Open Issues in Nomadic Pervasive Computing

Hen-I Yang, Shinyoung Lim, Jeffrey King and Sumi Helal

Mobile and Pervasive Computing Laboratory
Computer & Information Science and Engineering Department
College of Engineering, University of Florida,
PO Box 116120, Gainesville FL 32611, USA
{hyang, lim, jck, helal} @cise.ufl.edu
www.icta.ufl.edu

Abstract. As a user roams among islands of smart spaces, additional support within the systems are needed to provide the user with seamless experience and services wherever he or she goes. We refer to such environments as *nomadic pervasive computing*. In this paper, we identify the open issues and suggest necessary system supports to enable such environments. Several events must occur when users move from one location to another. The space must take the users' preferences and roaming services, and bind them to locally available physical resources. Measures should be taken to guarantee the delivery of emergency services, while ensuring some quality of service for others. The systems must be scalable and secure, and protect the users' privacy where possible. We intend for the issues raised here to foster further thoughts about requirements and system to enable nomadicity among smart spaces.

1. Introduction

We believe that in the near future, smart spaces, where ubiquitous computing services are available [1, 2], will become much more widespread. However, it is unlikely that these islands of smart spaces will be joined seamlessly or continuously with each other. Therefore, as users move about, they will have to roam from one autonomous smart space to another. Ideally, users should not have to loose or be denied all the features they have come to expect simply because they move to a new space. We coined the term *Nomadic Pervasive Computing* to refer to systems that encompass a federation of isolated smart spaces and the mechanisms that allow users to be continuously served as they move around.

As people get used to new technology, they come to expect its availability, and dependency on the technology grows. This is especially true of smart spaces because their goal is to embed technology into living environments, which can greatly impact many daily activity routines. Consider an elder person with early-stage Alzheimer, whose independent living depends on a cognitive assistance service that cues her to finish tasks she has started. When she leaves her apartment to go to the grocery store, she still counts on receiving the same kind of help. This is why we must have nomadic pervasive computing, so that critical services are not lost just because the user leaves her home space.

This paper discusses the challenges posed by users' nomadic behavior in pervasive computing systems. These challenges are divided into four categories: binding of services and user preferences, quality of service, scalability, and security and privacy. After we raise these challenges, a brief survey of the research related to these issues is also provided.

2. Challenges of Nomadic Pervasive Space

The realization of the continuous service provisioning from smart spaces demands the following challenges to be addressed.

2.1 Binding

Binding refers to the attempt of migrating a service running in one pervasive space (usually owned space) to another pervasive space (usually temporary, visited space), as well as reconnecting user's preferences to this visited space. We identify four issues with binding.

1. Separating goals and means.

It is crucial to separate goals and means in nomadic pervasive spaces. Consider a medicine reminder service. Broadcasting messages on the stereo system at home is an excellent means to prompt the resident to take medicine. But it may not be the most feasible modality when the person is shopping at the supermarket. Here, an alternative such as text messaging the user would be much more appropriate. Some aspects of this issue have been widely researched in regards to context awareness. In nomadic pervasive computing the problem has been intensified because different spaces may have drastically different resources and capabilities for context acquisition and behavior delivery.

2. Deciding where binding occurs

When a user arrives in a new smart space, at which level are the user's preferences bound to the space's configuration?

One idea is to bind at the *device level*. With this approach, each device is associated with particular domains, and services bind to the devices that support their needed domains. For instance, a climate control service, which affects the "temperature" domain, can be bound to devices such as an air conditioner, electric heater, or ceiling fan.

Another alternative is to bind at the *user profile level*. Here, the user's preferences are adapted and converted to conform to the protocols of the new space.

There are of course other levels at which binding could occur [3]. Research in nomadic pervasive computing should involve identifying the factors that affect where binding should occur for particular architectures or applications.

3. Interoperability and compatibility

Device descriptions, service/application descriptions and user preferences are some of the crucial information in specifying and coordinating how a smart space works. The *syntactic and semantic grammars* have to be well defined for the entire smart space to work properly. On top of this, the difference in programming models may also cause incompatibility. Many researchers have looked into extensive use of *ontology and standardization* of service description, but no definite conclusion has been reached yet. While this issue exists in single-location smart spaces, it is relatively easier to ensure compatibility of entities under the same roof than roaming around to different locations between systems, possibly internationally.

4. *Ownership of space and authentication of entities*

When users and devices move in and out of smart spaces, issues related to authentication and ownership take the central stage. In the *private* setting of a smart home, the owner of the house should have the full control of the dwelling. Any device and service brought in or approved by the owner can be trusted and integrated into the environment. In the *semi-private* setting, such as paying visits to a friend's smart home, some sort of rules have to be established to accommodate the visitors' needs while maintaining the control and preference of the space owner. For instance, it would be more hospitable to allow the visitor to setup the room temperature of the guest room, but not the master suite. In the *public* setting, such as a trip to the supermarket, a more restricted model may be adapted to better serve the large number of occupants to the public place, while still offer important services, such as patching through shopping list or medicine reminder to users' handheld device or smart phone. The system has to be able to recognize the boundary as to which services are legitimate and allowed, while others may cause adverse effects to the occupants or the interest of general public. How to characterize each person, device and service, and authenticate them and identify their ownership is one of the key concerns in nomadic pervasive space.

