# GAMA: A Spatially Explicit, Multi-level, Agent-Based Modeling and Simulation Platform

Alexis Drogoul[1], Edouard Amouroux, Philippe Caillou, Benoit Gaudou, Arnaud Grignard, Nicolas Marilleau, Patrick Taillandier, Maroussia Vavasseur, Duc-An Vo, and Jean-Daniel Zucker

IRD, UMI UMMISCO 209, 32 av. H. Varagnat, 93143 Bondy, France
alexis.drogoul@gmail.com

**Abstract.** Agent-based modeling is now widely used to investigate complex systems but still lacks integrated and generic tools to support the representation of features usually associated with real complex systems, namely rich, dynamic and realistic environments or multiple levels of agency. The GAMA platform has been developed to address such issues and allow modelers, thanks to the use of a high-level modeling language, to build, couple and reuse complex models combining various agent architectures, environment representations and levels of abstraction.

**Keywords:** Simulation platform, Agent-based modeling, GIS data, Multi-level model.

## 1 Introduction

Agent-based modeling (ABM) has brought a new way to study complex systems by allowing to represent multiple heterogeneous entities interacting in a non-linear fashion in a shared environment. Although it is now used in different domains, ABM still struggles with two issues. First the lack of a comprehensive and common representation of the environment(s) in which agents interact, which limits its usefulness for models where the environment itself is to be represented as a complex entity. Secondly a difficulty to go beyond the classical Object-Oriented Paradigm to express interactions between agents at different levels of abstractions (e.g. agents composed of agents). Even though tools have been proposed in the recent years, they are too complex for domain experts to build their models without a strong support from computer scientists. For instance, building a realistic model that relies on GIS data at different geographical scales still involves complicated coding tasks in most ABM environments.

The GAMA (GIS & Agent-based Modeling Architecture) modeling and simulation platform has been proposed to address such shortcomings. On one hand, this open source platform allows the definition of agent-based models with complex environment representations and generic multi-level capabilities. On the other hand, it provides field experts, modelers, and computer scientists with a complete modeling and simulation environment for building spatially explicit agent-based models with ready-to-use abstractions for the most common needs

(e.g. decision architectures, generic behaviors, such as movements regardless c
the environment representation), which are accessible through a dedicated higl
level modeling language (GAML) and remains extensible by Java programmer

## 2 Main Purpose

GAMA is based on: (i) a meta-model [5] dedicated to complex environment rep
resentation and multi-level models; (ii) a modeling language (GAML) and i
related elements (parser and compiler); (iii) an efficient virtual machine to ex
cute model and to interact with the simulation. Compared to other framewor
such as NetLogo [6] or Repast Symphony [2], its main advantage is to provic
this multi-level architecture (extendable via plug-in), and a very complex env
ronment representation easily defined via GAML.

**Multi-level Modeling.** Multi-level agent-based modeling requires to manipu
late agents at different levels of representation w.r.t. to time, space and behavio
Our approach of multi-level modeling is based on three principles. (1) An agen
represents a level of organization which is associated with a spatial and temp
ral scale. (2) Levels are hierarchically organised to define privileged interactio
between embedded levels. (3) Organizations can be dynamic: some agents ca
move from an organization to another in order to adapt their representation lev
dynamically.

The GAMA meta-model [5] has been designed to fulfill these three feature
An agent is an instance of a species (the kind of agent, as in classes and instanc
in OOP). Every agent possesses a spatial representation (its shape) that is l
cated in an environment. An agent, as organization level, can host populatio
of micro-agents. Its topology defines the spatial environment of hosted agent
Similarly, the macro-agent defines the execution time scale of hosted agents b
specifying the way they will be scheduled. This thus defines a hierarchical stru
ture of the model: any agent is hosted by another agent and possibly hosts po
ulations of other agents. The macro-agent will manage the relations between i
behavior and micro-agent behavior: it is indeed allowed to redefine the behavi
of hosted agents. A top-level agent (the world) has been introduced. It manag
the global variables and parameters of the simulation and its shape defines th
reference environment. Finally, agents can dynamically migrate from a popu
lation to another one (by changing species); this allows agents to change the
organization level, and thus representation level during the simulation course.

**Environment Representation.** GAMA is particularly powerful concernin
the management of complex environments. It allows to define several enviro
ments with different topologies (grid, graph or continuous). One continuous e
vironment is used as reference to synchronize all of them. Each GAMA agent h
a shape, that is a 3D simple (point, polyline or polygon) or complex (compose
of several geometries) geometry.

A particularly interesting feature of GAMA is the possibility to create agen
and to define their attributes (in particular their shape) from real data usin

shapefiles. Conversely, this allows the modeler to integrate geographical data into models under the form of active agents (one agents is created by geometry of a shapefile). In addition, GAMA manages the spatial projection of the data (to get a spatially coherent model) and the reading of attribute values. In order to ease geometries use and manipulation, high-level geometry transformations (e.g. buffer, convex-hull, etc.) and movement primitives (e.g. shortest path computation) are readily available in GAML.

## 3 Demonstration

GAMA can be used for lots of purposes including teaching, conceptual modeling and applied research. We illustrate its power with applied and abstract uses[1].

**Applied Models.** GAMA has already been used in various large scale applications that share a strong focus on the interactions between agents and complex environments. Epidemiological models have been developed to study avian flu persistence in North Vietnam and rift valley fever propagation in Senegal. It has been used to assess the effectiveness of control policies on the recurrent invasions of insects in the Mekong delta, to simulate rescue management in Hanoi after an earthquake or evacuation organisation in case of a tsunami in Nha Trang (Vietnam). The MAELIA project [3] uses it to study socio-ecological impacts of water management policy in the Adour-Garonne Basin (France).

