The Advantages of Using Mobile Agents in Software for Telecommunications

Luis Moura Silva       Leonor Almeida
University of Coimbra              SIEMENS S.A.
Polo II, Dep. Eng. Informatica       Rua Irmãos Siemens Nº 1
3030 Coimbra, Portugal              2720-093 Amadora, Portugal
Email: luis@dei.uc.pt                Email: leonor.almeida@lis2.siemens.pt

Abstract

Mobile agent technology is an emerging concept that can bring some important benefits for the telecom software providers. In some complex applications of telecommunications the client/server model is no longer effective and the use of mobile agents can be a good alternative.

In this paper we present ten good reasons for using this new technology in the development of software for telecommunication systems. Then, we present an overview of mobile agent platform, called JAMES. This platform is being implemented in Java and is mainly oriented for the management of telecommunications networks. This work has been done on behalf of a Eureka Project (Σ1921) with the following project partners: Siemens SA, University of Coimbra and Siemens AG. We describe the main architecture of the platform giving some emphasis to the key features for our target applications. One of the industrial partners of the project (i.e. Siemens S.A.) has developed a prototype application for TMN performance management. Although it is still a prototype it has been useful to validate the technological advantages of using mobile agents in software for telecommunications.
1. Introduction

The software for network management and telecommunication systems is facing some important challenges to deal with the complexity of the heterogeneous environments, the introduction of new technologies and the high-competition in the telecom world. Applications become more complex and the software providers are looking for solutions that allows a rapid deployment of applications into the market and allow a flexible customization according to the needs of the clients.

In order to be more competitive the telecom companies should adapt to the new emerging concepts and respond to the new challenges of the market. It is a fact that the computing models of the 80s are no longer efficient enough to solve all the new challenges. Thereby, new paradigms are necessary for the development of the distributed software.

For instance, the software for network management is currently implemented by using traditional Client/Server solutions. Typically, there are several network devices distributed across the network. In each device there is an installed static server that is responsible for collecting some information from the local device. These server processes are static and proprietary. They are very difficult to upgrade and/or customize. At a Manager host there is a client process that interacts with all the servers in the network devices, collects all the information from the network devices and provides the final information required to the end-user. This manager can represent a bottleneck to the applications and some of the applications are very inefficient in the use of the network bandwidth.

By all these reasons, this client/server approach presents some limiting drawbacks, namely:
- some of the applications impose a huge traffic in the network;
- the management agents are stationary and difficult to upgrade or modify;
- the manager site is usually a point of congestion;
- where there is a large amount of data the applications have problems of scalability;
- the applications are monolithic and difficult to use in heterogeneous environments;

Mobile agents is an emerging paradigm that is gaining momentum in several fields of applications, like mobile computing, electronic commerce, Internet applications, information retrieval, workflow and cooperative work, network management and telecommunications [1][2]. A Mobile Agent corresponds to a small program that is able to migrate to some remote machine, where it is able to execute some function or collect some relevant data and then migrate to other machines in order to accomplish a task [3]. The basic idea of this paradigm is to distribute the processing throughout the network: that is, send the code to the data instead of bringing the data to the code.

When compared with traditional Client/Server solutions mobile agents can potentially reduce the traffic in the network, provide higher scalability, easy customization, more flexibility and robustness. In the last few years the use of Mobile Agent technology has received an extraordinary attention from several Universities and research institutes and a notable investment from leading companies [4].

We have also realized the importance of agent technology and started a project to exploit the use of mobile agent technology. The project is called JAMES, and includes the implementation of a Java-based platform of mobile agents. This platform is mainly oriented for the development of applications in the field of telecommunication and network management. The project partners include Siemens S.A. (Portugal), University of Coimbra (Portugal) and Siemens AG (Germany). It was accepted by the European Eureka Program (Σ1921).

In this paper we will describe the main motivations and goals of our project. Section 2 describes the main advantages of using mobile agents in telecommunications and network management. Section 3 presents a brief overview of the JAMES Platform. Section 4 refers the implementation of a prototype application for performance management that was developed by using mobile agents, and finally, section 5 concludes the paper.

