



HAL
open science

PAMS - A New Collaborative Framework for Agent-Based Simulation of Complex Systems

Khanh Nguyen Trong, Nicolas Marilleau, Tuong Vinh Ho

► **To cite this version:**

Khanh Nguyen Trong, Nicolas Marilleau, Tuong Vinh Ho. PAMS - A New Collaborative Framework for Agent-Based Simulation of Complex Systems. Lecture Notes in Computer Science, 2008, 5357/2008, pp.287-294. 10.1007/978-3-540-89674-6_32 . hal-00741724

HAL Id: hal-00741724

<https://hal.science/hal-00741724>

Submitted on 15 Oct 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PAMS – A New Collaborative Framework for Agent-Based Simulation of Complex Systems

Trong Khanh Nguyen², Nicolas Marilleau¹, and Tuong Vinh Ho²

¹ Geodes, Institut de Recherche pour le développement (IRD),
32 av. H. Varagnat, 93143 Bondy Cedex, France

² MSI Lab, Institut de la Francophonie pour l'Informatique (IFI),
ngo 42, Ta Quang Buu, Ha Noi, Viet Nam
ntkhanh@ifi.edu.vn, nicolas.marilleau@ird.fr,
ho.tuong.vinh@auf.org

Abstract. Major researches in the domain of complex systems are interdisciplinary, collaborative and geographically distributed. The purpose of our work is to explore a new collaborative approach that facilitates scientist's interactions during the modelling and simulating process. The originality of the presented approach is to consider models and simulators as a board of the collaboration: a shared object manipulated by a group of scientists. Agent-based simulations are powerful tools for studying complex systems. In this context, we develop a collaborative platform dedicated to agent-based simulation (PAMS). This new environment integrates common collaborative tools (e.g. videoconferencing, instant messaging, whiteboard) and specific tools to share and manipulate models, simulators, experiments and results... The current version of PAMS is based technologies coming from distributed systems. Today PAMS has been designed to support major agent based simulation frameworks. This paper aims to give an overview of the PAMS environment by defining the collaborating approach, the framework architecture and an example of its utilization.

Keywords: Collaborative simulation, agent-based simulation, distributed systems.

1 Introduction

Modeling and simulation often requires cooperation between researchers from different disciplines. Data collection, model conceptualization and implementing those models using computational tools all require close teamwork amongst various players (domain experts, modelers and computer scientists). Modern-day research projects are interdisciplinary, collaborative, and researchers are often geographically separated. Given these modern conditions, the use of collaborative systems becomes essential to facilitate complex systems research when a team of geographically and professionally dispersed researchers must work together towards a common goal.

The advent of new information technologies and communications tools over the past fifteen years has enabled the development of a plethora of collaborative platforms [2]. A few of them, such as BSCW [11], E-Groupware[12], and Sakai [1] have

positioned themselves as collaboration-oriented extensions of generalist communications platforms in the scientific world. These products integrate specific functionalities facilitating: (i) access to knowledge and scientific information, (ii) interaction and collaboration between researchers, and (iii) a more effective dissemination of research results. These activities are often referred as “E-Research”. The platforms supporting such work are referred to as Virtual Research Environments (VRE) [13].

VRE systems are still relatively undeveloped. Most of the available environments remain incapable of supporting significant collaborative efforts; despite growing demands for such tools the scientific community [6]. Most of these collaboration systems are simply data repositories with web interfaces [5].

In the domain of modeling and simulation of complex systems, the use of agent-based simulation models (ABM) is increasingly popular. Many ABM platforms (Repast [8], NetLogo [9], Swarm [7] or GAMA [3]) have been developed and used by researchers [3, 10]. In the context of collaborative research, one major question is: How can geographically diverse researchers effectively work together to conduct ABM simulations without regard to the ABM platforms used?

One of our research interests focuses on the design and implementation of collaborative environments for computer modeling and the simulation of complex systems, essentially those based in ABM. The main idea is to place models, simulators, experiments and results at the center of the collaboration. From this idea, we have designed and developed methodologies and a form of “groupware” (known as “PAMS”) for supporting collaboration between domain experts, modelers and computer scientists. PAMS is a type of web-based groupware containing common collaborative tools (video-conferencing, instant messaging, and so on) and specific tools dedicated to the simulation domain (sharing experiments, results, experience exchange...).

