Java Agent Development Framework

Sujeet Kumar,   Utkarsh Kumar

Student, Information Technology Engineering, MDU University , Delhi , India

sujeetkumar0110@gmail.com ; utkarsh.kumar10@gmail.com

ABSTRACT

Java Agent Development Framework (JADE) is a software in JAVA, which is used for the development of agents, implemented in JAVA. It is a distributed agents platform, which has a container for each host where we run the agents. Additionally the platform has various debugging tools, mobility of code and content agents, the possibility of parallel execution of the behavior of agents, as well as support for the definition of languages and ontologies. In this paper, we will be having a detailed study on the functioning of JADE and we will put forward our conclusions on how its functioning can be enhanced.

Keywords:

Java Agent Framework Environmen; JADE ; Architecture ; Conclusion

1. Introduction

Agent-based technologies are still in their initial stages and rare truly agent-based systems have been realized. Agent-based technologies cannot realize their full potential, and will not become global, until standards to support agent interoperability are available and used by agent developers and adequate and appropriate environments for the development of agent systems are available.

In this paper, we present JADE (Java Agent Development Framework) that is a software framework to develop agent applications to be a good and a useful method for developing agents based technologies. JADE main features and, the architecture of the agent platform are discussed further in the paper.

2. JADE

JADE (Java Agent Development Framework) is a Java framework for the development of distributed multi-agent applications. It represents an agent middleware providing a set of available and easy-to-use services and several graphical tools for debugging and testing. One of the main objectives of the platform is to support interoperability by strictly following to the FIPA specifications concerning the platform architecture as well as the communication infrastructure. Moreover, JADE is very flexible, usable and adapted to be used on devices with limited resources such as PDAs and mobile phones.

JADE has been globally and widely used over the last years by many academic and industrial organizations varying from tutorials for teaching support in agent-related University courses to Industrial prototyping. As an example, White stein has used JADE to construct an agent-based system for decision-making support in organ

transplant centers.

The JADE platform is open source software, distributed by TILAB (Telecom Italia Laboratories) under the terms of the LGPL license and can be obtained at http:// jade.tilab.com. Since May 2003, the International JADE Board has been responsible and the main reason for supervising the management of the project. Currently, the JADE Board consists of five members: TILAB, Motorola, Whitestein Technologies AG, Prefatory, and France Telecom.

3. Internal architecture of the JADE run-time
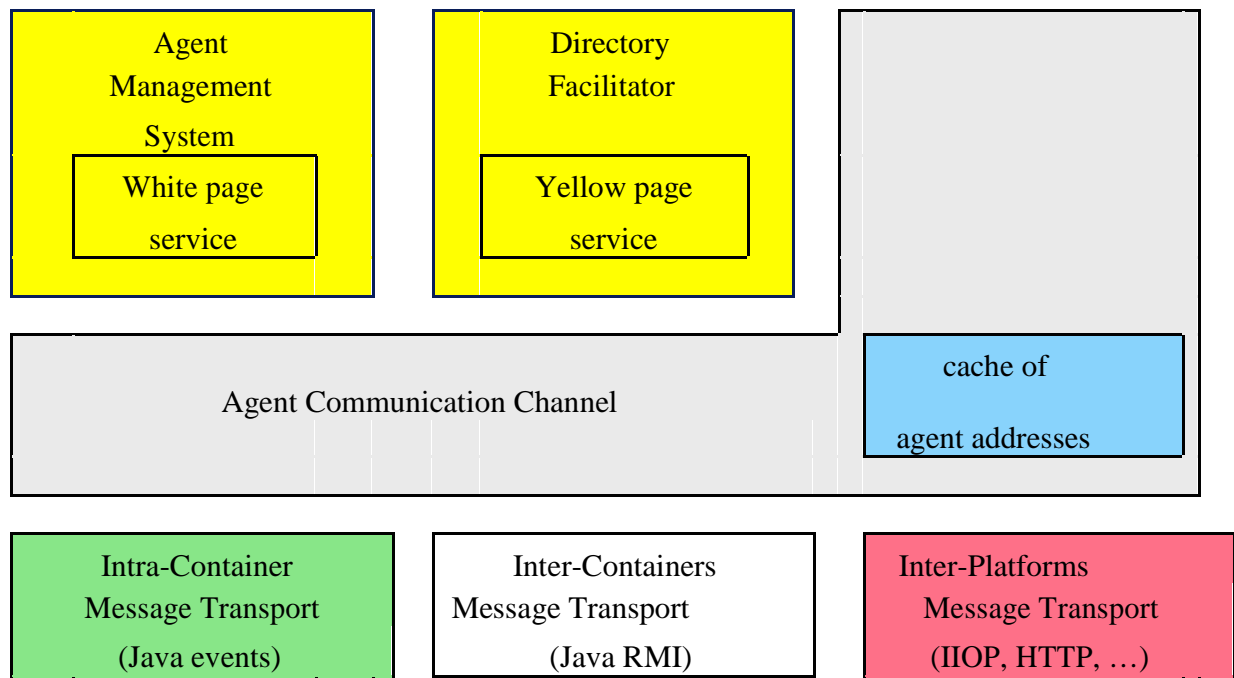
A JADE platform comprises of agent containers that can be distributed over the network. Agents live in containers which are the Java process that provides the JADE run-time and all the services required for hosting and executing agents. There is a special container, called the *main container*, which represents the bootstrap point of a platform: it is the first container to be launched and all other containers must join to a main container by registering with it. The programmer identifies containers by simply using a logical name by default the main container is named 'Main Container' while the others are named 'Container-1', 'Container-2', etc. Command-line options are available to override default names.
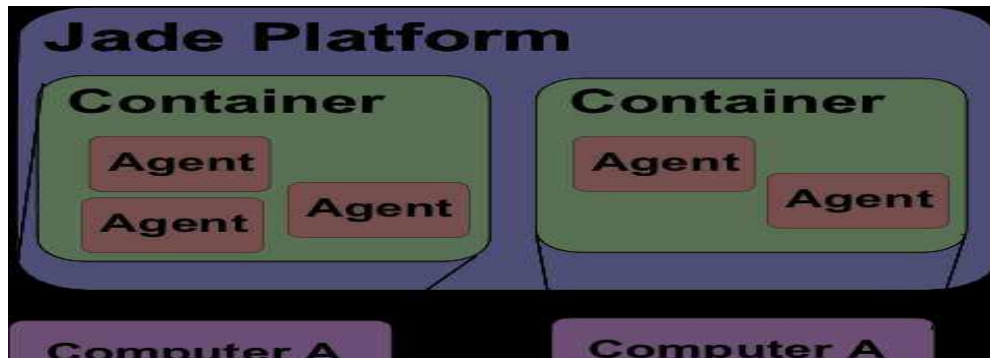
- The Responsibility of the main container is managing the container table (CT), which is the registry of the object references and transport addresses of all container nodes composing the platform and Managing the agent descriptor table (GADT), which is the registry of all agents present in the platform, including their current status and location

