Position Summary.
Middleware for Mobile Computing: Awareness vs. Transparency

Licia Capra, Wolfgang Emmerich, and Cecilia Mascolo
Department of Computer Science
University College London
Gower Street, London WC1E 6BT, UK
{L.Capra|W.Emmerich|C.Mascolo}@cs.ucl.ac.uk

Abstract

Middleware solutions for wired distributed systems cannot be used in a mobile setting, as mobile applications impose new requirements that run counter to the principle of transparency on which current middleware systems have been built. We propose the use of reflection capabilities and meta-data to pave the way for a new generation of middleware platforms designed to support mobility.

1. The Rationale

The increasing popularity of wireless devices, such as mobile phones, personal digital assistants and the like, is enabling new classes of applications that present challenging problems to application designers. These devices face temporary loss of network connectivity when they move; they discover other hosts in an ad-hoc manner; they are likely to have scarce resources, such as low battery power, slow CPU speed and small amounts of memory; and they are required to react to frequent and unannounced changes in the environment, e.g. variable network bandwidth.

Middleware technologies [2] have been designed and successfully used to support the development of stationary distributed systems built with fixed networks. Their success has been mainly due to their ability of making distribution *transparent* to both users and software engineers, so that systems appear as single integrated computing facilities.

However, completely hiding the implementation details from the application becomes both more difficult and makes little sense in a mobile setting. Mobile systems need to quickly detect and adapt to drastic changes happening in the environment. A new form of *awareness* is needed, as opposed to transparency, to allow application designers to *inspect* the execution context and *adapt* the behaviour of middleware accordingly.

2 Research Directions

We believe that reflection and metadata can be successfully exploited to develop middleware targeted to mobile settings. Through metadata we obtain separation of concerns, that is, we distinguish what the middleware does from how the middleware does it. Reflection is the means that we provide to applications in order to inspect and adapt middleware metadata, that is, influence the way middleware behaves, according to the current context of execution.

We have developed XMIDDLE [3], a middleware for mobile computing that focuses on synchronization of replicated XML documents. In order to enable application-driven conflict detection and resolution, XMIDDLE supports the specification of conflict resolution policies through meta-data definition using XML Schema.

The following step has been the definition of a global model for the design of mobile middleware systems, based on the principles mentioned above. In [1], we have discussed a reflective conceptual model and a reflective architecture of middleware systems targeted to support mobile applications that call for context-awareness, where by context we do not mean only location but everything in the physical environment that can influence the behaviour of the application, such as memory and battery power.

References

[1] L. Capra, W. Emmerich, and C. Mascolo. Reflective Middleware Solutions for Context-Aware Applications. Technical Report RN/01/12, UCL-CS, 2001. Submitted for Publication.

[2] W. Emmerich. Software Engineering and Middleware: A Roadmap. In *The Future of Software Engineering - 22nd Int. Conf. on Software Engineering (ICSE2000)*, pages 117–129. ACM Press, May 2000.

[3] C. Mascolo, L. Capra, and W. Emmerich. XMIDDLE: A Middleware for Ad-hoc Networking. 2001. Submitted for Publication.