The aim of this paper is to present the PAMS framework. Firstly, we shall summarize the platform’s functionalities. Secondly, we provide a short description of the PAMS architecture. Finally, a case study is described demonstrating how a collaborative simulation might be executed within this new paradigm.

2 PAMS – A New Collaborative Framework for Agent-Based Simulation of Complex Systems

The PAMS project introduces a new approach to collaboration in research projects. Models and simulators will no longer simply come in the form of “research results” distributed to the community through scientific communications (journal articles, workshops etc.). Rather, with PAMS, models and simulators become concrete entities available to interested parties on the web to support collaborative work and research. The originality of our approach is to consider the model or the simulator as an object shared by a group of researchers, which can be manipulated, configured, analysed and so on.

PAMS groupware is an environment allowing researchers to work together in designing or exploring models (execution of models based on various scenarios, interactions conceptualized in various contexts or of a specific simulation). For example, consider a scenario for collaboration between researchers in which they used the software environment in development.

Scenario1: “Collaborative experimentation from distance”

Mr. X and Y, researchers in ecology, respectively located in Paris and Hanoi, wish to execute some simulations in order to get results that will be illustrated in an article they will write together. The two researchers connect to the project website and begin a private discussion. Mr. X starts the simulator, and suggests that Mr. Y. shares his interface. Mr. X starts entering simulation parameters, which are not suitable to Mr Y. Mr Y starts a videoconferencing session to share with Mr. X his surprise. He posts a note on the simulator interface indicating the value of parameters that seem correct to him. After discussion, and display, by Mr. Y, part of the item they want to illustrate, the two fall into agreement on common values. Mr. X launches the simulation and then chooses to display only the graph of the evolution of biodiversity, while Mr. Y visualizes the spatial distribution of species. Each researcher posts annotations of visualization in real time on his own interface, which also appear on the interface of his colleague and they begin holding a discussion (registered as such in the instant messaging system). Due to time constraints, however, Mr. X must leave. They both decide to resume this discussion later and record the session. Three days later, returning to the site, the conversation restarts and resumes in the state where it had been left earlier...

This scenario shows that the use of a collaborative tool firstly: addresses the problem posed by the geographical dispersion of researchers and secondly, brings a fresh dimension to the simulation activities of complex systems.

An agent-based simulator is run many, many times using different parameter values in order to postulate various situations and to understand the dynamics of the studied system. In this context, users focus on inputs and outputs of a model. They are free to forget how the model works. For this reason, we assume that a perfect agent-based simulator is a “black box” that scientists (except for the box’s creators) use by defining inputs and analyzing outputs without really caring how the “black box” works.

Existing Open Source web-based groupware constitutes a solid basis upon which it is possible to add new modules providing collaboration in the field of complex systems. We chose one of them called “Sakai”[1] and added new collaborative modules dedicated to simulation activities, such as: (i) setting simulations; (ii) executing simulations on a remote server; (iii) visualising and analysing results; (iv) managing versions of available models; (v) archiving experiments and results; (vi) annotating experiments and results (giving contextual comment).

3 PAMS: A Modular Environment

3.1 Logical Architecture

PAMS is based on a multi-tier architecture called Model View Controller (MVC) [15]. This approach distinguishes graphical user interface (Presentation Tier) from the kernel of the application (Logic Tier) and databases (Data Tier).

Fives modules compose the kernel of the PAMS environment (See figure 1):

- Simulation platform drivers package** contains the kernel of agent-based platforms such as Repast [8] or Gama [3].

- Outputs package** manages simulation results coming from simulation platforms, and shares these data for other modules of the systems.
- Displays package** formats shared outputs in order to generate and manage user displays: monitors, plots and/or 2D grids (images).
- Recorder package** saves every value that an output has taken during a simulation in a database. This data is read by the experiment’s web browser interface.
- Controllers package** aims at managing experiments and simulators, ensuring the coherency and concurrency of objects shared by users (parameters, simulation outputs, experiments...).

