

Mobile Musical Agents – The Andante Project

Leo Kazuhiro Ueda^{*}
lku@ime.usp.br

Fabio Kon[†]
kon@ime.usp.br

Department of Computer Science
University of São Paulo, Brazil

ABSTRACT

We investigate the use of mobile agents for the creation of music within a distributed computing environment. We believe this technology has the potential to foster new ways of making music. This poster presents Andante, an object-oriented, open-source infrastructure for building distributed musical applications based on mobile agents.

Categories and Subject Descriptors: J.5 [Arts and Humanities]: Performing arts

General Terms: Algorithms

Keywords: computer music, mobile agents, distributed music

1. INTRODUCTION

Composers have always looked at contemporary scientific achievements to devise new forms of producing their art. Traditional Western music itself went through changes as new forms of producing sound were being discovered and new instruments were being devised. Over the past decades, we have witnessed an astonishing development of Computer Science that has led to an intensification of the relationship between Music and Science [5]. In recent years, network technologies, especially the Internet, brought us many new possibilities for music making.

In this context, we are interested in discovering how an advanced concept in distributed objects research, namely *mobile agents*, can be applied to introduce new forms of musical composition, distribution, and performance.

The Andante project offers an open-source software infrastructure which allows the construction of distributed applications that use *mobile musical agents* to compose and perform music. Using Andante, programmers can write their own agents and build such applications. We are currently working with composers and researchers in writing musical pieces and it is our wish to attract the interest of more people both to use the system for conducting musical experiments and to help us to further enhance the infrastructure.

^{*}Supported by a graduate fellowship from CAPES, Brazil.

[†]Supported by a grant from FAPESP, Brazil, process number 02/02678-0.

In [6], we introduced the Andante project describing preliminary implementations of both the infrastructure and the first sample application. In [7], we gave more details of the infrastructure architecture and described the latest, more mature implementation, including changes and extensions incorporated in the past months and a new, more sophisticated sample application. There is still a lot that must be done in the infrastructure and we would greatly benefit from any help from the OOPSLA community in further enhancing the system.

2. MOBILE MUSICAL AGENTS

A *mobile agent* is a computer program that can interrupt its execution on a host, migrate to another host travelling through a network, and resume its execution on the new host [2]. It is an autonomous object in the sense that it can decide itself to migrate and it may react to changes on the host environment.

We define a *mobile musical agent* (or simply *agent* from now on) as a mobile agent which participates in a musical process. It may do so by performing one or more of the following activities: encapsulating a compositional algorithm; interacting and exchanging musical information with other agents; interacting with real musicians; reacting to sensors; and migrating.

We can build musical systems using such agents in a variety of ways. For space limitations we briefly describe here two of them:

Collaborative music. In systems such as DASE (<http://www.soundbyte.org>), users interact and exchange audio files through the network in order to compose a collaborative musical piece. A mobile musical agent system could use this same idea, except that users would implement and dispatch their own autonomous musical agents, which would interact with each other.

Distributed music. Consider a museum or exhibit hall equipped with several computers connected by a wireless network. Each computer could be equipped with motion sensors and host a few agents. The agents would communicate with each other and play a distributed piece of music in a synchronized manner. A specific agent could receive information from motion sensors in order to follow a person who walks around the room (using its ability to migrate). The listener perception would be that part of the music is following him.

3. ANDANTE

The Andante infrastructure offers software components for building applications similar to the ones described above.

3.1 Technologies

The whole system is written in Java for its platform independence, as we expect programmers (who mostly use Linux), composers (who mostly use Mac OS), and instrumentalists (who mostly use Windows) to use our system. Java also offers the Java Swing and the Java Sound libraries, which allowed us to prototype the infrastructure more easily.

Although we have so far only used Java, we would also like to allow parts of the system to interact with components written in other programming languages. The reason for this is to allow the use of other technologies for sound generation other than the ones provided by the Java Sound API. We are using CORBA to achieve this, so we can be able to integrate Andante with systems such as CSound [1], written in C, and Siren [4], written in Smalltalk.

Our infrastructure is built on top of the *Aglets Software Development Kit* (ASDK), a mobile agent platform based on Java [3] (<http://aglets.sourceforge.net>).

3.2 Architecture

An agent performs its actions in a heterogeneous computer network environment. The computers in this network must run a host software that we call *Stage*, representing a place where multiple agents meet and interact.