GAMA can manage a large number of agents for real-scale applications, for example, nearby 200 000 for the MIRO model (Figure 1) [1]. This model addresses the issue of sustainable cities by focusing on one of its very central components, daily mobility. Therefore, improving urban accessibility merely results in increasing the traffic and its negative externalities (congestion, accidents, pollution, noise...), while reducing at the end the accessibility of people to the city. For that, an ABM has been developed and applied to Dijon and Grenoble, two mid-sized cities (nearby 120 000 inhabitant) in France. The simulator is used to realise scenarios determined by geographers for quantifying, for example, service accessibility and to organise serious game sessions for identifying cities management strategies.

**Abstract Models.** As presented in previous section, GAMA offers several modeling capabilities, like multi-level simulation, seamless integration of GIS Data or extensible architecture with plugin. Very simple conceptual models can be developed and used for demonstration or conceptual proof. For example, the multi-level architecture can be illustrated by a flocking example with Flock agent dynamically created when nearby boids converge (an illustration can be watched in the video). The Flock agent captures boids agents and computes its own geometry from their individual data. Boids will be released when the Flock will disappear (when it moves toward an obstacle).

Link to a video:
`http://code.google.com/p/gama-platform/wiki/VideosPresentation`

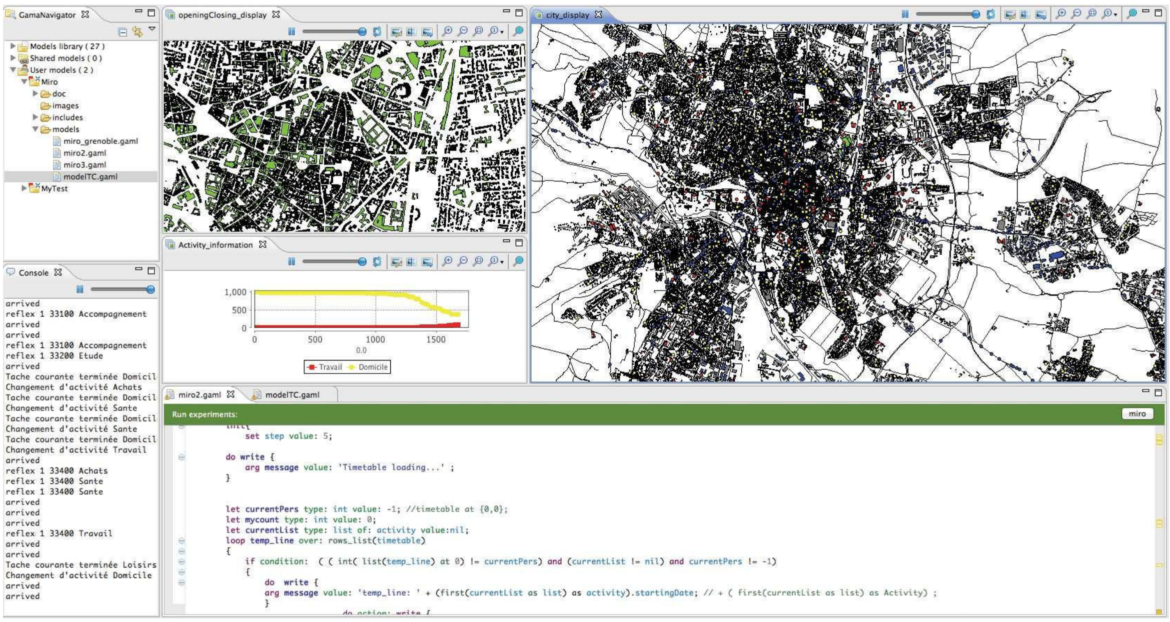


**Fig. 1.** MIRO simulation interface

## 4 Conclusion

The latest version of GAMA, bundled with a set of example models, as well as its source code, can be downloaded from its website[2]. The site also provides users with a complete modeler guide, several tutorials, the GAML language reference, as well as a guide for developers to extend the platform with their own plugins.

## References

1. Banos, A., Marilleau, N., MIRO team: Improving individual accessibility to the city: an agent-based modelling approach. In: Proc. of ECCS, Bruxelles (2012)
2. North, M., Howe, T., Collier, N., Vos, J.: A declarative model assembly infrastructure for verification and validation. In: Proc. of Advancing Social Simulation: 1st World Congress (2007)
3. Taillandier, P., Therond, O., Gaudou, B.: A new BDI agent architecture based on the belief theory. Application to the modelling of cropping plan decision-making. In: Proc. of IEMs, Germany, pp. 107–116 (2012)
4. Taillandier, P., Vo, D.-A., Amouroux, E., Drogoul, A.: GAMA: A simulation platform that integrates geographical information data, agent-based modeling and multi-scale control. In: Desai, N., Liu, A., Winikoff, M. (eds.) PRIMA 2010. LNCS, vol. 7057, pp. 242–258. Springer, Heidelberg (2012)
5. Vo, D.-A.: An operational architecture to handle multiple levels of representation in agent-based models. PhD thesis. Université P&M Curie, Paris (2012)
6. Wilensky, U.: Netlogo. Technical report, Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL (1999)

[2] `http://gama-platform.googlecode.com`