

Deakin Research Online

This is the authors' final peer reviewed (post print) version of the item published as:

Abawajy, J. H. 2009, Advances in pervasive computing, *International journal of pervasive computing and communications*, vol. 5, no. 1, Special issue: Advances in pervasive computing, pp. 4-8.

Available from Deakin Research Online:

<http://hdl.handle.net/10536/DRO/DU:30025932>

Reproduced with the kind permission of the copyright owner.

Copyright : 2009, Emerald Publishing Group

Advances in pervasive computing

GUEST EDITORIAL

J.H. Abawajy, *School of Engineering and Information Technology, Deakin University, Geelong, Australia*

Acknowledgements

There are many people who were involved in the effort to put this issue together. The Guest Editor would like to thank all the authors of the papers for their hard work and their contributions to the story of pervasive computing. He would also like to thank the many reviewers for their thoughtful and insightful comments and the time they spent reading papers. When the issue was first put together, an advisory committee offered advice and direction. The Guest Editor thanks Shumin Zhai, Chris Dryer, Paul Maglio and Chris Campbell for serving on this committee.

Abstract

Purpose – The purpose of this paper is to provide an overview of advances in pervasive computing.

Design/methodology/approach – The paper provides a critical analysis of the literature.

Findings – Tools expected to support these advances are: resource location framework, data management (e.g. replica control) framework, communication paradigms, and smart interaction mechanisms. Also, infrastructures needed to support pervasive computing applications and an information appliance should be easy for anyone to use and the interaction with the device should be intuitive.

Originality/value – The paper shows how everyday devices with embedded processing and connectivity could interconnect as a pervasive network of intelligent devices that cooperatively and autonomously collect, process and transport information, in order to adapt to the associated context and activity.

1. Introduction

Advances in sensors, embedded systems, wireless communications and networking, radiofrequency identification (RFID) tags, middleware systems, and the like have led to the evolution of the next generation distributed computing platform referred to as pervasive computing also known as ubiquitous computing. Pervasive computing moves processing and communication technology beyond the personal computer to everyday devices such as key chains, clothing, tools, cars, homes, coffee mug, toys, milk cartons, appliances (e.g. televisions, wristwatches, and toasters) and human body. These everyday devices with embedded processing and connectivity could interconnect as pervasive network of intelligent devices that cooperatively and autonomously collect, process and transport information, in order to adapt to the associated context and activity.

Pervasive computing seeks to provide proactive and self-tuning environments and devices to seamlessly augment a person's knowledge and decision making ability, while requiring as little direct user interaction as possible. For example, you can access and interact with information and services instantly from anywhere in the world. Therefore, there is no need to carry around devices containing our information as environments will automatically recognize who we are and obtain information about us through various means such as agents, adaptive user models, and information storage. Moreover, pervasive computing enabled devices will be more aware of their users and more aware of their own environment (e.g. their position and their surroundings). In addition to sensing the presence of a user, these devices are also able to sense the user's needs and goals to enhance person-to-person communication. Thus, pervasive computing can provide wonderful functionality without requiring that the user understand its inner workings.

Although pervasive computing revolution has already begun to affect our lives in ways we do not even notice, there are many challenges ahead to fully realize the objectives of pervasive computing. Among myriad tools expected to support these are: resource location framework, data management (e.g. replica control) framework, communication paradigms, and smart interaction mechanisms. Also, infrastructures needed to support pervasive computing applications and an information appliance should be easy for anyone to use and the interaction with the device should be intuitive. This issue contains a collection of papers that explore these topics. These papers will help technologists to share in the successes of others in this field and also to understand problems researchers are having in creating ubiquitous computing environments. Our experiences are conveyed to motivate future work in the area and to help all of us envision and create the future.

2. Resource location framework for hybrid ubiquitous networks

The resources of a network are made up of many kinds of software and hardware components, including service objects, information, computational units, storage devices, and peripherals. In the ubiquitous computing paradigm, these resources must be accessible virtually anywhere and at any time via any device. This requires an infrastructure to support resource location by a mobile user, who may be connected by means of a diverse range of networks and client devices.

