Towards an Agent-Based Infrastructure for Distributed Virtual Organizations

Michael Berger, Bernhard Bauer, Michael Watzke
Siemens AG, Corporate Technology, Intelligent Autonomous Systems Group
D–81730 Munich, Otto-Hahn-Ring 6, Germany
[Michael.Berger | Bernhard.Bauer | Michael.Watzke]@mchp.siemens.de

Abstract

Agent technology has the potential to play a key role in building and supporting virtual enterprises. To host and enable agents an open infrastructure is needed. Beside standards, such an infrastructure should be generic and deployment independent to deal with the heterogeneity of the computing and information technology, e.g. by supporting devices like mobile phones, PDAs, and personal computers as well as different kinds of mobile and fixed networks. In this paper we give an overview over the agent platform LEAP, developed for small end user devices. We show the main issues in downsizing it to PDAs / mobile phones and applying agent technology in the area of virtual team support.

1 Introduction

Software agents are an innovative technology for the efficient and intelligent realization of complex, distributed, and highly interactive heterogeneous systems and applications (see e.g. [1], [2]). Agent technology has the potential to play a key role in building and supporting virtual organizations and virtual mobile teams by automating daily processes, enriching higher level communication and enabling a more intelligent service provision, e.g. by personalization and service integration to deal with the enlarging amount of information.

In this paper we give an overview over the Lightweight Extensible Agent Platform (LEAP), developed for small end user devices based on JADE [3]. JADE is a FIPA [4] compliant open source software, distributed under the LGPL license [5]. We show the main issues in making the platform deployment independent to deal with the heterogeneity of the computing and information technology. Furthermore, this paper describes our activities in applying agent technology in the area of virtual team support.

2 LEAP Platform Development Process

LEAP emerged as an independent development branch of JADE (see [3], [6]) and will be merged with the JADE mainstream and will replace JADE's kernel. As such, LEAP concentrates on lightweight and extensible aspects, whereas JADE continues independently its evolution towards environmental functions such as monitoring, visualization, ontology's, and policies. The JADE APIs remain unchanged. Therefore, all existing applications will continue to run as before. In addition, developers can use LEAP to migrate existing applications or to develop a new generation of applications for small wireless devices.

In parallel to that, the agent application development tool ZEUS [7] will be evolved in order to compile into LEAP agents to simplify the development of applications.

3 Changes to Openness and Scalability

The LEAP activity focuses on restructuring the JADE core, compliant to Java 2 Standard Edition (J2SE) [8], in order to match the LEAP requirements. That means the new core is re-integrated into JADE to obtain a FIPA compliant platform that is:

- *Lightweight* enough to be deployed on small devices, such as mobile phones, supporting only a KVM [8] with J2ME/CLDC [8], instead of a standard JVM [8].
- *Transport layer independent* and in particular supporting transport protocols suitable for wireline and wireless environments, thus providing an homogeneous layer to agent application developers.
- *Extensible* such that, when deployed on a powerful machine, it can provide optional functionality such as agent mobility and platform management GUIs.

The solutions will be discussed in detail below.

3.1 J2ME/CLDC Compatibility

The analysis of the JADE classes (J2SE-compliant) have shown that there were incompatibilities with the J2ME/CLDC API. To solve these incompatibilities the first step was to set up a CLDC-compliant version of LEAP. The detected incompatibilities can be divided into two kinds of classes: some J2SE classes which are not supported at all by CLDC and some J2SE classes which are only partially supported (some methods are missing).

Beyond changing the implementation according to the underlying Java platform, additional work has been done on the JADE platform to minimize the footprint.