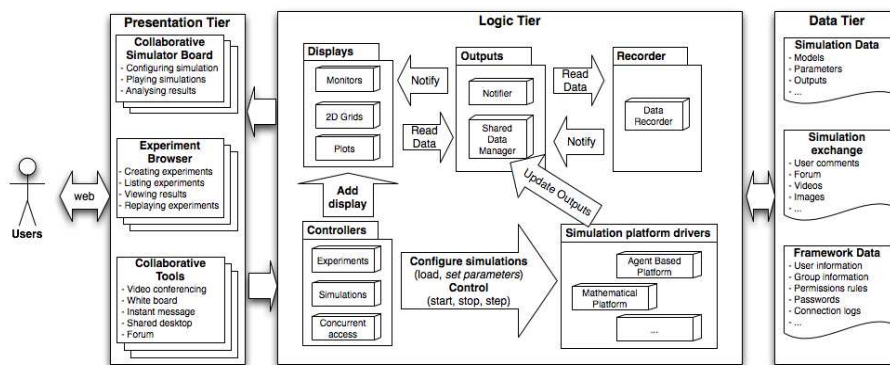


Fig. 1. PAMS logical architecture

The PAMS environment proposes a generic web-based collaborative GUI. This interface takes advantage of typical collaborative tools (video-conferencing, white-board and so on) coming from Sakai and Agora tools. In addition, PAMS provides functions dedicated to the simulation domain:

- A collaborative simulation board** for executing and sharing remote simulators
- An experiment browser** for managing and replaying completed experiments and exchanging results.

Today, PAMS supports simulators derived from two agent-based platforms: Repast and Gama. A few famous and simple simulators were deployed, e.g. the life game (Enn for Repast, Life for Gama) to test PAMS functionalities. In addition, specific simulators, as GamaAvi, with its origins in a multi-disciplinary research project, are and will be added. Scientists (epidemiologists, geographers, computer scientists or mathematicians) will use these simulators to run experiments and to collaborate.

3.2 Technological Architecture

The PAMS framework is a distributed system. It can be viewed as a container in which simulators are loaded, connected with a database, executed by dedicated servers and managed through a web interface.

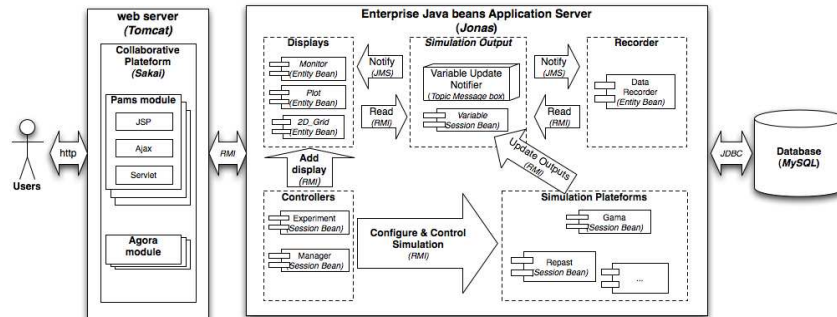


Fig. 2. Technological architecture

PAMS is based on common technologies from the domain of distributed systems (see figure 2):

- A **web application server** based on Jsp, Ajax and Servlet (*Tomcat*) to manage the collaborative and adaptable web interface that displays models, experiments, simulators and results with a simple, adaptable and didactic structure of data.
- An **Enterprise Java bean application server** (*Jonas*) to execute simulators, manage experiments, inputs, outputs and collaboration.
- A **Database** (*MySQL*) to store models, experiments inputs, outputs and exchanges between PAMS users.

The PAMS framework takes advantage of the Sakai environment [1]. Sakai is an online Collaboration and Learning Environment that permits research exchange. From the Sakai environment, we developed new modules that manage a collaborative GUI specific to the simulation domain (experiment viewer, simulation board etc). The PAMS module is associated with the Agora environment [14]. Agora is a plug-in of Sakai that offers typical collaborative features like videoconferencing, whiteboard, chat and others.

The use of Enterprise Java Beans is one of the keys that allows PAMS environment to be flexible, modular and modifiable. Each module of the framework is composed of several EJBs. Every EJB of a same module is used through a unique interface (determined for the module). To improve the PAMS framework, new EJB could be developed and dynamically deployed without revising old PAMS components. But, these new EJBs must follow predefined interfaces of the PAMS.

The PAMS environment can be deployed on a GRID of computers. Thanks to this distributed architecture, load-balancing rules can be imagined to spread experiment executions over a GRID.

Most of agent-based simulators depend on a specific platform such as Netlogo, Repast or Madkit. These agent-based frameworks must be installed on the computer before the setup of the simulator. Sometimes, simulators need a database to obtain working data. Simulator setups are not trivial: much experience in computer science is required. In addition, many simulators need resources (memory or processor) that are not available on a personal desktop or laptop. Thanks to the web interface of PAMS

and its distributed architecture, scientists have nothing to install before using the framework. In addition they can take advantage of resources provided by the Grid in which the platform is executed.

