Aglets is to Java as MAPNET is to .NET Framework

Mobile Software Review

by

Caesar Ogole

cogole@gmail.com

Under its own control, a mobile agent (kind of computer program) is able to migrate from one (machine) to another in a distributed communication network. At any one point in time, a mobile agent may be any of the following states: “sleep” (wait) or active (transporting itself). Mobile agents reside in a special machine called an agent server. Since most of the languages/interfaces for writing mobile agent applications are platform-dependent, much research interest is being put on the development of agents that can run on the different existing platforms. IBM’s Java-based aglets are just one of the common mobile agent platforms that offers support for multiple development languages. Also among the Java-based technologies are IKV++ Grasshopper and Tacoma.

The remaining section of this paper is devoted to discussion of a non Java-based mobile agent model. This is one that has been developed by Microsoft Corporation and it is based on Microsoft’s .NET Framework. It is called MAPNET. Written in C#, MAPNET defines a rich library that provides encapsulations to the details of a distributed communications network model. This makes it easier for a developer to come up with mobile agent solutions without having to worry about the underlying nitty-gritty of networking. Like most of the mobile agent platforms, MAPNET implements abstractions to the most important distributed system components including security, serialization, remote invocation, and web services.

Component-wise, MAPNET consists of an Operating System, .NET Common Language Runtime (CLR), Agents and Agent Servers. Depending on the developers design decisions, agents residing in different servers in distributed network can communicate with one another or, alternatively, an agent may transport itself (migrate) from one node to another. (To say it briefly), this kind of object behavior is defined or modeled in a MAPNET class called the Agent class, which happens to be a top hierarchy object that provides abstractions to the core properties and behavior of a mobile agent. Other classes that provide additional functionality such as the mechanisms for agent migration and remote access all extend (directly or
otherwise) from the Agent class. In a nutshell, MAPNET adopts the paradigm of object orientation and its principles such as inheritance for object communication.

MAPNET is not the only .NET-based mobile agent framework available. Another mobile agent toolkit for .NET environment is Ether Yatri .NET. This is a free open source package mainly used for educational purposes. It integrates into Visual Studio. Also available for a .NET Compact Framework is a system called NOMAD. Although this is not a mobile agent toolkit in itself, it integrates some aspects of agent technology so as to enhance the functionalities provided to its (nomadic) users.

None of these models cited here, sadly (may be), has support for wireless connectivity (except the NOMAD, which is by the way not a mobile agent system).

Now we know what to go for. No critical or biased comments this time. The objective of this survey was simply to throw a little light on the existence of .NET Framework’s “aglets”. They exist! And more developments are taking place in and around this environment, but IBM Java aglets is still in its strength, too. Perhaps, this is only a story of diverse technologies that are all governed by standards and specifications such as the MASIF (Mobile Agent System Interoperability Facilities) specification—a self explanatory set of conventional rules in the industry. As to whether one technology is superior over the other is typically contextual.