3APL-M Platform for Deliberative Agents in Mobile Devices

Fernando Koch
Institute of Information and Computing Sciences
Utrecht University, Utrecht, The Netherlands
fkoch@acm.org

ABSTRACT

The demonstration presents the 3APL-M platform for building deliberative multi-agent systems whose components execute on resource bounded devices.

Keywords

agents, BDI, mobile devices

1. INTRODUCTION

The benefit of mobile computing is to have an always-present software application able to process contextual information and to supply the user with the right information at the right time [6]. Agent paradigm seems to offer a set of features that are closely aligned with the requirements of intelligent mobile applications [4], provisioning to the requirements for intelligent personal assistant and mobile computing.

However, the problem is how to provision agent-based application equipped with enhanced inference capabilities (for the sake of the discussion), such as Belief-Desire-Intention (BDI), able to execute in the restricting mobile computing environment.

In this demonstration, I present the 3APL-M [1] platform for programming deliberative multi-agent systems whose components execute on mobile devices. The solution takes advantage of the 3APL language and definitions and provides an interface to integrate the applications to the external world. The library is distributed for the Java 2 Micro Edition (J2ME) programming platform.

2. SOLUTION

The 3APL-M platform architecture is presented in Figure 1. The main features are: sensor and actuator modules, which provide the interface to integrate to context-awareness and content delivery solutions; the 3APL machinery, which includes the infrastructures for the B.D.I. based inference systems, and; the communicator module, which provides the support for communication in a multi-agent system. The modules are explained below:

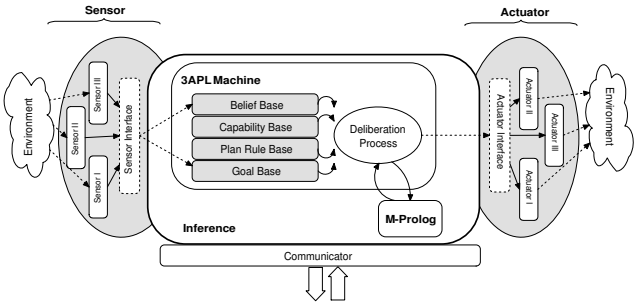

Figure 1: 3APL-M Architecture

- the *3APL machine* encapsulates the 3APL language components and provides the programming interface for the integration of the logic structures to the Java programming language;
- the *belief*, *capabilities*, *goal* and *plan rules* modules are implementations of the 3APL structures. These elements are part of the 3APL machinery and provide the internal data and processing structures for the platform;
- the *deliberation process* is the implementation of the executive module (deliberation cycle);
- the *plan base* is the data structure that holds the list of current plans generated by the deliberation process;
- the *m-prolog* is an implementation of the PROLOG language engine, optimized to be used for the low-level inference processing in 3APL-M. It also holds the data for the belief base.
- the *sensor* and *actuator* are the programming interfaces for the integration of the 3APL-M machinery to the external world. The *sensor* module provides the infrastructure for the creation of context-aware application (i.e. environmental sensors) and system input (i.e., device's keyboard). The *actuator* module provides the means for content delivery (i.e., integration to the device's display interface) and acting upon the environment.

AAMAS'05, July 25-29, 2005, Utrecht, Netherlands.

- the *communicator* module provides the is the generic interface for the data exchange infrastructure, required for multi-agent system module integration and communication to external services. The module provides internal support for FIPA communication [2][3], however any other protocol or data representation can be plugged in the system through the programming interface.

The 3APL-M platform works as a library loaded in the distribution package. This library supplies the application-programming interface (API) for the 3APL machine modules. The Java application makes calls to the library's modules for loading information, configuring the deliberation engine and executing the applications.

For detailed information about programming in 3APL-M, we refer to the documentation and source code examples available at the project's web-site [1].

3. DEMONSTRATIONS

I present three live demos developed with 3APL-M:

1. the **Block World** example, where 3APL-M integrates to Java to create an application that interacts to device's interface.
2. the **Wumpus World game** example, where 3APL-M is applied to this renowned artificial intelligence problem.
3. the **Mobile Personal Assistant** example, where 3APL-M is applied to implement concepts of interest based negotiation (IBN) [5].

For detailed information about 3APL-M, demonstrations and software we refer to project's web site [1].

4. CONCLUSION

3APL-M provides the support technology to develop deliberative multi-agent systems to be executed in mobile computing devices. The main features are the *sensor* and *actuator* modules, which provide the interface to integrate to context-awareness and content delivery solutions; the 3APL machinery, which includes the infrastructures for the B.D.I. based inference systems, and; the communicator module, which provides the support for communication in a multi-agent system. Hence, the platform provides the infrastructures for the technologies required by the new generation of mobile applications: context-sensitiveness, mental modelling, local processing, and pervasive content delivery. The B.D.I.-based inference module provides the solutions for applications capable of creating mental models and to represent the human thought structures.

The platform delivers a development environment compatible with the 3APL language structures. The demonstration applications proved that the resulting applications are small enough to be deployed on small devices with 20Mhz CPU and less than 512Kb RAM. The platform is compatible with Java 2 Micro Edition (J2ME) development and running environment, which has a large development community. Consequently, several development environments, platforms and programming libraries are commercially available. The strength of J2ME is industry adoption and to be Java-compatible, thus this running environment is present in a myriad of commercially available mobile computing devices.

5. REFERENCES

[1] 3APL-M web-site, http://www.cs.uu.nl/3apl-m.

[2] M. Aparicio, L. Chiariglione, E. Mamdani, F. McCabe, R. Nicol, D. Steiner, and H. Suguri. FIPA - intelligent agents from theory to practice. *Telecom 99*, October 1999.

[3] F. Bergenti, A. Poggi, B. Burg, and G. Claire. Deploying FIPA-compliant systems on handheld devices. *IEEE Internet Computing*, 5(4):20–25, 2001.

[4] F. Koch and I. Rahwan. The role of agents in intelligent mobile services. In M. W. Barley and N. Kasabov, editors, *Intelligent Agents and Multi-Agent Systems: 7th Pacific Rim International Workshop on Multi-Agents, PRIMA 2004, Auckland, New Zealand, Revised Selected Papers*, volume 3371 of *Lecture Notes in Computer Science*. Springer-Verlag GmbH, March 2005.

[5] T. Rahwan, T. Rahwan, I. Rahwan, and R. Ashri. Towards a mobile intelligent assistant: Agentspeak(l) agents on mobile devices. In P. Giorgini and M. Winikoff, editors, *Proceedings of the 5th International Bi-Conference Workshop on Agent Oriented Information Systems (AOIS)*, 2003.

[6] D. Saha and A. Mukherjee. Pervasive computing: A paradigm for the 21st century. *IEEE Perspectives*, 36(3):25–31, Mar 2003.