3.2 Transport Layer Independence

The requirement to be environment independent leads to the exchange of FIPA messages with the outside world (inter-platform exchange) over one or more Message Transport Protocols (MTPs). The message exchange between containers within the platform (intra-platform exchange) has to be carried out over one or more Internal Transport Protocols (ITPs). Within LEAP the following changes were made to support these requirements:

- The RMI [9] intra-platform calls were changed such that from the platform side remote method invocations are performed. The RMI invocations implementing intra-platform exchanges of information between containers were replaced by commands (with type identifiers and several parameters), with the same semantics.
- The command dispatcher, responsible for the dispatching of commands from one agent container to another one, selects an appropriate transport protocol to deliver a command and does the serialization of the involved objects. Due to the fact that the CLDC configuration of J2ME does not support any "built-in serialization", there was the need to develop a surrogate serialization mechanism to enable object transportation. This serialization is done similar to the standard Java serialization for objects being sent via the command structures.

3.3 Extensibility

We categorized the components of the platform as part of the LEAP core or as LEAP optional functionality. The components belonging to the LEAP core are needed by all profiles and they are meant to implement the basic platform functionality. Components containing optional functionality are not required by all profiles and are intended to work only on devices with sufficient memory and processing power. With that, the LEAP platform is open and third-party developers can use the extension mechanisms to implement own functionality, divided in:

- *Platform extensions:* intended to provide new general functionality to cope with application-specific needs, like new transport protocols.
- *Platform integration:* intended to provide application-specific code, like new ontology's.

4 Applications and Field Trials

In order to demonstrate and evaluate some of the benefits that LEAP will bring to users, companies and their customers, two field trials will be conducted. One will take place with ADAC (a German company providing road side assistance by "yellow angels") in Germany, the other with installation and repair engineers from BT's network maintenance in the UK.

The "yellow angels" and the installation and repair engineers will be able to use the LEAP application in the field via mobile devices (PDAs / mobile phones). The emphasis will be on helping team members to receive work requests; synchronize tasks; get dynamic guidance on the best route; retrieve documentation; reach other team members to get help and information; exchange tasks on the fly; and arrange convenient meeting places, based on personal preferences and current locations.

Both will be based on a LEAP application that is designed to meet the needs of mobile workers operating within virtual teams, and that uses generic agent-based services. Initially, three agent-based services will be built knowledge / travel management, and work co-ordination.

5 Evaluation and Conclusion

The project LEAP[1] [10] reached its first milestone (LEAP V1.0) by running FIPA compliant agents on PDAs in December 2000. At the moment the LEAP platform is running on: Palm Vx, Quartz emulator, Psion 5MX coupled through IrDA and a mobile phone to the Internet, and Siemens-MIDP emulator for mobile phones.

LEAP V2.0 (which will physically run on mobile phones) will be ready in September 2001 and will be released under the LGPL license agreement. The applications will be implemented in autumn 2001 and the Field Trials will be conducted in May 2002.

References

[1] Maes, P.: *Modeling Adaptive Autonomous Agents.* In: Langton, C. (ed.): Artificial Life Journal, Vol. 1, No. 182, MIT Press, pp. 135-162, 1994.

[2] Müller, J.P.: *The Design of Autonomous Agents: A Layered Approach.* Lecture Notes in Artificial Intelligence, Vol. 1177. Springer-Verlag, Heidelberg, 1997.

[3] http://sharon.cselt.it/projects/jade/

[4] http://www.fipa.org/

[5] http://www.gnu.org/copyleft/lgpl.html

[6] Berger, M. et al.: *A Scalable Agent Infrastructure.* In: Proceedings of the Workshop on Agent Infrastructures at Agents'01 conference, Montreal, 2001.

[7] http://www.labs.bt.com/projects/agents.htm

[8] http://java.sun.com/aboutJava/

[9] http://java.sun.com/products/jdk/rmi/

[10]http://leap.crm-paris.com/

[1] The platform, the services, the applications, and the field trials are part of the project LEAP founded by the European Commission through the grant IST-1999-10211. The authors would like to thank the other partners of the project: ADAC, Broadcom, BT, CSELT, Motorola, and University of Parma.