

# Towards a Domain Specific Modeling Language for Agent-Based modeling of Land Use/Cover Change

Cédric Grueau

Email: cedric.grueau@estsetubal.ips.pt

<sup>1</sup> Departamento de Sistemas e Informática

Escola Superior de Tecnologia de Setúbal - Instituto Politécnico de Setúbal

Campus do IPS, Estefanilha, 2910-761 Setúbal, Portugal

<sup>2</sup> Centro de Informática e Tecnologias da Informação - CITI (DI/FCT/UNL)

Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,

Quinta da Torre, 2829-516 Caparica, Portugal

**Abstract.** While agent-based models are widely used for the simulation of human-natural systems, it remains challenging for scientists to specify their models in a manner in which they can be understood and used by others. In this research project, we survey existing solutions that have emerged to cope with models specification issues. We then propose another approach: a domain specific modeling language for agent-based simulations of land use/cover change. This language is intended to constitute a means to promote models' validation and reuse. We also present the general methodology for the development of the modeling language which consists in extending existing conceptual modeling languages with domain-specific language profiles. We also present the first step towards its development comprising a domain model based on a domain ontology.

**Keywords:** Domain ontology, Multi-Agents Systems, Domain Specific Modeling Language, Land-Use Science

## 1 Introduction

Computer-based simulations have become an important asset to understand the complex interactions between human societies and the land resources on which they depend. Many of these simulations rely on agent-based modeling (ABM) to represent social interactions. A spatial context (static or dynamic) is provided to represent the context where social interactions occur. In the particular domain of agent-based modeling of land use/cover change (ABM/LUCC), simulations are used to understand and test possible effects of alternative policy or management interventions in the socio-spatial system under study. Through simulation, modelers can understand and reproduce, with virtual experiments, the emergence of nearly any kind of macro-structure or macro-dynamics from the interactions of lower level computer processes, called agents. Building such simulations is

challenging but ABM have been enthusiastically adopted by ABM/LUCC scientists during the last decade. [19], [14] and [1] provide a broad range of examples of ABM/LUCC applications as diverse as innovation diffusion, city size distributions study, demographic and lot effects of agriculture in the Brazilian Amazon or forest management in Asia. However the fast development of this type of applications quickly raised concerns related to transparency issues [20]. Transparency pertains to how easy it is for an external person to understand a simulation model. ABM transparency issues are directly related to the manner in which models are described and specified by their authors. In practice, the representation of the model (and submodels) of the system in scientific papers will differ according to the background of the system’s designer. These descriptions integrate pieces of natural language, mathematical formulas, pseudo code, UML diagrams and sometimes computer code. The diversity and lack of consistency between representations make it difficult to reuse and validate models. If design documents are unavailable, it is necessary to reverse-engineer the program back to the design to link with the high-level description of the model given in any source text [20]. Moreover, when, for example, reusing a model published in a scientific journal paper, one has no guaranty that the model was correctly implemented and that the results of the simulations conform to the model described. Such concerns represent a real challenge for the users and decision makers that rely on the simulations results.

## 2 Research questions and objectives

In order to respond to ABM/LUCC transparency issues, we propose to follow an approach that will address three questions. The first question is how to represent ABM/LUCC information structure in a manner in which it is decoupled from the simulation software and can be independently processed. The second question to address relates to the level of abstraction to achieve in order to represent the concepts of the domain. In fact, the abstraction should enable the representation of concepts and relationships for a multidisciplinary audience who is not necessarily expert in computer science, but, at the same time be precise enough to represent all concepts of the domain. A third question to answer is how can we ensure that an executable implementation conforms to a system’s model?

To respond to these questions we propose to adopt a research approach consisting in the development of a Domain Specific Modeling Language (DSML). The objective of the DSML is to define an abstract representation of the ABM/LUCC domain, a domain model, and use it to instantiate executable systems (e.g. source code for simulation). By following such an approach, a particular system can be represented using the concepts defined in the domain model and its properties can be clearly specified using the modeling language. The objective of providing this DSML is moving towards a unified design that would facilitate the interpretation of the system by developers or practitioners that would want to reuse the system. It should answer the first two questions raised above. Our goal is to provide a modeling language that would meet users needs and that would

be straightforward enough to be adopted by the main users of the domain. By following a model-driven approach we also pretend to diminish the gap between system representation and system implementation. Providing the possibility of deriving executable code from the modeling language, using model transformations, would allow us to respond to the third question of our research.

