Activity-centered ubiquitous computing support to localized activities

Helder Pinto and Rui José

University of Minho, Information Systems Department
Campus de Azurém, 4800-058 Guimarães, Portugal
{helder,rui}@dsi.uminho.pt

Abstract. This paper presents ActivitySpot, a ubiquitous computing framework for supporting localized activities, i.e., activities strongly associated to a specific physical environment, performed by occasional visitors. The ActivitySpot framework implements an activity-centered approach to ubiquitous computing, by defining a conceptual model inspired by Activity Theory and implementing a software infrastructure derived from this conceptual model. ActivitySpot has been evaluated by experiments run at different public spaces and results demonstrate the framework's suitability to the targeted type of environment.

1 Introduction

The user interaction model used in many computing prototypes has been based on the application- or service-centered paradigms. We share the conviction, however, that an approach centered on the user, particularly on the user activity, is a promising path for bringing computing closer to people and to transparently support activities that take place in the physical world [1–4]. Taking an activity-centered approach for system design becomes especially important in situations in which people have little or no prior knowledge about the physical environment or about the local means available for the activity they are going to perform. This is the case of localized activities, i.e., activities that are strongly related to a specific physical environment and that can only be achieved there (e.g., visiting an exhibition at a museum, visiting a relative at the hospital, etc.). Most of these activities are performed at public spaces by occasional visitors who are not used to the environment nor to local information services that may be available.

This work investigates the use of an activity-centered approach to the design and development of ubiquitous computing support to localized activities performed by occasional visitors to public spaces. From the wide spectrum of design and implementation challenges posed by this overall objective, our work focuses on the following:

– **Activity model**. Representing how humans perform an activity is a difficult task, as people may have different mental models of the same activity. Furthermore, human activity is unpredictable and it may be risky to rigidly formalize the steps that compose it [5]. A generic model of human activity

should thus focus on what is less dependent of individual mind-sets: what the environment affords people to do, i.e., which functionalities, interaction media, and resources are available to each activity. Representing this model in a machine-understandable manner also requires it to be simple enough to require minimum specific know-how and effort from public space administrators managing ubiquitous computing support to localized activities.
- **User interaction model**. Occasional visitors to public spaces do not have time to learn how to use a probably unknown ubiquitous computing system. Therefore, user interactions must be very simple and usage instructions must be blended with the environment and the system itself. Moreover, a ubiquitous computing environment is potentially instrumented with heterogeneous interaction devices and is used by visitors who bring their own personal devices. The challenge here is how to deal with this heterogeneity while not compromising the simplicity of user interaction. Another challenge is to deal with the possibly varied interaction devices the same person may use within the course of an activity and to make that person feel that all interactions, whatever device is used, are integrated and all part of the same activity.
- **Personalization**. Providing a personalized experience to occasional visitors is a specially challenging issue to ubiquitous computing systems. Since users may have not previously visited the space, we cannot assume the existence of a local personal profile or information about a particular user and her/his resources. Accessing the users' personal domain (profile, personal equipment and resources, context, etc.) poses several issues: availability and disclosure of the personal domain; agreed mechanisms for the association between the personal domain and the local infrastructure; authorization and authentication; and, inevitably, privacy concerns. Given the difficulty of tackling such a wide and multi-disciplinary problem, this work investigates simple mechanisms allowing visitors to make their personal domain, or at least part of it, available to the local infrastructure.

In the following section, we present Activity Theory, the theoretical framework upon which we ground our conceptual model for representing activities and user interaction. Section 3 describes this conceptual model and presents ActivitySpot, the activity-centered software framework we propose for supporting localized activities. Section 4 reports two field experiments we carried out with the ActivitySpot framework and discusses their results. Finally, Sect. 5 presents some of the related research and Sect. 6 concludes the paper.

