Building Context Awareness into Physical Service Environments

G. Cortese, F. Davide

Giovanni Cortese, g.cortese@computer.org
Fabrizio Davide, Telecom Italia Learning Services fabrizio.davide@telecomitalia.it

I. Introduction

With the deployment of WLAN connectivity in both the private and public space the opportunities for businesses to offer a variety of services to users (including visitors and internal users) moving within their physical premises are increasing. At the same time, emerging sensing and semantic interpretation technologies enable the implementation of presence, identity and localization services that are key enablers for a range of intelligent services, that exploit context information to adapt to user and environment state. Examples of such "physical service environments" include airports, shopping malls, conference centers and hospitals. In this paper we discuss a set of challenges that we must confront with before we can expect an ubiquitous deployment of context aware services.

II. Scenario

Physical Service Environments (Figure 1) provide services that are enhanced by knowledge of the physical environment such as proximity, environmental characteristics, and specific knowledge on the local and user. A physical environment seamlessly integrates the services provided by all the computing devices it contains – including embedded systems, network connected appliances and portable devices of mobile users – as well as those supplied by more traditional application servers running in a separate computer room or remotely in a computing grid.

Users moving in such pervasive computing environment access all service available with ease; in turn, users' devices provide services to and share data with other components in the environment.

Context awareness add a level of intelligence to physical service environments. Environmental Context is information describing the current state of the environment. This includes sensed, dynamic context data, such as location information, presence and identity of people, specific situations occurring within the space, as well as static data such as spatial model of environment, inventories of objects, and so on. Service environments should gather, organize and 'publish' context information so that applications can use them.

Similarly, users are expected to share with the environment a range of information about themselves, including their position, preferences, profile information, tracks of past interactions with the services in the environment, skills, plans and schedules, which the local service environment can exploit to deliver services adaptive to users.

III. Case Study

We have implemented at Telecom Italia premises a testbed where to experiment with context aware computing. This testbed encompasses a 400 mq area of the company office space, in which we have deployed WLAN connectivity, WLAN-based location technology, and a few WebCams. We developed XDM, a context data management middleware, and a few services on top of it. Example of services in the testbed include indoor guide, find-a-friend, device leasing, monitoring of queues at facilities such as copiers, ATM etc.

Service provisioning to mobile users is based on Web protocols; a Tomcat application server is the container where a number of context aware services are deployed. Context aware provisioning of services allows to dynamically adapt services and content available to the user (either in the form of context-sensitive menus or information push to the user) based on criteria such as position, time, and environmental conditions.

A description of an early version of this prototype can be found in [1].

IV. Generalizing the Case Study: Architectural Challenges for Physical Service Environments

Drawing on the experience described, we have been working in the last two years having as a driving goal the realization of widely accepted, robust and manageable infrastructures for deployment of context aware services.

By generalizing our initial prototype, we have identified a number of challenges to be tackled, and possible approaches, which we summarize in this section.

A manageable, service-grade pervasive computing environment

Context Aware services go hand in hand with pervasive computing. Middleware architectures for pervasive computing which can get a wide acceptability by the industry must establish in order for context aware services to take off.

The first challenge is then to design a middleware architecture that provides communication, coordination (including transactions), discovery, and composition of services across the whole range of computing devices operating in a physical service environment, including J2EE application servers, embedded systems, and mobile devices with limited capabilities. Contract negotiations, possibly on the fly, and integration with payment/ billing infrastructures are also key element of such middleware [2].

Existing service oriented technologies including Web Services, OSGi, GRID should be enhanced and integrated into a single scalable framework which supports development of industry-grade pervasive applications. Sensor networks in the long term should also become interoperable components in this framework.

Context Information Management architectures

Standardized context information management architectures are needed to effectively decouple context acquisition from its use. Though a few research projects have shown directions [3], we are still far away from a standard architecture which a) allows vendors of sensors and semantic interpretation software to generate information according to well-known schemas b) let applications easily discover and use available context information, to provide adaptive behavior. The processes of sensor data generation, fusion, their transformation into context information with a clear semantics reusable from different applications, and sharing between producers and consumers should be organized according to standard interaction patterns and ontologies.