### 3 State of the art

#### 3.1 Model-Driven Engineering and ontology-driven conceptual modeling

In MDE, models are described by modeling languages, where modeling languages themselves are described by so called metamodeling languages. A modeling language consists of an abstract syntax, at least one concrete syntax and semantics. Domain-Specific Modeling Languages (DSML) are specification languages that offer, through appropriate notations and abstractions, expressive power focused on, and usually restricted to, particular problem domains [4]. A DSML can be used to generate members of a family of systems in an application domain. The well-designed DSML is based on a thorough understanding of the underlying application domain, giving exactly the expressive power to generate required family members easily. Thus, a prerequisite for the design of a DSML is a detailed analysis and structuring of the application domain. Guidelines for acquiring such an understanding are provided by the research area of domain analysis, which investigates forms to model domains. Domain models are artefacts produced at the early stage of the software development process and illustrates specific conceptual knowledge in a problem domain, excluding irrelevant knowledge from this domain.

Relevant to our work is ontology-based domain analysis. The term Ontology originates in philosophy, where it was used to characterize a science about the nature of things: the types of things that exist and the relationships among them. In Computer Science environments, ontologies have been used, among others purposes, for building Domain Oriented Software Development Environments. Incorporating ontologies in the Software Engineering life cycle offers several advantages [23], ontologies seem to be well suited for an evolutionary approach to the specification of requirements and domain knowledge. Moreover, ontologies can be used to support requirements management and traceability. Formal specification may be a prerequisite in the design and implementation phases, in model driven approaches. In this context, ontologies can be incorporated into the DSML design phase, instead of other techniques [2], to produce a domain model. A domain ontology specifies the invariant conditions of the domain of interest which should be respected by any model built for that domain. The transition from an ontology to a domain modeling language might be obtained by a transformation between a language expressing the ontology and the DSML. It is also possible to apply domain ontologies to conceptual schema development by extending existing conceptual modeling languages, such as the Unified

Modeling Language (UML), with domain-specific language profiles that are defined through domain ontologies [6]. Since, ideally, domain ontologies should be grounded in foundational ontologies [10], we decided to develop our domain ontology taking as basis the Unified Foundational Ontology (UFO) [12], [11]. This choice is motivated by our concern in building a grounded domain ontology that could be expanded, compared or merged with other ontologies. Domain ontologies that use common foundation ontologies can be merged easily or compared. UFO is a foundational ontology that has been based on a number of theories from Formal Ontology, Philosophical Logics, Philosophy of Language, Linguistics and Cognitive Psychology. UFO has been used to evaluate, redesign and integrate (meta) models of conceptual modeling languages, as well as to evaluate, re-design and give real-world semantics to domain ontologies. It is composed by three main parts. UFO-A is an ontology of endurants, and it is the core of UFO. UFO-B is an ontology of perdurants (events) that is suitable to represent discrete events simulations. UFO-C is an ontology of social entities (both endurants and perdurants) built on the top of UFO-A and UFO-B. UFO-C distinguishes agents and objects. Agents are capable of performing actions with some intention, while objects only participate in events. UFO is appropriate to build our domain ontology as it provides support to represent ABM simulation through UFO-B and UFO-C.

### **3.2 Agent-Based Modeling and simulation for Land Use/Cover Change**

Some authors have recently addressed the problem of communicating about ABM. The ODD (Overview, Design concepts, Details) protocol, described in [7] has been defined for describing individual-based models and agent-based models. ODD is aimed at describing these types of models in scientific publications and is essentially focused on communication and reimplementation of ABM. But ODD is designed to describe only one definite model version [8] and can not be directly compiled to computer code. ODD can only represent the first step on the way to establish a general protocol for describing individual- and agent-based models [21]. Other authors have concentrated in ABM representation issues. In [13], a development environment for the definition of multi-agent systems (MAS), called Dsml4mas, is proposed. It is a model-driven framework that encompasses the platform independent specification of MAS, the model validation, the model transformation and code generation, as well as the execution of generated source code. However, Dsml4mas does not offer the possibility to represent important elements of the ABM/LUCC domain such as the agents' environment. Moreover, it is not based on a foundational ontology.