2 Modelling activity and user interaction

We believe that the best approach to overcome the activity and user interaction modelling challenges is to ground our research on previous work on human activity analysis. The importance of a theoretical framework of human activity is that it provides ubiquitous computing researchers with an agreed set of terms to describe activity and with concepts that drive them in the construction of systems that intend to support activity. Among several frameworks produced mainly by the fields of psychology and philosophy, we chose Activity Theory [6] as the background for this work, based on its maturity acquired by several decades of research and its set of simple and solid concepts. Among these concepts, we are particularly interested in the different levels of analysis of an activity: *activities*, at the uppermost level, are distinguished on the basis of their motive and the object toward which they are oriented; *actions* are distinguished on the basis of their goals; and, finally, *operations*, on the basis of the conditions under which they are carried out. For example, an activity motivated by food is composed of several goal-oriented actions (e.g., collecting ingredients, preparing a recipe, etc.) and operations which vary in function of conditions (e.g., going to the kitchen-garden, picking vegetables, taking ingredients from the fridge, etc.).

An activity may be carried out in a variety of ways by employing different actions and operations, which may respectively be part of different activities and actions. Individual characteristics and changing local and personal context are the factors driving the structure of a localized activity. For example, a public space like a museum may support different activities, which in turn may employ different actions and operations, all depending on several factors, like the visitor role (e.g., regular museum visitors, authors, external security inspectors, etc.), age, preferences, available resources, or context. In a ubiquitous computing environment supporting these hypothetical localized activities, each ubiquitous computing device can be seen as a tool that may be used for the execution of one or more operations. Figure 1 depicts our Activity-Theory-inspired abstract model of activity applied to the museum example.

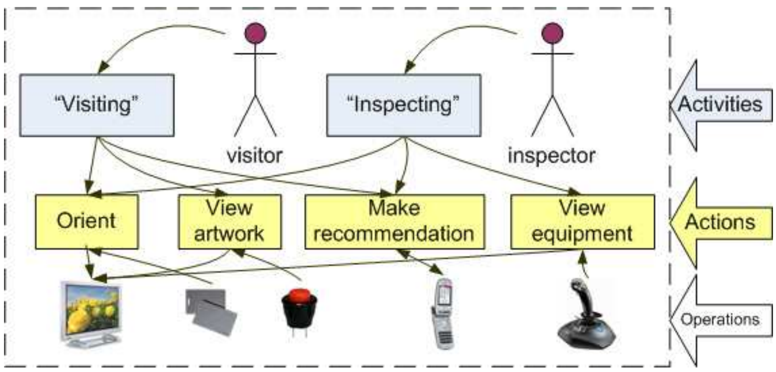

Fig. 1. Example of an activity-based model for a ubiquitous computing environment

For the sake of clarity, the model omits the details of operations. In a ubiquitous computing system, an operation can be a user interaction, a sensor read, a web-service request, a database query, etc. We just represent the user-facing devices, which are the most visible part of operations. The model exemplifies how flexible an activity structure can be: a plasma screen can be used both by visitors and inspectors to achieve different actions; an "orient" or a "make recommendation" action can be executed in different activities, with different goals in mind (a recommendation made by a visitor has a different goal from a recommendation made by an inspector).

Given that user interaction with a ubiquitous computing system is done through multiple, heterogeneous means, we reduced user interaction analysis to basic human-computer interaction concepts: stimulus and response. We assume that, for a given stimulus through a given device, a response is produced, synchronously or not, through the same device or through other device or set of devices. We also assume that people interact with ubiquitous computing systems mainly through simple devices. We consider a simple interaction device in a ubiquitous computing environment as being the equivalent of a mouse, a keyboard, or a screen in a desktop computer. We are talking about elementary, easy-to-use interaction means that cannot be used only by themselves to carry out an activity. The execution of an activity is thus distributed by the interactions made with each of those devices. Every user interaction, whatever the underlying medium, is framed within the user activity and is integrated with other previous and further interactions, becoming more meaningful and contributing to compose the whole activity.

