An Application Framework for a Situation-aware System Support for Smart Spaces

Arlindo Santos and Helena Rodrigues

Centro Algoritmi, Escola de Engenharia, Universidade do Minho, Campus de Azúrem, 4800-058 Guimarães, Portugal

Abstract. Despite a considerable research effort towards system support for smart spaces, in recent years we have been witnessing a growing perception about a persistent gap between the promises of the area and its real achievements. We are investigating a situation-aware system support for smart spaces that builds on a new set of assumptions and design principles, based on user-controlled associations between global services and local resources. The realisation of this concept raises important challenges across a model for global autonomous applications that enables them to integrate the functionality and resources of multiple situations. This research work aims at investigating such a model and evaluating the concept of situation-aware system support for smart spaces.

1 Introduction

Smart spaces are ordinary environments equipped with visual, audio and other sensing systems, pervasive devices, and networks that can perceive and react to people, sense ongoing human activities and respond to them [7]. Realising this vision requires infrastructures that are able to transparently manage the resources in the physical environment and provide an integrated execution environment for application development. They should provide the framework for the integration of an open, diverse and a priori unknown set of services into a functioning system, addressing key issues such as discovery, selection and spontaneous interaction between entities. Despite the wide range of existing platforms, it is not much easier today to build a smart space than it was years ago [7, 2, 14], and it is clear that the field has not matured yet to the point of enabling incremental research, a cornerstone for any research area (see section 2).

We are investigating a new approach to system support for smart spaces that builds on a new set of assumptions and design principles [11]. The new approach is based on user-controlled associations between global services and local resources and breaks the one-to-one association between a particular physical space and a particular combination of services. Instead, the concept of Situation, seen here as a socially meaningful activity that involves multiple people and can take place anywhere, is introduced as the context for the aggregation of resources and global functionality under a common purpose. A key step in enabling a situation is to associate it with the set of applications that supports the

functionality deemed appropriate for that specific situation. The local resources are also associated, subject to negotiation, to all the different situations that could occur at their physical environment. This way, multiple layers of functionality could easily be created on top of the same space to provide any type of specific support.

The realisation of this concept raises important challenges across a model for global autonomous applications that enables them to integrate the functionality and resources of multiple situations, while enabling them to be combined and appropriated in many ways. This research work aims at investigating such a model and evaluating the concept of situation-aware system support for smart spaces. The object of research is directed towards the evaluation of infrastructure as a tool for development of applications by authors without experience in programming. We hope these applications to be re-usable elements that can easily become building blocks to the creation of multiple systems.

2 Related work

There are a few ambitious middleware systems that were proposed as meta-operating systems capable of providing an integrated programming model for application developers [6, 13, 3, 4, 1]. Despite this considerable research effort, in recent years we have been witnessing a growing perception about a persistent gap between the promises of the area and its real achievements. A major conclusion at influent events in the field, such as UbiSys06 and CMPPC07, was that most systems have never had any real use outside their own development environment and incremental research has been the exception. We argue that there are four fundamental reasons for this limited success [11]. The first is a chicken and egg problem with applications. Without widely accepted reference scenarios, it is very hard to identify requirements and make informed design decisions on the type of system support that may be needed. Without a rich and operational infrastructure it is very hard to create an integrated environment where meaningful applications may emerge. The second is the implicit knowledge that prevails in those systems. Even though discovery, selection and dynamic interaction are all key goals for most platforms, there is too much hidden behaviour and too many assumptions about the environment that must be in place to bootstrap and deploy the system [2]. A third problem results from a vision of smart spaces as caring environments that sense and intelligently react to people, which raises very complex requirements associated with the need to model, detect or infer peoples feelings, intents or situations of life [8]. A final reason is the strong coupling between physical space and functionality, but we argue that Human activity is too dynamic, subtle and mobile to be captured in the infrastructure of any specific physical space.