2. Ten Good Reasons for Using Mobile Agent Technology

Mobile agents are autonomous programs that can migrate through the machines of the network to accomplish some tasks on behalf of some user. The agents carry their code and their internal data while they are migrating between different machines. According to [5]
Mobile code can be seen as the merging of two concepts that have been successfully deployed in distributed computing: code-on-demand and remote evaluation. Java Applets are an example of code-on-demand while Java Servlets are based on the idea of remote evaluation. In both cases, the mobility of code is only between a client and a single server. Mobile agents go further than these concepts since the agents can migrate across several machines and provide a more decentralized approach.

The design and implementation of large-scale applications can benefit from the use of mobile agents. An extensive study on the advantages of mobile agent technology can be found in [6]. However, we will present next a list with the most important advantages for using mobile agents in the field of telecommunications.

1- Reduce the Network Traffic
Mobile agents fit well when the application has to deal with a huge amount of data or when it requires a high network bandwidth. The basic idea behind mobile code is simple but extraordinary: instead of pulling the data into the code just move the code near the data sources. This way, it would be possible to reduce the number of interactions and the amount of data that is sent through the network. This is an important advantage over traditional client/server solutions, as represented in Figure 1.

![Figure 1: Client/server versus Mobile Agents.](image)

Mobile agents are able to carry all the processing functions near to the data sources. This could result in substantial savings in the network bandwidth and reduce the number of interactions over the network. In this paradigm the agents are not restricted to push and pull operations. If some task requires the interaction with several machines in the network, the software agents are able to migrate across those machines and collect the data from several network devices.

2- Software Upgrading On-Demand
Usually, client/server solutions are very monolithic and difficult to upgrade. When there is a new to install or replace a particular piece of code this usually requires the upgrade of whole package of software. Moreover, in most cases this operation has to be done locally by a system manager.

Mobile agents are seen as small components of software that can be easily upgraded and replaced while the application is executing. The operation of upgrade can be done remotely through some manager site of the network, without requiring any physical movement of the system manager to the network devices. Besides, since mobile agents are small entities of software they simplify the replacement of small pieces of code without disturbing the rest of the application. This notion of “on-the-fly” upgrading is of paramount importance in telecommunication systems.

3- Easy Customization of Software
The previous functionality can also be exploited for flexible customization of the software. Some of the program modules can be implemented as mobile agents that are sent to the network devices. Then, according to the preferences of the end-user the agents can be easily customized and adapted to the configuration of the network.

4- Easy Introduction of New Services in the Network
Mobile agents can be used to extend the capabilities of applications and to introduce new services in the machines or devices of the network. With a dynamic directory service the client machines can locate new services in the network and interact with the corresponding servers. Moreover, if we integrate the use of Mobile agents with emerging technologies, like Jini from Sun Microsystems [7], it would become easier to provide the notion of plug-and-play devices and the introduction of new network services.

5- Higher Robustness for the Applications
The Client/Server approach is highly dependent of the availability of the network. If the client machine or the network goes down the application fails and has to be restarted from the very beginning. Mobile agents are inherently more robust to the occurrence of failures: if the client machine fails or the network is temporarily unavailable the agent is able to execute in the server machines and the application is able to tolerate these failures. Moreover, the agent platform can provide...
some more support for fault-tolerance to increase the reliability of the applications. With this support, the mobile agents may have the ability to sense their execution environment and react autonomously to changes and failures in the network.

6- Support for Disconnected Computing
With the use of mobile agents the client part of the application is moved to the server machines. The client machine can be disconnected from the network while the agent is performing its tasks. Later on, when the client machine plugs again in the network the agent will be able to report back the results of its computing task. By this reason, the use of mobile agents is seen as a very attractive solution for mobile computing [8].

7- Reduce the Software Development Cycle
Developing distributed applications with mobile agents is relatively easy for three main reasons: it follows the idea of component-based software; the applications are composed by several agents that can be deployed in a modular way; and the processing tasks can be distributed by several modules instead of relying in a complex central processing module. As soon as this technology matures, some RAD development tools will appear that will provide an environment of agent-oriented programming in the same line of the existing object-oriented programming tools.

8- Higher Scalability
Mobile agents allow the distribution of the “intelligence” throughout the network devices and decentralize the processing tasks among several machines. Compared with the client/server approach, where most of the processing is done in a centralized machine, the mobile agents paradigm provides a more scalable solution. This issue is very important in applications that have to execute in tens or hundreds of machines and where the traditional approaches present some execution bottlenecks.