The Stage also offers the means for the agents to perform their actions. In particular, to produce sound, an agent needs to use the Stage's sound generation service. To provide this service, the Stage uses another component of the architecture: the *Audio Device*.

We have thus defined three key elements of the Andante architecture: the Agent, the Stage, and the Audio Device. Figure 1 depicts an abstract overview of the architecture.

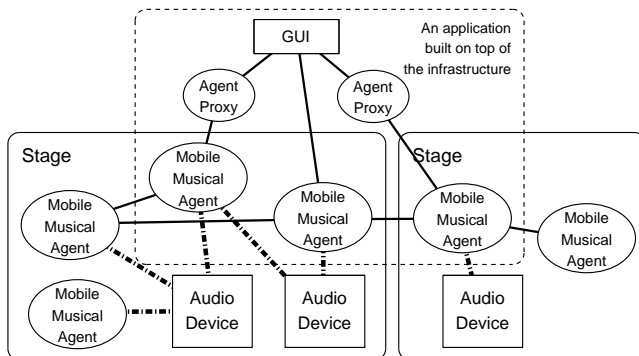


Figure 1: Architecture overview

An additional element, the GUI, is shown in the figure. It is not necessarily a component of the architecture, but it is the main component of applications built on top of the infrastructure and plays the important role of supporting human interaction with agents.

The application takes advantage of a fourth component of the architecture: the *Agent Proxy*. This element provides location transparency for agents. When an agent migrates, it informs its new location to its proxy, which in turn is responsible for the communication between the agent and the GUI. The GUI may also choose to communicate directly

with the agent, or to be the proxy for one or more agents itself.

4. APPLICATIONS

The following two applications were built using the implementation of the Andante infrastructure described in Section 3. They intend to give a concrete demonstration of the usefulness and feasibility of the architecture.

NoiseWeaver. The NoiseWeaver application generates and plays stochastic music in real-time. It implements one kind of agent: the *NoiseAgent*, which generates a single melody in real-time. In the generated melody, simulations of selected types of stochastic number generators determine the pitch, intensity, and duration of the notes. These generators work according to a large number of parameters that can be controlled by the user during the musical performance. We call them *noise* generators because they simulate the frequencies that occur in the spectrum of $\frac{1}{f^\beta}$ noises [5]. Using a GUI, a user can create and dispatch several NoiseAgents so that they interact with each other and generate stochastic music.

Maestro. The *Maestro* application allows a distributed collection of agents to be controlled by a script. The main element of the script is the *score*, where a user can determine time-stamped changes in the agents properties and location. The Maestro offers a GUI for editing and running scripts.

5. CONCLUSION AND FUTURE WORK

The Andante project expects to be more than a computer system. We are hoping to create an open community where artists, computer scientists, and software developers collaborate to create musical ideas, mobile musical agents, and to develop the enabling software infrastructure. To help with this effort, we keep a Web site at <http://gsd.ime.usp.br/andante>.

With the initial prototype of the infrastructure implemented, we are now moving the main focus of the project in the direction of musical creation. Our next steps will be guided by the interaction with composers. We will continue the development of the infrastructure and the applications as we plan to add new functionalities and refinements. Besides that, we plan to design and to build new applications to better explore agent mobility and human-agent and agent-agent interactions.

6. REFERENCES

- [1] R. Boulanger, editor. *The CSound Book*. The MIT Press, 2000.
- [2] D. Kotz and R. S. Gray. Mobile Agents and the Future of the Internet. *ACM Operating Systems Review*, 33(3):7–13, July 1999.
- [3] D. B. Lange and M. Oshima. *Programming and Deploying Java Mobile Agents with Aglets*. Addison-Wesley, August 1998.
- [4] S. T. Pope and C. Ramakrishnan. Recent developments in Siren: Modeling, control, and interaction for large-scale distributed music software. In *Proceedings of the 2003 International Computer Music Conference*, Singapore, 2003.
- [5] C. Roads. *The Computer Music Tutorial*. The MIT Press, 1996.
- [6] L. K. Ueda and F. Kon. Andante: A mobile musical agents infrastructure. In *Proceedings of the 9th Brazilian Symposium on Computer Music*, Campinas, Brazil, August 2003.
- [7] L. K. Ueda and F. Kon. Andante: Composition and performance with mobile musical agents. In *Proceedings of the International Computer Music Conference 2004*, Miami, November 2004.