Extrovert Gadgets

Configuring and reconfiguring ubiquitous applications composed of communicating artifacts.

BY **Achilles Kameas** AND **Irene Mavrommati**

IN PROJECT E-GADGETS (www.extrovert-gadgets.net), we explore ubiquitous computing applications composed from e-Gadgets, that is, artifacts that intrinsically possess sensing, acting, processing, and communication abilities. e-Gadgets are functionally autonomous objects, having both physical and digital substance, that can be freely associated with several different methods, and collectively function as ubiquitous applications (GadgetWorlds). In that sense, the e-Gadgets approach differs from Smart-Its [1] in that it visualizes all inherent and learned services, capabilities, and properties of the object and from information appliances [4]; in that objects may be uniformly regarded at different levels of granularity, from e-Gadget components to e-Gadgets to GadgetWorlds.

The vision of e-Gadgets is to give users the ability and knowledge to configure and reconfigure objects and allow for their creativity to emerge in a ubiquitous environment. Thus, people become co-creators as they create personalized ubiquitous applications and adapt their surroundings to serve the tasks at hand. This vision is expressed in Gadgetware Architectural Style (GAS) [2]—a generic framework shared by both artifact designers and users for consistently describing, using, and reasoning about such applications. GAS architecture provides a uniform, technology-independent upper layer (providing the end-user functionality) and a lower layer that accommodates existing standards in distributed system architectures (such as UPnP, Jini, and so on) and communication protocols (such as WiFi, Bluetooth, among others).

Two important concepts in GAS are plugs and synapses. Plugs are software classes that make an e-Gadget's properties, capabilities, and services visible to external actors (people, other e-Gadgets, agents). Composition of an application is effected through the definition of synapses (associations) between pairs of plugs. People can apply these intuitive concepts though the Plug/Synapse model, a conceptual abstraction that enables uniform access to e-Gadget services, capabilities, and properties and allows users to compose and use applications that realize a collective behavior in a high-level programming manner. Another benefit of this approach, apart from simplicity and universal applicability, is that system design is largely provided ready to the end user or the application designer, because the domain and system concepts are specified in the generic architecture.

In order for the ubiquitous application to function as a system, each artifact must run GAS-OS—the operating system and middleware that manages resources shared by e-Gadgets, determines their software interfaces, and provides the underlying mechanisms that enable e-Gadgets to communicate. GAS-OS provides synapse management, discovery, and routing services at the application level; it supports the composition of e-Gadgets, without having to access any code that implements their interfaces. Compatibility is improved with the use of a layered ontology to match descriptions of plugs. The ontology encodes a common basic structure and reference terms (core ontology) and levels where services, applications, and decision-making procedures are defined. GAS-OS is written in Java and supports IP-based communication using (without being bound to) IEEE 802.11g.

To evaluate the approach and implement the concepts, more than 12 sample e-Gadgets and two different instantiations of an editing tool (the Editor) have been created. The Editor was developed as an end-user tool to facilitate the composition of applications. The purpose of the Editor is threefold: to indicate and make visible the available objects (and their plugs) and applications; to establish synapses and form new applications; and to assist with debugging, editing, and servicing, among others. Two versions of the Editor have been created, one for laptops and one for PDA-like devices. The Editor and sample e-Gadgets have been evaluated in the course of several end-user trials; main conclusions indicated that users were keen to become "creators" of ubiquitous applications if they were given simple metaphors and usable (but not necessarily intelligent) tools [3]. C

References

1. Holmquist, L.E., Mattern, F., Schiele, B., Alahuhta, P., Beigl, M., and Gellersen, H.W. Smart-Its friends: A technique for easily establish connections between smart artifacts. In *Proceedings of UBICOMP 2001* (Atlanta, GA, Sept. 2001).
2. Kameas, A. et. al. An architecture that treats everyday objects as communicating tangible components. In *Proceedings of the International Conference on Pervasive Computing and Communications* (Texas, Mar. 2003), 23–26.
3. Mavrommati, I., Kameas, A., and Markopoulos, P. An editing tool that manages devices associations in an in-home environment. *Personal and Ubiquitous Computing 8,* 3-4. ACM Press/Springer-Verlag, London, U.K., 255–263.
4. Norman, D.A. *The Invisible Computer.* MIT Press, Cambridge, MA, 1998.

Achilles Kameas (kameas@cti.gr) is an assistant professor at the Hellenic Open University and unit manager of the DAISy group at the Computer Technology Institute, Patras, Hellas.
Irene Mavrommati (mavrommati@cti.gr) is a project manager and interaction designer of the DAISy group at the Computer Technology Institute & University of the Aegean, Patras, Hellas.