Resource location frameworks make networked resources easier to deploy and configure, which contributes to a rich mobile computing environment, in which a new resource (e.g. a map navigation service) becomes usable as soon as it is plugged in. This reduces the effort required for configuration and saves valuable system administration time, because a new resource adjusts to its surroundings with little additional help. Users also benefit from a resource location framework because a client (e.g. a global positioning system (GPS)-enabled PDA in the case of a navigation service) can find and use the new resource immediately without forcing the user to search for the resource manually, identify its type, and prepare to use it. If the resource is removed, or a more appropriate resource joins the current network, or the client roams away from the network, the client automatically disassociates itself from the resource it is using and tries to find another.

Pervasive computing systems operate in large-scale, highly heterogeneous domains with very unpredictable dynamics. Some interesting design issues in developing a resource location framework for pervasive computing includes: Scalability, interoperability, Context

awareness, Adaptability and Efficiency. So far, relatively little attention has been paid to overcoming these challenges. We have therefore developed the Eureka which is an integrated resource location framework for ubiquitous computing. It can adapt locally to changing conditions, such as network type, remaining power, network stability and resource popularity, while providing gateway bridging to support network-level interoperability. It also incorporates context awareness to improve the quality of its resource location service. Eureka also offers support for interoperability between different protocols running on various types of network. By providing a bridge service through gateway nodes, Eureka makes it possible to locate resources across an entire global network. Eureka requires a lower control overhead for resource location than comparable protocols, which saves energy, reduces latency and allows scaling to large ubiquitous environments.

3. Data object replica control scheme

Peer-to-peer (P2P) paradigm is a distributed computing paradigm where entities can join and leave the network at anytime and can directly interact as equal peers and share information, services and resources in a distributed manner. P2P systems are usually designed to accommodate a large number of nodes and to adapt to dynamic node joining and leaving. Following the great success of P2P file sharing, a natural next step is to explore the possibility of aggregating huge idle P2P storage spaces across the Internet and make it available for applications that need it.

P2P storage system has been an active research topic and are generally classified as structured P2P storage systems and unstructured P2P storage system. Although unstructured P2P storage system is easy to build and maintain, it is difficult to guarantee the quality in accessing the stored data. Thus, many of P2P storage systems are built on top of structured P2P networks. While P2P storage systems are gaining popularity, a central challenge is how to deal with the problem of node failure or departure of the network. Although the underlying P2P routing can adapt to dynamic node joining and leaving, the data object stored in nodes will be lost when nodes fail or leave the network.

A common solution to this problem is for peers to increase the availability of their data by replicating it on other peers in the network. If the data objects are read-only (or non-mutable), then the P2P storage system will only need to consider where to replicate the data objects. The system becomes much complicated if the data objects are mutable. For synchronous replication in DHT-based mutable P2P storage systems, replication system must address the state synchronization problem, the replica acquisition problem and the replica migration problem. To address these problems, Jiang *et al.* (2009) discuss a mutable replica control scheme (MUREX) to keep one-copy equivalence for synchronous replication in structured P2P storage systems. MUREX is based on the concepts of multi-column read/write quorums, replica pointers, on-demand replica regeneration and leased locks. Jiang *et al.* have proved that MUREX guarantees one-copy equivalence and causes no deadlock. Furthermore, they have analyzed and simulated MUREX showing that MUREX has constant communication cost in the best case and has good operation success rate.

4. Publish/subscribe communication scheme

Distributed systems consist of a set of processes that can communicate with each other by exchanging messages. There exists quite some variety amongst message communication

paradigms for distributed systems. One paradigm that supports asynchronous, anonymous and many-to-many communication is the publish/subscribe paradigm. In publish/subscribe communication; subscribers register their interests in certain event conditions with the publisher. The publisher will notify the subscribers when the event that matches their registered interests occurs. Publish/subscribe paradigm systems can be generally characterized as topic-based, subject-based and Content-based. Events are classified and labeled by publisher as belonging to one of a predefined set of subjects in both topic-based and subject-based schemes. In contrast, rather than being restricted to (or even requiring) pre-defined subject fields, subscribers in Content-based scheme have the added flexibility of selecting filtering criteria along multiple dimensions, using thresholds and conditions on the contents of the message.

