Enabling context-aware HCI with reusable components

***Nearchos Paspallis*[1], *Georgia Kapitsaki*[2], *Achilleas Achilleos*[2]**

npaspallis@uclan.ac.uk, {gkapi, achilleas}@cs.ucy.ac.cy

1. **University of Central Lancashire—Cyprus, 12-14 University Avenue, Pyla, CY-7080**
2. **University of Cyprus, 75 Kallipoleos Street, Nicosia, CY-1678**

Abstract

The advancement of technology has to a great extend enabled the vision of ubiquitous computing: Data and computation processing are remotely available via cloud infrastructures, and they are universally accessible via the UI of multiple devices (including smartphones, tablets and desktops) which are nearly always connected via wired and wireless networks (Weiser, 1991). Proliferation of smartphone devices has also boosted the formation of context-aware applications and services, where the context of the devices and of the users are implicitly utilized to improve the end-user experience by means of intelligently selected data or optimal device configuration (Dey, 2001).

We have developed the RSCM (*Really Simple Context Middleware*) as a means of enabling quick prototyping and realization of context-aware applications via reusable components (Paspallis, 2013). This middleware-based system builds on algorithms which automatically select and activate self-contained context sensors. RSCM is built for and extends the standard Android architecture. While the Android system already provides support for accessing standard, low-level context data such as geo-location, instant acceleration, compass, etc., it requires developers to build their own, custom solutions for acquiring higher-level context information such as location-at-home-or-work, user-walking-or-running-or-driving, etc.

A proof of concept of the middleware is available, and a number of applications have already utilized the middleware to build context-aware applications. A set of standard context plugins have been made available, including location and battery sensors. Advanced context reasoners are also being developed: First, a location-home-or-work reasoner is implemented. It works by keeping track of the users' location and by inferring whether their location is at home or at work by computing the frequency of specific locations at the standard periods where they are expected to be at home or at work. Second, an intelligent battery monitor is warning the user to charge their phone not just when the battery level is below a certain threshold level, but also when it is expected that the user will be moving to an area without access to the grid (e.g. on transit). This is realized with another reasoner, which keeps track of the users' behaviour, like what time they leave for work and what time they get back, the rate at which they consume battery while on transit, etc.

One of the main advantages of the RSCM system is the inherent reusability of the underlying context plugins. Instead of encapsulating the plugins in individual libraries (e.g. JAR files), they are modelled

as reusable components which can be dynamically looked-up for and installed on an Android device (i.e. as a standard, market published APK). To fully realize this, a component repository has been developed on top of Google's App Engine, providing both a user-friendly web page, and a computer-friendly JSON-formatted listing. Developers are encouraged to browse and utilize existing plugins as well as to add their own listings.

The RSCM is under continuous development. As an open source project, everyone is invited to use it, and contribute to its source code. The project provides a number of tutorials, as well as sample plugins and context-aware applications.

Bibliography

Dey, A. (2001, Feb.). Understanding and Using Context. *Personal and Ubiquitous Computing, 5*(1), 4-7.

Paspallis, N. (2013). Retrieved from Really Simple Context Middleware (RSCM): https://code.google.com/p/rscm/

Weiser, M. (1991, Feb.). The Computer for the 21st Century. *Scientific American*, pp. 94-104.