3 The ActivitySpot framework

The ActivitySpot framework provides a set of conceptual and software tools for designers and developers applying an activity-based approach for supporting occasional visitors to ubiquitous computing environments. The concepts basing the framework are derived from Activity Theory, namely those associated to the activity structure analysis. Therefore, the principles of the ActivitySpot framework originated from the concepts of activity, action, and operation, as well as the activity structure hierarchy and flexibility. We also consider that activities or actions depend on local and personal context, either as an execution condition or as a variable influencing the response of an operation. Finally, the framework includes the basic concepts of stimulus and response to model user interaction. The conceptual model is implemented in the architecture described in Sect. 3.2.

The main strength of this conceptual model, besides its activity-centered character, is its simplicity. Such a simple model leads to simple architectural abstractions and, therefore, to an easier task for public space administrators using ActivitySpot for supporting localized activities.

3.1 Environment specification

In order to be independent of physical space and activities and thus support any localized activity scenario, the ActivitySpot framework is based on a generic specification format for activities, actions, and interaction devices available in an environment. Each environment supported by ActivitySpot has a specification of: a) which actions can be executed – name, supported stimulus and response types, a reference to the component implementing the action controller, and execution conditions; b) which activities are available – name, execution conditions, and references to the actions composing it; and c) which local devices can be used – stimulus or response type, physical location, and references to other devices which have some physical or logical association.

Activity specification is currently done by means of an XML document. Future developments of ActivitySpot will include a graphical user interface providing high-level abstractions easing the generation of the activity specification.

3.2 The ActivitySpot architecture

The ActivitySpot architecture (see Fig. 2) implements the activity-centered conceptual model described earlier. Following the generic character of the environment specification, the ActivitySpot architecture provides abstractions powerful enough to be instantiated in several concrete scenarios and simple enough to facilitate the adoption and usage by public space managers without requiring deep computer science know-how.

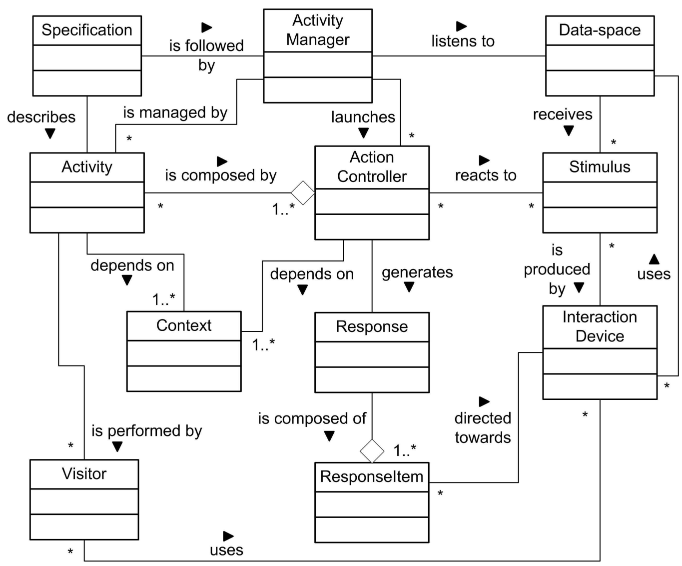

Fig. 2. The ActivitySpot architecture

The main architecture component is the Activity Manager. It manages activity execution by coordinating stimuli, execution conditions verification, and response generation. Another crucial component is the EQUIP [7] data-space, which is used as a communication middleware between interaction devices and Activity Manager. Whenever a visitor generates a stimulus through an interaction device, a corresponding stimulus description is sent to the data-space. The

Activity Manager senses this stimulus, identifies its author (e.g., a mobile phone number, a MAC address, an RFID code, etc.), and triggers all the action controllers that fill the execution conditions and that support the respective stimulus type. Each of these action controllers processes the stimulus sent by the Activity Manager and, in the case the stimulus was effectively targeted to the respective action, a corresponding response is produced – a response may be composed of one or several response items directed towards specific interaction devices. The Activity Manager sends the response items to the data-space, which propagates them to the interaction device presenting that response type.