Technologies for context data sharing and distribution.

Context data collection and distribution is inherently a peer-to-peer application. Publish-subscribe and data sharing mechanisms are often used in implementations of context management infrastructures. For example, in the VICom project [6] we adopt a context data management architecture based on a combination of Lime [4], which provides mechanisms for data sharing between peer components, and DHT-based middleware [7], which have very promising scalability properties, to match the demand of large scale context data management infrastructures.

Privacy is a major challenge to context information management. Access control on sensed data should be in the hand of service users. Methods for controlled disclosure and permission of usage of such information, to selected services and under controlled condition should be devised.

User context

Mobile user context seamlessly extends from the user mobile device into her personal computing environment.

Data (such as documents, folders) belonging to the user workspace (often hosted in her home or enterprise), and which are relevant to 'current' activities should be made by the middleware always accessible to the mobile user. Within the Simplicity project, we are exploring how to provide transparent, network-based storage of profile data and documents to a mobile user, by leveraging emerging technologies for peer-to-peer storage management [8].

Semantic interoperability

In dynamic environments (which include both the Internet and pervasive computing spaces, where components should work together without being designed for interoperability), semantic interoperability is a key issue. Semantic interoperability is one the grand challenges of pervasive computing. It is however also a specific challenge for adding context awareness to pervasive computing environments. Standardization of context ontologies can help (and for 'low level context' such as user position and movement, environmental conditions e.g. sound, light the standardization approach should work); however for higher level context information achieving uniformity in context representation will remain an issue.

V. CONCLUSIONS

We have discussed some of the challenges we see for a generalized deployment of context aware services, which heavily depend on emergence of standard architectures for pervasive computing. We are actively researching some of these themes. We are also active in service oriented architectures for pervasive computing, an area in which we are aiming at establishing a research network with other industries and universities.

VI. ACKNOWLEDGEMENTS

We thank our research partners which contributed and work with us to develop this vision, including RadioLabs/ University of Tor Vergata, Politecnico di Milano, the PHYSE consortium including Universities of Lancaster and Grenoble, Bull, IBM, ITC-IRST, Agent Oriented Software.

VII. REFERENCES

[1] G. Cortese, F.Davide, A. Detti "eCASA : an Easy Context-Aware System Architecture" IEEE Vehicular Technology Conference October 6-9

[2] Boddupalli, B., F. Al-Bin-Ali, N. Davies, A. Friday, O. Storz and M. Wu. "Payment Support in Ubiquitous Computing Environments". To appear in Proc. of 5th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2003), Monterey Bay, U.S.).

[3] Dey, A. K. "Understanding and using context", Personal and Ubiquitous Computing, Vol 5, No. 1, pp 4-7, 2001

[4] Amy L. Murphy, Gian Pietro Picco, and Gruia-Catalin Roman. Lime: A Middleware for Physical and Logical Mobility. In Proceedings of the 21 st International Conference on Distributed Computing Systems (ICDCS-21), May 2001.

[5] N. Blefari Melazzi , G. Bianchi, G. Cortese, S. Kapellaki, K. Kawamura, C. Noda, S. Salsano, I. S. Venieris. "The Simplicity project: easing the burden of using complex and heterogeneous ICT devices and services - Part I: Overall Architecture". Submitted.

[6] VICom project – http://www.vicom-project.it/

[7] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron. SCRIBE: A large-scale and decentralized application-level multicast infrastructure. IEEE Journal on Selected Areas in communications (JSAC), 2002. To appear.

[8] Kubiatowicz, J., Bindel, D., Chen, Y., Czerwinski, S., Eaton, P., Geels, D., Gummadi, R., Rhea, S.,

Weatherspoon, H., Weimer, W., Wells, C., and Zhao, B. OceanStore: An architecture for global-scale persistent storage. In *Proceeedings of the Ninth international Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2000)* (Boston, MA, November 2000), pp. 190-201.