9- Easy Integration of Vendor-Proprietary Systems
Nowadays, with the globalization of the market it is common to find in the same network an heterogeneous collection of vendor-proprietary equipment and software interfaces. Telecom software providers have always some serious difficulties to deal with the integration of different equipment. The implementation of integration protocols requires a large development cycle and not all the vendors provide standard interfaces. With the use of mobile agents everything that is required is a platform for agent mobility and a Java virtual machine. Mobile agents can then encapsulate some protocols and provide the necessary conditions for seamless system integration.

10- Higher Responsiveness in the Interactions with other Systems
In some systems the network devices have to interact in a real-time with other external systems. Sending mobile agents to these devices will prevent some uncertain delays caused by network congestion. This is an important issue when the application has to maintain a predictable level of quality of service (QOS).

For these reasons we decided to invest some R&D efforts in this mobile agent technology and see how this paradigm can be technologically exploited in the applications developed by Siemens. This is the main goal of the JAMES Project.

Several commercial implementations of mobile agents have been presented in the market, including Aglets from IBM [9], Concordia from Mitsubishi [10], Odyssey from General Magic [11], Voyager from ObjectSpace [12], Jumping Beans from AdAstra [13], Kafka from Fujitsu [14] and Grasshopper from IKV [15]. Although these software products have some interesting features they are too much general-purpose and do not provide any special support for network management.

In our project, we are developing from scratch a new Mobile Agent infrastructure that is being tuned and customized for the applications we have in mind in the area of telecommunications and data network management. In next section we present a brief overview of the JAMES Platform.

3. The JAMES Platform
The JAMES Platform provides the running environment for mobile agents. There is a distinction between the software environment that runs in the manager host and the software that executes in the Network Elements (NEs): the central host executes the JAMES Manager while the nodes in the network run a JAMES Agency. The agents are written by application programmers and will execute on top of that platform. The JAMES system will provide a programming interface that allows the full manipulation of Mobile Agents. Figure 2 shows
a global snapshot of the system, with a special description of a possible scenario where the mobile agents will be used.

Figure 2: An Overview of the JAMES Platform.

Every NE runs a Java Virtual Machine and executes a JAMES Agency that enables the execution of the mobile agents. The JAMES agents will migrate through these machines of the network to access some data, execute some tasks and to produce reports that will be sent back to the JAMES Manager. There is mechanism of authentication in the JAMES Agencies to control the execution of agents and to avoid the intrusion of non-official agents. The communication between the different machines is done through stream sockets. A special protocol was developed to transfer the agents across the machines in a robust way and is atomic to the occurrence of failures.

The application developer writes the applications that are based on a set of mobile agents. These applications are written in Java and should use the JAMES API for the control of mobility. After writing an application the programmer should create a JAR with all the classes that make part of the mobile agent. This JAR file is placed in a JAMES Code Server. This server can be a different machine or in the same machine where the JAMES Manager is executing. In both cases, it maintains a code directory with all the JAR files available and the mapping to the corresponding mobile agents.

The host machine that runs the JAMES manager is responsible by the whole management of the mobile agent system. It provides the interface to the end-user, together with a Graphical User for the remote control and monitoring of agents, places and applications. The JAMES GUI is the main tool for management and administration of the platform. With this interface, the user can manage all the Agents and Agencies in the system.

For lack of space we will not describe the inner details of the JAMES platform. However, in the following list we present the key features of our mobile agent system:
- Portability of the applications, through the use of the Java language;
- High-performance in mobility through the use of caching and prefetching techniques;
- Security mechanisms for code authentication;
- Resource control service to manage the use of underlying resources (CPU, memory, disk and operating system resources);
- System monitoring;
- Fault-Tolerance through the use of checkpointing and reconfiguration;
- Easy-to-use programming interface;
- Scalable execution of mobile agents, through the use of decentralized protocols;
- Easy customization of the software;
- “On-the-fly” software upgrading;
- Interoperation with classical network management protocols, like SNMP;
- Integration with Corba services;
- Support for Java-based technologies, like JavaSpaces and Jini [7];
- Distributed management and easy configuration of the network;

During this project, the platform will be used in two software products: one in the area of telecommunication and other for data network management. Siemens S.A. developed a prototype application in the area of performance management by using mobile agents. This prototype is finished and we are now conducting a benchmarking study to compare the use of mobile agents with traditional client/server solutions to see if we corroborate some of the advantages of this new paradigm in the field of distributed computing. A brief overview of this application prototype will be presented in the next section.