However, content-based publish/subscribe present a unique challenge not only for efficient matching of events to subscriptions but also for efficient event delivery. In fact, efficient event delivery in a content-based publish/subscribe system has been a challenging problem. To address this problem, Yuan *et al.*, discuss event routing strategies for content-based publish/subscribe systems. They propose two routing algorithms referred to as strict subscription-cover based routing (SSCBR) and relaxed subscription-cover based routing (RSCBR) (Yuan *et al.* 2009). SSCBR maintains the least covered subscriptions while RSCBR maintains more covered subscription to balance the overhead in memory, time and network traffic.

5. Instant Messaging and presence awareness

Instant Messaging (IM) provides a simplest and convenient form of interaction, i.e., near-synchronous message exchange. IM has recently gained enormous popularity and is now penetrating almost every aspect of daily lives, at home, at work, at school, and on the road. Currently, the usage of IM is only to support human-to-human communication. A distinctive feature that differentiates IM from other communication tools such as email or telephone is presence awareness that in the basic level provides the availability status or responsiveness of a user to engage in a conversation. This work envisions the potential of extending the Instant Messaging (IM) paradigm into pervasive computing environments. In this attempt, all smart entities, human or resource, can interact using IM as the unified interface. To realize this vision, a Smart Instant Messaging (SIM) system is proposed, featuring context-aware presence management, user-centric resource configuration and adaptive grouping mechanisms. This system extends the jabber-based IM framework and relies on an ontology-based supporting middleware to handle the chore of retrieving and interpreting context information. Three versions of clients are prototyped and performance is evaluated concerning memory usage and response time.

6. Human-computer interfaces

Information appliances have human-computer interfaces. An information appliance should be easy for anyone to use and the interaction with the device should be intuitive. Careful design is critical for an intuitive interaction with the device. Although the desktop computer can do many things, this functionality can be separated into more appropriate devices. Some examples of successful popular devices are cellular phones, pagers, televisions, wristwatches, and toasters. Of course, there can be times when these devices become difficult to use, but in their basic form, they meet the criteria for information appliances.

Devices will become more "aware". A device will be more aware of its user and more aware of its own environment. Devices will not only be able to sense the presence of a user but also be able to sense the user's needs and goals. Devices will be aware of their position and their surroundings. Biosensing will become prevalent throughout the environment, not only for entertainment and medical reasons, but also to enhance person-to-person communication. When devices become more aware, they can be responsive and seem "smarter".

Computers will have the sensory devices analogous to human senses: sight, sound, speech, touch, and smell. Perhaps the best way for computers to really help humans is for computers to become more a part of the physical, human world. Maybe it is the nature of humans to create things with an image of themselves in mind.

References

Jiang, J.-R., King, C.-T., Liao, C.-S., Liu, C.-H. (2009), "MUREX: a mutable replica control scheme for structured peer-to-peer storage systems", *International Journal of Pervasive Computing and Communications*, Vol. 5 No.1, pp.9-24.

Yuan, H.L., Guo, C.G., Peng, Z. (2009), "The subscription-cover-based routing algorithm in content-based publish/subscribe", *International Journal of Pervasive Computing and Communications*, Vol. 5 No.1, pp.25-41.

Further reading

Zhang, X., Law, C.-F., Wang, C.-L., Lau, F.C.M. (2009), "Towards pervasive instant messaging and presence awareness", *International Journal of Pervasive Computing and Communications*, Vol. 5 No.1, pp.42-60.

Corresponding author

J.H. Abawajy can be contacted at: jemal@deakin.edu.au