2.2 Quality of Service (QoS)

QoS of nomadic pervasive computing systems is influenced by many factors, for instance, the number of migrating services, the priority of services, and pattern of user's mobility. The characterizing criteria and quantification of QoS in nomadic pervasive computing remain to be investigated and define.

1. *Conflict resolution: priority and scheduling*

Conflict resolution can be regarded as another facet of the authentication and ownership, but it is important enough to be considered separately. How do we decide which of the conflicting commands issued by different applications should overwrite all the others? How do we decide the priority of conflicting preferences of users? When including time dimension in the consideration, how do we schedule the execution of the commands so as to handle emergency situations promptly and preserve the quality of service in general.

2. *Criteria of QoS*

The diversity and impact of smart space on its occupants and the spaces are much larger than networking services or traditional systems. What are the *quantifications* needed to be considered in addition to the traditional QoS measures such as response time, throughput, service level or degree of jitter? How do we *maintain the acceptable QoS* in face of missing components, such as delayed/lost user profiles, incompatible service descriptions, missing sensors or actuators? A flexible and adaptive mechanism need to be in place to guarantee certain level of QoS.

3. *Service level agreement*

   Not all services are created equal. Some are inherently more important than others. When moving from one smart space to another, how do we classify services into appropriate levels, and how do host smart house negotiate with visiting users and entities in terms of the service level, pricing and resource contention. For instance, the emergency button service should be mandated to be supported universally, regardless of the resource availability, while medicine reminder should be activated in most spaces and personal music preference profile may not deem necessary in many scenarios.

4. *Mobility*

   Wireless network coverage has gradually grown from points (such as hot spots) into planes, and networked applications can now make use of vertical and horizontal handoffs [4] rather than falling back on weakly connected and disconnected modes [5]. However, until smart spaces reach the same level of ubiquitous availability, nomadic pervasive computing will rely on the use of *weakly connected* and *disconnected mode*.

   When considering how to migrate from one smart space to another, we should determine what elements can be taken from projects such as Internet Suspend/Resume and what criteria are specific to smart spaces. Which *states and contexts* can be used as digital representations of users – would user profiles alone be sufficient? What is equivalent between preserving the state of a workstation as compared to a smart space? Can we forward the states to the target destination ahead of time to allow seamless service provision? How do we decide whether a running service should migrate, stay, or terminate?

2.3 Scalability

1. *Service delivery*

   Considering the number of potential occupants in public spaces, and that each person carries a different set of critical services and preference profiles, it is obvious that distributing and delivering services represents a major scalability issue. What is a suitable architecture for supporting a large number of users and devices in the smart environment? How can we congregate similar services into a single instance that serves many users? Should *layered infrastructure* be enforced to separate service logic from its user-specific customization? These are all crucial issues that need to be considered when developing scalable service delivery.

2. *Scalability of data and processes*

   Just as these systems must support a larger number of services, the complexity of context-aware applications and the number of diverse entities within smart space results in considerably larger number of processes and amount of data than compared to traditional applications in non-pervasive computing environments. Whether the system takes a data-centric, process-centric, or context-centric approach, the complexity remains an issue that cannot be easily addressed. Through the efforts of data assimilation and congregation, in-node processing or service choreography, the scalability issue might be mitigated, but a generic solution to scalability issue remains to be an interesting research topic.

2.4 Security and Privacy

Once leaving the relatively safe settings of ones' own house, additional challenges in security and privacy becomes evident. On one hand, the system needs to protect the hosting smart environment from malicious visitors and devices. On the other hand, the privacy of the users need to be protected so their private information will not be exploited or illegally dispersed once they visit other smart spaces. The challenge here is larger than the sum of parts. All the usual mechanisms for providing security and privacy in the protocol stack applies, but in addition, the frequent joining and disengaging by foreign users and devices implies that the proper and efficient authentication protocol and mutual agreement on privacy policies must be well defined.

Some technical issues include, but not limited to, the ownership and usage of users' records and preferences during the stay in a host smart space, the extent of trust regarding the chain of authentication on entities and services associated with a smart space or a user, and the efficient and effective measures of access control in such a dynamic environment.

For instance, when a user goes to grocery shopping, the medicine reminder service must follow so she wouldn't forget to take the pills. An efficient mutual authentication between the supermarket and the service must be in place, along with an agreement on the usage and recording of exchanged information, such as the medical condition and medicine to be taken, so both the user and the supermarket are protected, and the privacy of the user is preserved.