4. A Prototype Application for Performance Management

Current TMN architectures are based on highly centralized approaches for the location of service control and network management. The software for performance management in the TMN framework is currently dominated by systems based on client/server technologies. In most cases, this approach results in monolithic, not scalable and hardly flexible solutions.
The use of Mobile Agents for TMN data collection provides several potential benefits. First of all, mobile agents allow some processing at the data sources which increases the scalability and reduces the traffic in the network. The robustness of the application is also improved through the use of autonomous mobile agents and the application provider will have a more flexible way to deal with the diversity of network configurations. The application developer can create diverse collector agent versions, each one matching the proprietary raw data format involved, the software versions involved and the customer involved. Software upgrading is another important advantage that comes from the use of mobile agents. The upgrade of an application requires only the coding of new agents or the automatic deployment of new agent versions over the managed network. This way, the extension of the functionality or the installation of new software versions becomes more flexible and effortless.

In order to evaluate the use of mobile agents in real PM/TMN environments we have designed a prototype application to provide O&M Destinations Reporting, a component of TMN Performance Management.

A TMN application collects the performance data from the NEs and generates a set of reports about the performance of the network. In the current scenario, data collection and report building are two dissociated tasks: data is collected through file-transfer and is organized in a central relational database. This database is later queried to produce the performance reports. This two-step approach is based on a traditional client/server approach and presents some technical limitations: the data collection can introduce too much traffic in the network and the overall system can have some problems of scalability.

The benchmark application, designated as “Destination Reporting Application”, is designed to reproduce a representative subset of the TMN application reports. It makes use of mobile agents to collect and process the management data in the network. The application is structured into three different modules:
- a graphical interface that handles the end-user requests and outputs the reports;
- a mobile agents handler, responsible for the control of the reporting features;
- application specific mobile agents which are assigned to fulfil the required reports.

The benchmark application produces two different types of reports: on-demand and scheduled reports. On-demand reports correspond to requests to be immediately executed over the traffic destination raw data files stored in the remote NEs. Figure 3 presents a snapshot from an on-demand report that shows the evolution of two combined traffic destination counters.

Scheduled reports are a sort of report templates with a pre-defined behaviour. These report templates can be used at a later time with some attributes provided by the user, like the time window to be evaluated, the NEs to be considered, the type of traffic destination to be analyzed, etc.

Figure 3: Snapshot from the Prototype Application: a Time Behavior Report in Raw Table Format.

5. Conclusions
In order to be competitive the telecom service providers need new technologies that facilitate the rapid introduction of new services in a cost-effective manner. Mobile agent technology is an emerging paradigm that brings some important advantages over the use of traditional client/server solutions.

The market research firm Ovum predicts a $4 billion software agent market in the year 2000 with the widespread application of agent technology appearing in the computing, telecommunications, consumer, entertainment and military market segments [16].

Mobile agents are inherently distributed by nature and fit well in complex applications that have to deal with large amounts of data, have problems of scalability, have to deal with heterogeneous equipment, require the easy customization of the software or the introduction of new services.

In this paper we presented the main motivations for the use of mobile agent technology in network management and telecommunications. Then, we presented a
brief overview of a Java-based platform of mobile agents that is mainly oriented to this field of applications. We are currently conducting a benchmarking study with a prototype of an application for performance management. This study will be useful to take some conclusions about the practical use of mobile agents in TMN applications.

We plan to exploit the use of this technology in other functions of network management (fault, configuration, performance, accounting, security and traffic management) and in other fields of applications, like electronic commerce.

ACKNOWLEDGMENTS

This project is partially supported by Agência de Inovação and was accepted in the Eureka Program (Σ!1921). Special thanks to the rest of the team that is working in the project: Paulo Simões, Paulo Martins, Victor Batista, Guilherme Soares, Luis Santos, Fernando Bernardino and Rodrigo Reis (from University of Coimbra), Carlos Renato, João Boavida, Jochen Reban and Patricia Monteiro (from Siemens S.A).

REFERENCES