On the other hand, the problem with the creation of ubiquitous software applications is very severe. Particularly, the application programming model [3, 5, 10] (context-aware, service-oriented, user-centric, environment-centric, task-oriented, etc) is important for software planning and development phases. Ro-

man et al. [12] have identified six patterns that were required for all applications: multi-device utilization, user-centrism, run-time adaptation, mobility, context-sensitivity, and ubiquitous computing environment independence; and have identified five design guidelines: low-level system support, boot-strapping, scripting, application development and management support, and end-user support, which were considered essential to support ubiquitous computing environments and to increase number of application developers.

The Ubiquitous Web Applications Working Group focuses on extending the Web for enabling distributed applications of many kinds of devices including sensors and effectors. Application areas include home monitoring and control, home entertainment, office equipment, mobile and automotive [15]. Also following the Web 2.0 path [9], the Facebook social network offers an application framework oriented to application rapid development cycle and sharing with objective of increasing the number of application and developers.

Current state-of-the-art on ubiquitous application frameworks have not been designed for supporting situation-aware application characteristics and requirements, situation we have identified as a new research path for ubiquitous applications development frameworks.

3 Research hypotheses

This work builds on the assumption that a situation-based infrastructure represents an important paradigm shift in System Support for Smart Spaces and opens new paths towards the long-time goals of this area. The overall system model addresses the limitations identified in section 2 by approaching system support for smart spaces from a new perspective that is characterised by associating functionality with the concept of a Situation. Applications for a situation are viewed as self-contained blocks of functionality that are globally available and may serve multiple situations. A person creating system support for a particular situation will attach to that situation whatever applications may be useful. Applications will take advantage of the resources and content that have previously been associated with the situation. Functionality emerges from the combination of the application logic with the interaction capabilities of local resources within the framework of a particular situation. This particular approach reduces the complexity of basic applications and also blurs the distinction between system support and applications. The goal of this work is to develop an application framework for a situation-aware system support for smart spaces, addressing the necessary meta-level concepts, protocols, application development interfaces and tools to automate and control the life cycle, sharing, and execution of applications. This dissertation wants to show that:

An application framework for a situation-aware system support for smart spaces will provide the basis for third-party developers to experiment creatively and open the way for more creative and open-ended appropriations of the functionality supported by the system, while reducing the required programming skill level.

4 Work plan

Situation-aware applications correspond to a logical block that is able to provide functionality directly to users within the context of a situation, by leveraging on the resources and content associated with that situation. This view is inspired by the Facebook, OpenSocial and other Web 2.0 application models, in which application development is open and existing applications can be shared between multiple pages, have their own functionality, but their data and services be easily combined and appropriated in many ways.

Currently, we are building an Application Programming Interface to enable applications to access the main situation-aware infrastructure services. Developers will be able to add the situation context to their application by utilizing situation accounting, presence, document, location, and messages data. This task aims at supporting case studies prototyping, which evaluation will provide early feedback on state-of-the-art and new application patterns.

Next, the definition of the application model will consider the application's life cycle, the hosting and the integration of application output with situation resources both for desktop and web applications. At this stage we will consider mainly the reutilization and sharing application patterns. From the perspective of making applications developments more expedite and accessible to other domains specialists, including developing, deployment and evaluation, this work will investigate the development of a tool for rapid authoring of situation-aware applications.

Finally, the evaluation task aims at evaluate in what extent the Application Programming Framework successfully contributes for the realisation of classical and new smart space scenarios. More precisely, we want to measure in what extent we reduce the overwhelming effort it was to take someone elses middleware and build smart spaces computing applications on top of it, and because of that how the goal of having a widely accepted middleware that is effectively used by multiple developers of smart environments is yet to happen.

5 Conclusions

The goal of having a widely accepted middleware is yet to happen, and there is lack of common ground and metrics for evaluation. There is absence of established reference scenarios or clear demands for smart spaces and ubiquitous computing: all domains have different constraints and so hamper common approaches to system support for applications.

