Mobile Middleware for Heterogenous Environments

Olaf Droegehorn, Bjoern Wuest, Christian Deist, Klaus David

University of Kassel
Chair of Communication Technology
Wilhelmshöher Allee 73
34121 Kassel, Germany

Introduction

For the future success of the upcoming wirless networks, like UMTS, B3G and 4G, a major change in the used business models is foreseen. To enable users to provide services and content in every location and context is one of the most driving factors.
In the literature, several approaches for the design of a middleware platform have been reported. It is obvious, that these platforms are on the one hand optimised for a special application domain and on the other hand are assuming a typical client/server relation. In this paper, we design an open architecture for modern middleware implementations.

Framework for Applications in Mobile Environments

To enable seamless operation of different middleware characteristics and even to allow the adoption of the currently running middleware, a common layer of functionality is required. This layer, we call the "Framework for Application in Mobile Environments 2", provides a framework for specialized middleware implementations that remain flexible while yet powerful.
Unlike most framework designs like, FAME2 does not dictate any network topology or protocols to use. FAME2 enabled platforms use the device and network as needed, not demanding any hardware or infrastructure to adapt to the needs of the framework. This opens the framework for a multi-domain environment in the vertical market, providing optimal implementations for each application scenario while still remain interoperable. The core provides this flexibility and openness by providing a common, standard and slim interface to services registered at and interacting with the core.
To achieve the portability and flexibility of the middleware required in the application scenario above, we separate the middleware into two layers. One layer composed of local functionalities while the second layer hosts distributed functionalities. To have an anchor on each device running the middleware, FAME2 consists of an omnipresent core element that is an interface to the management of the provided middleware. Figure 1 depicts the architecture, embedded in a Java environment. The local layer of the middleware is made up of a single instance of the core as well as a set of modules. The distributed layer consists of components. Modules as well as components can communicate with the core as well as with each other. Communication between modules and components is defined by the modules and components to allow optimal implementations. This applies to other mechanisms like scalability, information representation etc., too.

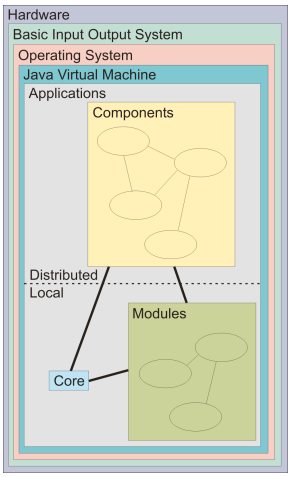

Figure 1: Architecture of FAME2 within a JVM

Since the core is the only common part of all FAME2 enhanced devices, the presence of modules and components depend on set up and may differ between devices while sharing the same basic environment. That is, a backbone server may provide services accessed by thousands of users simultaneously while a PDA may provide services accessed by only a few users. However, both share the same platform and basic functionality and permit easy interoperation between them. The borders between client and server blurs, the only limiting factor is the capacity and power of the participating devices. In contrast to pure peer-to-peer networks, this novel concept leverages the flexibility of peer-to-peer networking while still allow for integration of the proven and well known client/server networking.

Conclusion and Outlook

A novel approach of a Framework for the development of mobile applications using a component based middleware architecture has been introduced. The proposed system is able to use well known techniques for software development (like Java/Jini/XML) and integrates new concepts like the combination of server-centric and fully distributed communication models. The next steps of the platform development will integrate more service components like billing, security issues etc.

References

[1] **Robert J. Stets, Galen C. Hunt, Michael L. Scott**, *Component-Based APIs for Versioning and Distributed Applications*, IEEE Computer, July 1999, p. 54ff

[2] **Savitha Srinivasan**, *Design Patterns in Object-Oriented Frameworks*, IEEE Computer, February 1999, p. 24ff

[3] **Paul Chambers**, *Application Servers: Wired for Wireless*, GemStone Systems Ltd., September 2000, http://e-serv.ebizq.net/dvt/chambers_1.html

[4] **Olaf Drögehorn ET. Al.**, *Towards a Location- and Context Aware Middleware*, International Conference on Internet Computing, June 2001, Las Vegas

[5] **Cynthia Della Torre Cicalese, Shmuel Rotenstreich**, *Behavioral Specification of Distributed Software Component Interfaces*, IEEE Computer, July 1999, p. 46ff