4 Evaluation

Evaluation of the ActivitySpot framework is affected by the limitations inherent to infrastructure evaluations: it can only be evaluated in the context of use and thus must be evaluated indirectly through applications built on top of it. There is the danger of getting distracted by the demands of application development and to lose sight of the real purpose of the effort, which is purely to evaluate an infrastructure [8]. However, the evaluation of the ActivitySpot infrastructure is more centered on human aspects rather than technical ones. The evaluation model has to consider, e.g., the user perception regarding ActivitySpot's activity-centered character, user interaction model, and personalization. This thus makes our evaluation dependent on application usability and usefulness.

We considered some reference models for evaluating user perception and intention towards technology (Social Cognitive Theory [9] and UTAUT [10]) or how well an activity-centered system fits in the activity it is intended for [11]. Although none of these models by themselves provided the ideal set of constructs for evaluating ActivitySpot, they were a useful source for building our own evaluation system. The constructs that compose our evaluation system are the following: activity and action perception – the degree to which an individual perceives the system as a tool for performing an activity or executing an action; performance/usefulness – the degree to which an individual believes that using the system helps him or her to attain gains in job performance; facilitating conditions[1] – the degree to which the ubiquitous computing infrastructure facilitated the system usage; and personalization – the degree to which a system adapts its functionality (logic, contents, layout, etc.) to the user and the importance the user confers to it.

We tested ActivitySpot in two different scenarios and collected evaluation data from surveys and log analysis. We further describe each of the ActivitySpot installations and we conclude the section by discussing the evaluation results.

4.1 PhD poster session at the School of Engineering

The first experiment was conducted during a one-day PhD poster session in our university campus, with about 200 exhibited posters. We deployed assistance

[1] We use the same terminology of UTAUT's construct, but with an application that is specific to our scenario

for two different activities: visiting the poster session and presenting a poster. Although in both cases many users were university members or students, the scenario, as an extraordinary event, provoked the situation that characterizes our work: novelty of activity and physical setting. Both activities took place in the poster exhibition area. ActivitySpot provided both activities with assistance to view an exhibition plan, bookmark posters, publish comments, publish photographs, view the visit state, vote for a poster, and view poster status.

Users had to explicitly choose their activity by sending an initialization SMS message to the ActivitySpot message center. The interaction devices supported by ActivitySpot were SMS, public displays, Bluetooth/IrDA file transfer, and RFID tag readers. After the activity, users were able to interact with ActivitySpot through a Web interface, where they could access the details of their activity (e.g., bookmarked posters, published comments and photographs, etc.).

4.2 Vila Flor Cultural Center

The second evaluation scenario, a six week long experiment held at a cultural center, in Guimarães, aimed at assisting spectators at three different moments of the shows: before, at the interval, and afterwards. The activity was composed of actions allowing spectators to obtain detailed information about the current show, post comments and photographs, view information about next shows, vote for the current show, or view information about the activity state.

Since there were no simultaneous shows, ActivitySpot implicitly inferred the intended activity, i.e., the activity was automatically initialized for the current show after the first interaction made by the visitor. In addition to the interaction means available in the first experiment, 2D codes were also provided. Before trying ActivitySpot, visitors had to associate, at the registration desk, their personal devices (e.g., phone number, Bluetooth address, RFID tags) to their identity in the system. Moreover, 2D code users had to download (at the ActivitySpot's Web site or *in situ* over Bluetooth) a reader application. In between shows, users could obtain, through a Web interface, additional information about their activity in each of the attended shows and view posted photographs or the list of the most voted shows.

4.3 Analysis and discussion

Both experiments allowed us to demonstrate different technical capabilities provided by the ActivitySpot framework, such as the support to simultaneous different activities, coordination and integration of heterogeneous interaction means, action reuse in different activities, and implicit and explicit activity initialization. However, only the second experiment could provide us enough data for concluding about the validity of our approach, since we had a larger user base and richer data collected from surveys and usage logs. Therefore, the discussion will focus on the second experiment. Due to space limitations, a brief summary of the survey results is here presented.