4 Case Study

The aim of this case study is to show how a group of scientists can use the PAMS environment to run simulations and collaborate. Scenario 1 (“Collaborative experimentation from a distance”) can be taken as an example. In this instance, Mr. X and Mr. Y, want to study the famous Life game model [4].

Consider that a PAMS service is running on a server. This service supports various agent based simulation frameworks, in particular the Repast environment. Several simulators have been installed, deployed and are available. For example, the environment proposes the Enn simulator, which is a Repast version of the Life game model.

Using a web browser, Mr. X and Mr. Y connect themselves to the PAMS platform. After the identification step, they access to their private workspace. Mr. X is the initiator of the experiment. He has to create an experiment and to determine the participants. From a list available in the agent-based simulator, Mr. X selects the Enn simulator. A new display appears that shows information about the Enn simulator: aims of the model, inputs, outputs and so one. On this screen, Mr. X can see every public experiment done with the Enn simulator. But, he prefers to create a new one. For that, X inputs a comment about the new experiment (its aim) and selects participants from a list of subscribed persons. In the case of this scenario, Mr. X selects Mr. Y. and submits the form. The experiment is now created. X is waiting Y’s connection to start simulations.

Y selects the Enn simulator from the list that contains available agent-based simulators. Information about this simulator is displayed, and Y sees that X has invited him to participate in an experiment. Y selects connects to the experiment.

X and Y are seeing the same display: the simulation board of the Enn simulator. For that, he takes the token. Mr. Y’s display is freezing. Y cannot perform the action on the simulator board, but he sees modifications. After doing modifications, X releases the token.

Y does not agree with X’s parameter modification. To explain his surprise, Y starts a videoconferencing session integrated in the Agora meeting tool (see figure 2). Thanks to video, audio and whiteboard tools, Y discusses with X. X and Y exchange their opinions through a user-friendly GUI. To illustrate its says, Y shows an article by sharing its desktop and convinces X. X wants to modify simulation parameters according to Y’s recommendation. He takes the token and changes parameter values. Then, X starts the simulation.

X and Y see the evolution of predetermined outputs in real time. During the simulation, they discuss the evolution of the outputs. From these results, X and Y begin an analysis and make hypothesis about the phenomena they see.

Due to time constraints Mr. X must leave. They both decide to resume this discussion later. So they record the session. Three days later, returning to the site, the conversation starts and resumes in the state where it had been left earlier...

Figure 3 shows the screen shots (of the PAMS environment) during the execution of the above scenario.

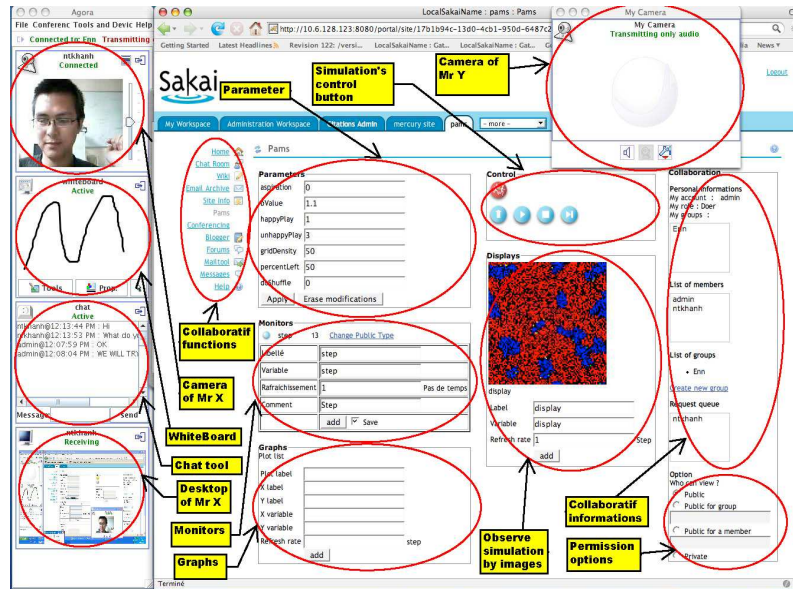


Fig. 3. Screen shot of the PAMS environment (Desktop of Mr. Y)

5 Conclusion

PAMS is a collaborative framework for simulating Agent-based models of complex systems. It is based on an original approach in which the simulator supports the exchange: it is a shared object manipulated by every member of a scientific group. In addition, we consider that users view a simulator as a “black box”. So users concentrate and collaborate on the input and output of the simulators.