In [3], authors present an ontology defining an agent-based simulation framework and discuss the possibilities for using the Web Ontology Language's (OWL) automated reasoning capabilities. How to benefit from OWL and Semantic Web technologies for simulation is also the topic of other works. In [22] Polhill and colleagues illustrate how deploying an agent-based model on the Semantic Grid facilitates international collaboration on investigations using such a model, and

contributes to establishing rigorous working practices with agent-based models as part of good science in social simulation. The experimental work-flow is described explicitly using an ontology, and a Semantic Grid service with a web interface implementing the work-flow. Users are able to compare their parameter settings and results, and relate their work with the model to wider the scientific debate. But this work mainly focuses on simulations parameters and the ontology used does not describe the models themselves.

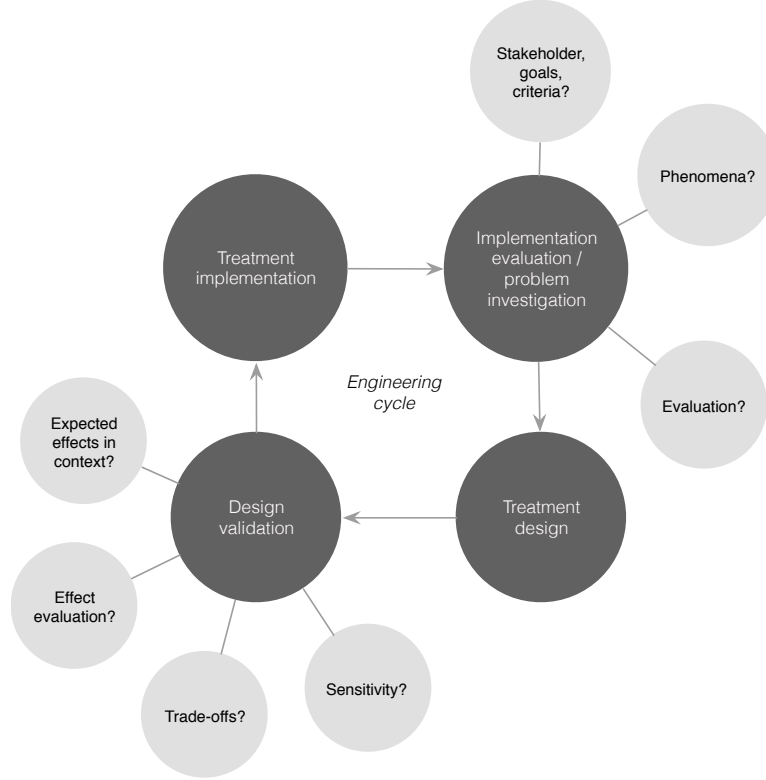
MR POTATOHEAD [18] is another approach that tackles the design of agent-based models of land use change. This framework involves the creation of a standard design patterns at the conceptual level to enable the comparison of agent-based models of land use/land-cover change. MR POTATOHEAD is tailored to a particular subset of models and enables a more detailed comparison to be made than the more generally applicable ODD. The MR POTATOHEAD framework is implemented using OWL.

Polhill and Gotts [20] presented another interesting approach to address ABM/LUCC transparency issues. The authors propose to implement ABM simulations using ontologies, instead of object-oriented languages. Their work illustrates that using OWL to represent ABM as an ontology and executing an inference process, enables representing each step of a simulation as a particular ontology. This approach responds directly to transparency issues such as model validation because it is possible at any time to access the state of simulation and understand how a model behaves. However, the implementation in OWL limits the number of entities that can be represented in the model, as opposed to OO languages. Authors also point out that ontologies are not best suited to describe the algorithmic processes by which the state of the model is changed. Our ontology seems better suited for this task as it relies on the OO paradigm. All of these works represent a step forward to more transparent ABM but they individually only respond partially to the issues we want to address in our research.

## 4 Research Methodology

As the research project sought to produce and assess an artifact (a DSML), we based our research approach on other approaches that focus on the development and performance of (designed) artifacts in a context. Design science [15] and Action Research [5] are two research approaches that aim to increase the relevance of research by incorporating a social problem-solving activity in research without sacrificing rigour. Some authors ([5] and [17]) have combined both to offer guidelines and frameworks to conduct research projects. To achieve our research goals, we designed a research approach that carries out a series of tasks that are synthesized in figure 1. We included, in the methodology, some of the concepts advocated in Design Science, which include artefacts validation and design evaluation. The problem investigation phase is supported by a systematic literature review (SLR) [16], in order to rank the papers of main journals and conferences in our domain which tackle the issues related with ABM/LUCC specification.

Based on this selection, we will systematically characterize the consequences of ABM/LUCC transparency issues and select main stakeholders.