Among the 29 volunteers who used ActivitySpot, we received 24 survey answers. The survey was divided into four sections corresponding to the main constructs composing our evaluation system. Regarding the perception of activity and action we conclude that participants in general perceived ActivitySpot as a tool aiming at supporting their activity, where the possible actions were interdependent and part of a whole. Regarding system's performance/usefulness, results show that most participants perceived gains in using ActivitySpot, certainly because of the ability the system offered them to adopt an active role by voting on or sending comments about the show. User perception regarding ActivitySpot's personalization was high, though most of them did not regarded this feature as important. Finally, the survey allowed us to identify some conditions that hindered ActivitySpot's usage (e.g., little time available for using the system). Log analysis allowed us to observe that most interactions were made with RFID tags, SMS, and the Web interface, while 2D codes did not attract users, maybe due to its novelty and technical requirements. We also confirmed the influence of scenarios and applications in evaluating infrastructures, as some of the actions offered by ActivitySpot were not appealing enough (e.g., viewing the activity state or viewing information about next shows).

5 Related work

Project Aura [3] implements the concept of task-driven computing by capturing user intent and mapping it into a task corresponding to a set of abstract services, which are further concretized by the environment infrastructure providing continuous support to user tasks regardless of the environment in which the user is. Each user Aura represents the set of services required to accomplish a task or activity and allows the user to move from environment to environment while keeping the task in execution with the resources available in that environment.

Christensen and Bardram [2] also grounded on Activity Theory to develop a pervasive computing system (the ABC platform) supporting collaborative activities within health-care environments. Their effort is centered on environments where users are well-known (e.g., hospital staff). Like Aura, user activities are described as an abstract composition of applications which are instantiated in each environment where the user goes to (e.g., a display in a patient's room).

Our work differs from these activity-based approaches by comparing the association made between activity and the dimensions of space, person, and time. While our work is focused on activities taking place in a single physical environment, theirs is targeted to the migration of activities between different environments. While activities supported by ActivitySpot may be performed simultaneously by multiple users, Aura and ABC support mainly activities executed by a single person. Finally, while ActivitySpot is mainly targeted to activities that are unrolled within a well defined period of time, activities supported by Aura and ABC generally do not have time boundaries.

Our work also relates to a number of projects based on scenarios were ubiquitous computing supports occasional visitors to public spaces. Exploratorium

[12], Sotto Voce [13], and GUIDE [14] are examples of such projects. All these projects are based on some sort of electronic guidebook running on a PDA or a tablet PC, where users look for information related to the physical environment they are visiting. Unlike these projects, ActivitySpot does not require interaction with a specific device, but rather explores basic, heterogeneous interaction means that do not require previous training because of their generalized usage, which is definitely an advantage for visitors.

Finally, ActivitySpot shares with Gaia [15] and Interactive Workspaces [16] the objective of providing a generic computational infrastructure for ubiquitous computing. Both infrastructures also provide their own abstractions for modelling user interaction as well as easing the task of application developers.

6 Conclusion

This paper presents ActivitySpot, a ubiquitous computing framework for supporting localized activities performed by occasional visitors to public spaces. This work is based on an activity-centered approach for system design, which becomes especially important in situations in which people have little or no prior knowledge about the physical environment or about the activity they are going to perform. The main contributions of this work are the Activity Theory-inspired conceptual model and a software infrastructure, derived from this model, providing a generic tool set for ubiquitous computing environments supporting localized activities. ActivitySpot was evaluated by experiments run in different public spaces. Data collected from the experiments showed evidence of the suitability of a user interaction model based on simple, widespread, and easy-to-use devices and modalities. Although the scenarios developed on top of our infrastructure had some negative influence on part of the results, it was demonstrated that an activity-centered approach has the potential of relieving users from the cognition efforts common to application-centered approaches. Furthermore, though lacking evaluation support, our simple activity and user interaction model eases the work of public spaces administrators managing ubiquitous computing infrastructures supporting localized activities. Future work is going to be focused on developing and evaluating higher level tools for public space administrators, as well as developing the mechanisms for associating the personal domain to the local infrastructure.