Scientists are able to collaborate in this manner thanks to a web-based GUI that allows remote, shared access to simulators. This GUI contains common tools (video-conferencing, instant-messaging, whiteboard, and so one), which are improved by specific tools dedicated to the domain of the simulation of complex systems. This GUI supplies collaborative functions to setup simulators, to execute simulators on a remote server, to visualize and analyze simulation results, and to keep logs of each experiment.

Currently, the PAMS environment supports two agent-based simulation platforms (Repast and Gama). Adding new drivers will support of every simulation platform in the future. The modularity of PAMS permits our environment to be improved in many ways. Adding new kinds of displays or collaborative tools is one example.

The PAMS environment will be improved in many ways: (i) adding new collaborative tools (e.g. an annotating system to comment experiments), (ii) optimizing the system (e.g. adding load-balancing strategies), and (iii) supporting new agent based simulation platforms (e.g. Madkit, Repast and so on). Before that happens, we will have to test the existing version on a concrete research project applied to, for instance, geography or epidemiology. Feedback from these tests will provide the vital keys we require to further development and future improvements of the PAMS environment.

References

1. Severance, C., Hardin, J., Golden, G., Crouchley, R., Fish, A., Finholt, T., Kirschner, B., Eng, J., Allan, R.: Using the Sakai collaborative toolkit in e-Research applications. *Concurrency and Computation: Practice and Experience* 19(12), 1643–1652 (2007)
2. Saint-Voirin, D.: Contribution à la modélisation et à l'analyse des systèmes coopératif: application à la e-maintenance. Université de Franche-Comté, Besançon (2006)
3. Amouroux, E., Quang, C.T., Boucher, A., Drogoul, A.: GAMA: an environment for implementing and running spatially explicit multi-agent simulations. In: *Prima-2007*, Bangkok (2007)
4. Conway, J.: The Game of Life. *Scientific American* 223, 120–123 (1970)
5. Henriksen, J.O., Lorenz, P., Hanisch, A., Osterburg, S., Schriber, T.J.: Web based simulation center: professional support for simulation projects. *Winter Simulation Conference-2002* 1, 807–815 (2002)
6. Ahmed, K., Brahim, B.: Towards a Web Based Simulation Groupware: Experiment with BSCW. *Information Technology Journal* 1812(5638), 332–337 (2008)
7. Terna, P.: Simulation Tools for Social Scientists: Building Agent Based Models with SWARM. *Journal of Artificial Societies and Social Simulation* 1(2) (1998)
8. North, M.J., Collier, N.T., Vos, J.R.: Experiences Creating Three Implementations of the Repast Agent Modeling Toolkit. *ACM Transactions on Modeling and Computer Simulation* 16(1), 1–25 (2006)
9. Wilensky, U., Evanston, I.L.: NetLogo. Center for Connected Learning and Computer Based Modeling, Northwestern University (1999)
10. Railsback, S.F.: Agent-based based Models in Ecology: Patterns and Alternative Theories of Adaptive Behaviour. In: *Agent-Based Computational Modelling*, pp. 139–152. Physica-Verlag (2006)
11. Horstmann, T., Bentley, R.: Distributed authoring on the Web with the BSCW shared workspace system. *StandardView* 5(1), 9–16 (1997)
12. Becker, R., Becker, B., Knotte, M., KreiBmeyer, I.: *Manual eGroupware 1.4*. Creative Commons (2007)
13. Yang, X., Allan, R.: Web-Based Virtual Research Environments (VRE): Support Collaboration in e-Science. In: *WI-IATW 2006: Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, pp. 184–187. IEEE Computer Society, Los Alamitos (2006)
14. Severance, C., Hardin, J., Golden, G., Crouchley, R., Fish, A., Finholt, T., Kirschner, B., Eng, J., Allan, R.: Using the Sakai collaborative toolkit in e-Research applications. *Concurrency and Computation: Practice and Experience* 19(12), 1643–1652 (2007)
15. Reenskaug, T.: The Model-View-Controller (MVC) Its Past and Present. *JavaZONE Conference*, Oslo (2003)