *IEEE Pervasive Computing*, vol. 8, no. 3, pp. 16–23, 2009.

[2] A. Schmidt, M. Langheinrich, and K. Kersting, "Perception beyond the here and now," *Computer*, vol. 44, pp. 86–88, 2011.

[3] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn, "The rise of people-centric sensing," *IEEE Internet Computing*, vol. 12, no. 4, 2008.

[4] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the world-wide web," *Communications of the ACM*, vol. 54, pp. 86–96, April 2011.

[5] M. Mamei and F. Zambonelli, "Programming pervasive and mobile computing applications: the tota approach," *ACM Trans. Software Engineering and Methodology*, vol. 18, no. 4, 2009.

[6] C. Boldrini, M. Conti, F. Delmastro, and A. Passarella, "Context- and social-aware middleware for opportunistic networks," *Journal of Network and Computer Applications*, vol. 33, no. 5, pp. 525–541, 2010.

[7] M. Viroli, M. Casadei, S. Montagna, and F. Zambonelli, "Spatial coordination of pervasive services through chemical-inspired tuple spaces," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 6, no. 2, p. 14, 2011.

[8] P. Markopoulos, "Awareness systems and the role of social intelligence," *Artificial Intelligence and Society*, vol. 24, pp. 115–122, 2009.

[9] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: Social-based forwarding in delay tolerant networks," *IEEE Transactions on Mobile Computing*, 2011.

[10] M. A. Rahman, A. E. Saddik, and W. Gueaieb, "Building dynamic social network from sensory data feed," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 5, pp. 1327 –1341, May 2010.

[11] M. Endler, A. Skyrme, D. Schuster, and T. Springer, "Defining situated social context for pervasive social computing," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, march 2011, pp. 519 –524.

[12] C. Borcea, A. Gupta, A. Kalra, Q. Jones, and L. Iftode, "The mobisoc middleware for mobile social computing: challenges, design, and early experiences," in *1st international conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications*. ICST, Brussels (B), 2008.

[13] X. Xie, B. Du, and Z. Zhang, "Semantic service composition based on social network," in *17th International World Wide Web Conferece, Beijing (CH)*, 2008.

[14] Z. Maamar, H. Hacid, and M. N. Huhns, "Why web services need social networks," *Internet Computing, IEEE*, vol. 15, no. 2, pp. 90 –94, march-april 2011.

[15] Z. Maamar, L. K. Wives, Y. Badr, S. Elnaffar, K. Boukadi, and N. Faci, "Linkedws: A novel web services discovery model based on the metaphor of "social networks"," *Simulation Modelling Practice and Theory*, vol. 19, no. 1, pp. 121–132, 2011.

[16] G. Biamino, "Modeling social contexts for pervasive computing environments," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, march 2011, pp. 415 –420.

[17] N. Esfahani and S. Malek, "Social computing networks: a new paradigm for engineering self-adaptive pervasive software systems," in *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering, Cape Town, South Africa*. ACM, May 2010, pp. 159–162.

[18] G. Castelli, M. Mamei, and F. Zambonelli, "The changing role of pervasive middleware: From discovery and orchestration to recommendation and planning," in *PerWare Workshop at the 9th IEEE International Conference on Pervasive Computing and Communications, Seattle (WAS)*, March 2011, pp. 214–219.

[19] A. Rosi, M. Mamei, F. Zambonelli, S. Dobson, G. Stevenson, and J. Ye, "Social sensors and pervasive services: Approaches and perspectives," in *PerCol Workshop at the 9th IEEE International Conference on Pervasive Computing and Communications, Seattle (WA)*, March 2011, pp. 525–530.