7 Acknowledgements

This work has been financially supported by "Fundação para a Ciência e a Tecnologia" under grant SFRH/BD/13299/2003 and is partially supported by FEDER through POS_Conhecimento.

References

1. Banavar, G., Beck, J., Gluzberg, E., Munson, J., Sussman, J., Zukowski, D.: Challenges: An application model for pervasive computing. In: Sixth ACM/IEEE Inter-

national Conference on Mobile Networking and Computing, Boston, USA, ACM Press (2000) 266–274
2. Christensen, H., Bardram, J.: Supporting human activities – exploring activity-centered computing. In: Ubiquitous Computing 2002 (UbiComp 2002). Volume 2498., Göteborg, Sweden, Springer-Verlag (2002) 107–116
3. Sousa, J.P., Garlan, D.: Aura: an architectural framework for user mobility in ubiquitous computing environments. In: 3rd Working IEEE/IFIP Conference on Software Architecture, Montreal, Canada, Kluwer Academic Publishers (2002) 29–43
4. Norman, D.: Human-centered design considered harmful. Interactions **12** (2005) 14–19
5. Bardram, J.: Plans as situated action: An activity theory approach to workflow systems. In: Fifth European Conference on Computer Supported Cooperative Work (ECSCW '97), Lancaster, United Kingdom, Kluwer Academic Publishers (1997) 17–32
6. Wertsch, J.: The concept of activity in soviet psychology: An introduction. In Wertsch, J., ed.: The Concept of Activity in Soviet Psychology. Sharpe, Armonk, USA (1981) 3–36
7. Greenhalgh, C.: EQUIP: a software platform for distributed interactive systems. Technical Report Equator-02-002, Department of Computer Science, University of Nottingham (2002)
8. Edwards, W., Bellotti, V., Dey, A., Newman, M.: Stuck in the middle: The challenges of user-centered design and evaluation for middleware. In: 2003 Conference on Human Factors in Computing Systems (CHI 2003), Fort Lauderdale, USA (2003)
9. Compeau, D., Higgins, C.: Computer self-efficacy: Development of a measure and initial test. MIS Quarterly **19** (1995) 189–211
10. Venkatesh, V., Morris, M., Davis, G., Davis, F.: User acceptance of information technology: Toward a unified view. MIS Quarterly **27** (2003) 425–478
11. Kaptelinin, V., Nardi, B., Macaulay, C.: The activity checklist: A tool for representing the "space" of context. Interactions **6** (1999) 27–39
12. Fleck, M., Frid, M., Kindberg, T., O'Brien-Strain, E., Rajani, R., Spasojevic, M.: From informing to remembering: Ubiquitous systems in interactive museums. IEEE Pervasive Computing **1** (2002) 13–21
13. Woodruff, A., Aoki, P., Hurst, A., Szymanski, M.: Electronic guidebooks and visitor attention. In: 6th International Cultural Heritage Informatics Meeting (ICHIM 2001), Pittsburgh, USA (2001) 437–454
14. Cheverst, K., Davies, N., Mitchell, K., Friday, A.: Experiences of developing and deploying a context-aware tourist guide: The GUIDE project. In: 6th Annual International Conference on Mobile Computing and Networking (MobiCom 2000), Boston, USA, ACM Press (2000) 20–31
15. Cerqueira, R., Hess, C.K., Román, M., Campbell, R.H.: Gaia: A development infrastructure for active spaces. In: UbiTools'01 Workshop at Ubicomp 2001, Atlanta, USA (2001)
16. Fox, A., Johanson, B., Hanrahan, P., Winograd, T.: Integrating information appliances into an interactive workspace. IEEE Computer Graphics and Applications **20** (2000) 54–65