4. JADE Packages

There are different packages present in JADE for example

- jade.core.event:- this package implements the distributed event notification service
- jade.core.management:- this package implements the distributed agent life-cycle management service
- jade.core.messaging:- this package implements the message distribution service
- jade.core.mobility :- this package implements the mobilitity and cloning service
- jade.content and its sub-packages contain the collection of classes that

support creating and manipulating expressions according to a given content and ontology
- jade.domain contains the implementation of the AMS and DF agents

5. Conclusions

JADE is written and represented in Java language and is made by various Java packages, giving application programmers both ready-made pieces of functionality and abstract interfaces for custom, application dependent tasks. Java was the programming language of preference because of its many attractive features, particularly geared towards object-oriented programming in distributed heterogeneous environments; some of these features are Object Serialization, Reflection API and Remote Method Invocation (RMI).
JADE provides its users with standard agent technologies while keeping runtime overheads low. The development of JADE is still continuing and further improvements, enhancements, and implementations have already been planned, included support for agent mobility as specified by FIPA98

References

Java Agent Development Framework *Sujeet Kumar & Utkarsh Kumar*

[1] Broadcom Eireann Research Ltd. ASL – Agent Service Layer. 1998. Available from http://www.broadcom.ie/asl.

[2] B. Eckel. Thinking in Java. , Upper Sandle River, NJ. Prentice Hall, 1998.

[3] T. Finin and Y. Labrou. KQML as an agent communication language. . J.M. In: Bradshaw (ed.), Software Agents, pp. 291-316. Cambridge, MA, 1997.

[4] E.J. Firedman-Hill. Java Expert System Shell. 1998. Available from http://herzberg.ca.sandia.gov/jess

[5] Foundation for Intelligent Physical Agents. Specifications. 1997. Available from *http://www.fipa.org*

[6] M.R. Genesereth and S.P. Ketchpel. Software Agents. Comm. of ACM, 37(7):48-53.1994.

[7] A.D. Lux and D. Steiner. Understanding Cooperation: an Agent's Perspective, in Proc. ICMAS'95. San Francisco, USA. 1995.

[8] H.S. Nwana, D.T. Ndumu and L.C. Lee. ZEUS: An advanced Tool-Kit for Engineering Distributed Mulyi-Agent Systems. In: Proc. of PAAM98, pp. 377-391, London, U.K. 1998.

[9] Object Management Group. 95-11-03: Common Facilities RFP3 Final Draft. 1995. Available from http://www.omg.org/docs/1995/95-11-03.ps.

[10] R.S. Patil, R.E. Fikes, P.F. Patel-Scheneider, D. McKay, T. Finin, T. Gru ber and R. Neches. The DARPA knowledge sharing effort: progress report. In: Proc. Third Conf. on Principles of Knowledge Representation and Reasoning, pp 103-114. Cambridge, MA. 1992.

[11] M. Straßer, J. Baumann and F. Hohl (1997): Mole - A Java based Mo bile Agent System. In: M. Mühlhäuser: (ed.), Special Issues in Object Oriented Programming. dpunkt Verlag, pp. 301-308, 1997.

[12] H. Suguri. COMTEC Agent Platform. 1998. Available from http://www.fipa.org/glointe.htm

[13] K. Sycara, A. Pannu, M. Willia mson and D. Zeng. Distributed Intelligent Agents. IEEE Expert, 11(6): 36-46. 1996.

[14] Developing Multi-Agent System With JADE by Fabio Bellifemine, Giovanni Caire , Dominic Greenwood

[15] *europa.nvc.cs.vt.edu/~cegyhazy/cs6804/cs6804/**JADE**.ppt*

Java Agent Development Framework *Sujeet Kumar & Utkarsh Kumar*