3. Related Work

Internet Suspend/Resume (ISR) is a mechanism that applies virtual machine technology on top of a location-transparent distributed file system. It relates to the binding issue, that is, the rapid personalization and depersonalization of anonymous hardware during transient use [6, 7]. ISR assumes transitions occur between workstations. It does not require users to carry portable storage or mobile devices, but the distributed file system requires more functional components for identifying the user, device and service environment [7]. Other systems, such as Aura [8, 9], make use of distributed file systems to expand the capabilities of mobile devices.

A QoS-aware middleware, such as UbiData [10], manages the quality of service of a distributed system, and provides application-level support for user mobility. It assumes that user locations are known and that the network interface is predefined. This architecture concentrates on adapting applications to heterogeneous system environments, but it is relatively unconcerned with using the user contexts and profiles to support mobility.

A different middleware, described by [11], address QoS and scalability issue by providing the tools to manage dynamic resources, user mobility, dynamic user needs, and system faults. It provides an adaptation framework with three layers: the task layer, model layer, and runtime layer. While this model is sound from the architectural point of view, it still requires additional features for assurance of quality of service in each layer for user mobility.

The Wireless World Research Forum (WWRF) provides a Service Reference Model describing service semantics, generic service elements, and a service platform for solving the binding, security and privacy issues. The early specification is promising, but still lacks functional details between each component [12].

Most of the related work focuses on one individual problem as relevant to one particular domain, but many of these problems cross several domains, and new problems arise when attempting to address the collective issues. Nomadic pervasive computing is one of these cross-domain problems. We have identified the domains and analyze specific issues that must be addressed to provide continuous service across different smart spaces.

4. Conclusion

It has been almost two decades since Mark Weiser wrote about what became ubiquitous computing. The research and prototype projects developed since have grown dramatically in scale and complexity, signaling that mainstream ubiquitous computing environments are coming. As more and more smart spaces come online, users will be interacting with different spaces at different times. As users begin to rely on provided services, it seems natural that they would want useful services and customized settings from one space to "follow" them into another.

The support for mobility between ubiquitous computing spaces, the defining characteristic of nomadic pervasive computing, poses intriguing challenges beyond the scope of individual smart spaces. We hope that the issues we have raised in this paper will foster further investigations into how nomadic users can enjoy a continuum of seamless experiences and services wherever they go.

References

[1] A. Helal, W. Mann, H. Elzabadani, J. King, Y. Kaddourah and E. Jansen, "Gator Tech Smart House: A Programmable Pervasive Space", IEEE Computer magazine, March 2005, pp 64-74.

[2] A. Helal, B. Winkler, C. Lee, Y. Kaddourah, L. Ran, C. Giraldo and W. Mann, "Enabling Location-Aware Pervasive Computing Applications for the Elderly" Proc. 1st PerCom Conf. , 2003, Texas.

[3] E. Jansen, B. Abdulrazak, H. Yang, J. King and A. Helal, "A Programming Model for Pervasive Spaces", Technical Report: http://www.harris.cise.ufl.edu/projects/publications/progmodel-report.pdf

[4] Mark Stemm and Randy H. Katz, "Vertical Handoffs in Wireless Overlay Networks" ACM Mobile Networking (MONET), Special Issue on Mobile Networking in the Internet, Summer 1998.

[5] Satyanarayanan, M., Kistler, J.J., et al., “Experience with Disconnected Operation in a Mobile Computing Environment ”,Proceedings of the USENIX Symposium on Mobile and Location-Independent Computing, Jun. 1993, Boston, MA

[6] Michael Kozuch and M. Satyanarayanan, “Internet Suspend/Resume,” the Fourth IEEE Workshop on Mobile Computing Systems and Applications, June 2002.

[7] Michael Kozuch, at. al., “Seamless Mobile Computing on Fixed Infrastructure,” Research at Intel, IPR-TR-04-28, 2004.

[8] Joao Pedro Sousa, et. al., “Aura: An Architecture Framework for User Mobility in Ubiquitous Computing Environments,” Proceedings of the 3rd Working IEEE/IFIP Conference on Software Architecture, pp. 29-43, August 2002.

[9] Joao Pedro Sousa, et. al., “Capitalizing on Awareness of User Tasks for Guiding Self-Adaptation,” 17th Conference on Advanced Information Systems Engineering, Porto, Portugal, 2005.

[10] Klara Nahrstedt, et. al., “QoS-Aware Middleware for Ubiquitous and Heterogeneous Environment,” IEEE Communications Magazine, November, pp.140-148, 2001.

[11] Shang-Wen Cheng, et. al., “Software Architecture-based Adaptation for Pervasive Systems,” International Conference on Architecture of Computing Systems (ARCS'02): Trends in Network and Pervasive Computing, April 8-11, 2002.

[12] WWRF-Wireless World Research Forum: http://www.wireless-world-initiative.org/