Ubiquitous Computing and Databases

George Roussos
University of London, UK

Michael Zoumboulakis
University of London, UK

INTRODUCTION

The concept of the so-called ubiquitous computing was introduced in the early 1990s as the third wave of computing to follow the eras of the mainframe and the personal computer. Unlike previous technology generations, ubiquitous computing recedes into the background of everyday life:

*It activates the world, makes computers so imbedded, so fitting, so natural, that we use it without even thinking about it, and is invisible, everywhere computing that does not live on a personal device of any sort, but is in the woodwork everywhere.* (Weiser & Brown, 1997, p. 81)

Ubiquitous computing is often referred to using different terms in different contexts. Pervasive, 4G mobile, and sentient computing or ambient intelligence also refer to the same computing paradigm. Several technical developments come together to create this novel type of computing; the main ones are summarized in Table 1 (Davies & Gellersen 2002; Satyanarayanan 2001).

BACKGROUND

One of the major challenges in turning the ubiquitous computing vision into reality is the development of distributed system architectures that will support effectively and efficiently the ability to instrument the physical world (Estrin et al., 2002; National Research Council, 2001). Such architectures are being developed around two core concepts: self-organizing networks of embedded devices with wireless communication capabilities and data-centricity. To augment physical artifacts with computational and communications capabilities, it is necessary to enable miniaturized hardware components capable of wireless communication. However, these same characteristics that allow for instrumentation of physical objects also impose significant constraints. Systems architectures require significant changes due to the severely limited resources available on these devices. One possible solution is offered by the emergence of data-centric systems. In this context, data-centric refers to in-network processing and storage, carried out in a decentralized manner (Estrin et al., 2000).

One objective of data-centricity is to let systems exploit the anticipated high node densities to achieve longer unsupervised operating lifetimes. Indeed, smaller form factor wireless sensor nodes have limited resources and often cannot afford to transfer all the collected data to the network edge and forward to centralized information processing systems. A practical example of the data-centric approach for network routing is directed diffusion (Intanagonwiwat, Govindan & Estrin, 2000). This mechanism employs in-network processing by routing data along aggregation paths, thus, removing the need for an address-centric architecture. It exploits data naming as the lowest level of system organization and supports flexible and efficient in-network processing.

In summary, databases have a dual role to play in ubiquitous computing: in the short term, they need to provide the mapping between physical and virtual entities and space in a highly distributed and heterogeneous environment. In the longer term, database management systems need to provide the infrastructure for the development of data-centric systems. Each of the two phases is discussed in turn in the following sections.

DATABASES IN UBIQUITOUS COMPUTING

As noted earlier, a key requirement of ubiquitous computing is the use of contextual information to adapt system behavior on-the-fly and deliver services that fit the specific user situation, a feature that has become known as context-awareness (Abowd & Mynatt, 2000). Context-aware systems need to combine three elements: details of the physical environment must be sourced from embedded sensors, user profiles must be retrieved from distributed repositories holding fragments of identity information, and these data must be correlated to a domain-specific knowledge model that can be used to
reason about the specific situation. In these circumstances, databases must support two aspects of the virtual-physical integration: on the one hand, provide mappings between physical artifacts and associated digital resources and, on the other, hold profile, historical, and state data. This functionality is best illustrated in a case study, for example, ubiquitous retail (Kourouthanassis & Roussos, 2003).

Database Systems Challenges

The introduction of ubiquitous computing technologies in a retail environment can help develop a number of novel applications that can transform the shopping experience and at the same time improve consumer satisfaction. A core ingredient for such ubiquitous retail systems is the augmentation of consumer products with information processing capabilities and the development of supporting infrastructures.

To this end, several new technologies have been recently introduced. The Electronic Product Code (EPC) is a globally unique identifier that is attached to a particular product item stored in a radio frequency identification (RFID) tag. Unlike barcodes that are used to identify product classes, the EPC distinguishes a particular product instance, for example a specific can of cola rather than the class of cola cans, and may be used to trace additional information about it. This information is retrieved by querying the Object Naming Service (ONS), which matches the EPC to the production server that holds information about the item in Product Markup Language (PML), for example, its ingredients and date of production (Mealling, 2003). Using these technologies, a smart shopping cart will identify the products placed into it using the EPC, retrieve additional information about the product.