We propose an application framework for a situation-aware system support for smart spaces which aims at democratizing application development following the Web 2.0 path. Our objective is to promote incremental research, given that infrastructure will be open and designed for re-use and sharing of local resources and applications in different situations. Multiple entities will have the possibility to use this infrastructure to create a diverse set of smart spaces, leading to a rich test bed for the identification of new requirements and for improving our understanding of this problem domain.

References

1. F. J. Ballesteros, E. Soriano, G. Guardiola, and K. Leal. The plan b os for ubiquitous computing. voice control, security, and terminals as case studies. *Pervasive and Mobile Computing*, 2(4):472 – 488, 2006. Special Issue on PerCom 2006.
2. A. Friday, M. Roman, C. Becker, and J. Al-Muhtadi. Guidelines and open issues in systems support for ubicomp: reflections on ubisys 2003 and 2004. *Personal Ubiquitous Comput.*, 10(1):1–3, 2005.
3. D. Garlan, D. Siewiorek, A. Smailagic, and P. Steenkiste. Project aura: Toward distraction-free pervasive computing. *IEEE Pervasive Computing*, 1(2):22–31, 2002.
4. R. Grimm, J. Davis, E. Lemar, A. Macbeth, S. Swanson, T. Anderson, B. Bershad, G. Borriello, S. Gribble, and D. Wetherall. System support for pervasive applications. *ACM Trans. Comput. Syst.*, 22(4):421–486, 2004.
5. I. Y. Hen, E. Jansen, and S. Helal. A comparison of two programming models for pervasive computing. In *Applications and the Internet Workshops, 2006. SAINT Workshops 2006. International Symposium on*, page 4 pp., 2006.
6. B. Johanson, A. Fox, and T. Winograd. The interactive workspaces project: Experiences with ubiquitous computing rooms. *IEEE Pervasive Computing*, 1(2):67–74, 2002.
7. T. Kindberg and A. Fox. System software for ubiquitous computing. *IEEE Pervasive Computing*, 1(1):70–81, 2002.
8. L. Leahu, P. Sengers, and M. Mateas. Interactionist ai and the promise of ubicomp, or, how to put your box in the world without putting the world in your box. In *UbiComp '08: Proceedings of the 10th international conference on Ubiquitous computing*, pages 134–143, New York, NY, USA, 2008. ACM.
9. T. Oreilly. What is web 2.0: Design patterns and business models for the next generation of software. *Communications & Strategies*, First Quarter 2007:13, 2007.
10. H. Pinto, R. Jose, and J. C. Campos. An interaction model and infrastructure for localized activities in pervasive computing environments. In *2007 IEEE INTERNATIONAL CONFERENCE ON PERVASIVE SERVICES*, pages 232–241, 345 E 47TH ST, NEW YORK, NY 10017 USA, 2007. IEEE, IEEE. IEEE International Conference on Pervasive Services, Istanbul, TURKEY, JUL 15-20, 2007.
11. H. Rodrigues and R. José. Towards a new research agenda in system support for smart spaces. Draft report, Centro Algoritmi, Escola de Engenharia, Universidade do Minho, July 2009.
12. M. Roman, J. Al-Muhtadi, B. Ziebart, R. Campbell, and D. Mickunas. System support for rapid ubiquitous computing application development and evaluation. In *System Support for Ubiquitous Computing Workshop (UbiSys '03) in conjunction with UbiComp '03*, Seattle, WA, 2003.
13. M. Roman, C. Hess, R. Cerqueira, R. H. Campbell, and K. Nahrstedt. Gaia: A middleware infrastructure to enable active spaces. *IEEE Pervasive Computing*, 1:74–83, 2002.
14. O. Storz, A. Friday, N. Davies, J. Finney, C. Sas, and J. Sheridan. Public ubiquitous computing systems: Lessons from the e-campus display deployments. *Pervasive Computing, IEEE*, 5(3):40–47, July-Sept. 2006.
15. W3C. Ubiquitous web applications working group, 2009. http://www.w3.org/2007/uwa/Activity.html.