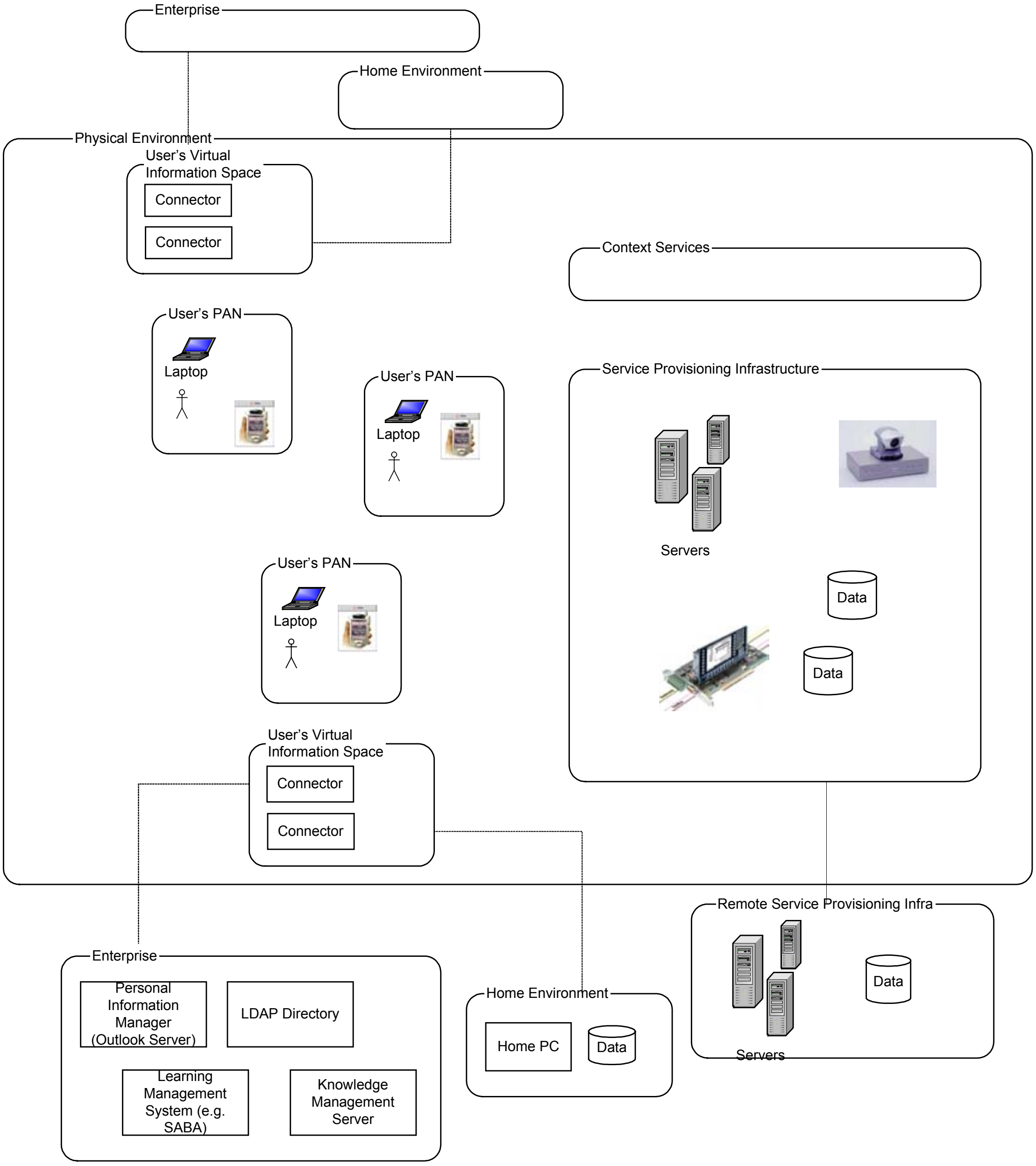

Figure 1 Physical Service Environment (Draft Picture)