**Fig. 1.** Research methodology, adapted from [24]

Before starting to design the DSML, we will conduct a survey to evaluate how scientists and practitioners capture requirements for their systems and how they perform the analysis activity. The target population for answering the survey is composed by our stakeholders: the authors of the papers that have been ranked and selected during the SLR. The survey will also focus on the CASE tools that are used to model ABM/LUCC and on the properties that a DSML should provide for stakeholders. The treatment design phase focuses on the design of the DSML. The initial stage of this phase consists in the definition of a domain model supported by a domain ontology (see section 5). During the treatment design phase, we will need to build an editor that enables the construction of conceptual schemas of the phenomena belonging to the domain. It requires the creation of an UML profile based upon the domain ontology that specifies a domain axiomatization in terms of concepts, relationships between concepts and the rules that govern these relationships (i.e. the invariant conditions that define

the domain). This operation will be implemented using the Object Constraints Language (OCL). We will target a specific Agent-based Modeling and Simulation Platform (ABMSP) <sup>3</sup> to produce the code corresponding to the implementation of a conceptual schema of a domain subsystem. The outcome of these implementations are simulation runs which provide insight to model designers and decision makers. To properly test our DSML, we will perform, during the design validation phase, a usability and a quality analysis of our DSML. These analyses should provide insight on how users manage to learn and use the language and what is the outcome of the modeling of the same subsystem, by different groups of users. We will first evaluate the DSML usability with a population of graduate students, to provide feedback from preceding to the following tasks and carry out necessary adjustments and correction to the DSML. In the last phase, the Treatment Implementation, we will scale up the evaluation using domain experts. As advocated in [5], this operation can be performed using the Action Research approach and will allow to measure the acceptance of the artifact using a population of targeted stakeholders.

## 5 First results

After a first attempt to characterize the ABM/LUCC domain [9], we decided on a more grounded approach to construct our domain model. As tackled in previous sections, we opted to build a domain ontology based on UFO where relevant MAS/LUCC concepts to be considered for the language are included. In the context of this paper, we will only illustrate a small part of the ontology including some of the concepts specific to the MAS/LUCC domain. Figure 2 shows a fragment of the UML classe diagram illustrating the type of agents that can be defined and the representation of the agents' environment. It illustrates that two types of agents exist in ABM/LUCC. Stereotypes represent the foundational ontology concepts and the class names, the domain concepts. For instance, to represent groups of agents with a social role, the UML class SocialAgent is created with the « InstitutionAgent » stereotypes which represents the equivalent concept in the eUFO-C3 ontology layer.

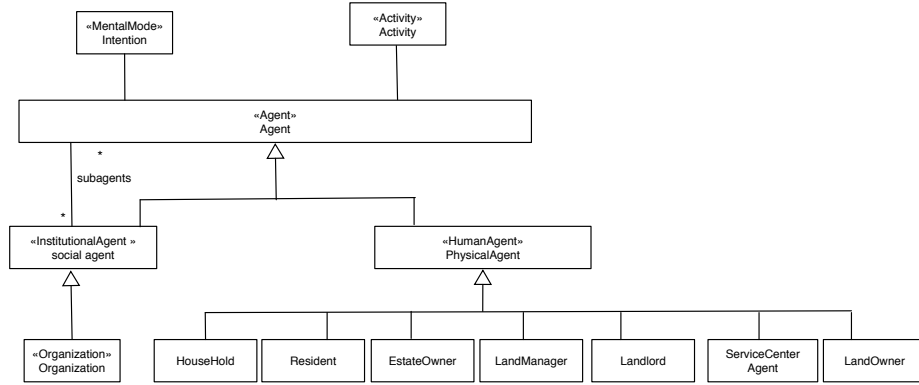
The classes describe our domain concepts and classes' stereotypes represents the corresponding UFO ontology concepts.

### Working Example

The Ontology is currently under a validation process, which consists in adapting a published model and verifying if it supports the ontology. We have chosen to represent SLUDGE model (Simulated Land Use Dependent on eDGe Effect externalities), [21] for its representativeness of ABM/LUCC. SLUDGE is a simple combined cellular automaton and agent-based model designed to explore the effects of positive and negative distance-dependent spatial externalities on

---

<sup>3</sup> such as RePast ([repast.sourceforge.net](http://repast.sourceforge.net))



**Fig. 2.** Extract from the MAS/LUCC domain ontology - the agents' layer

economic and landscape pattern outcomes. The first results of this validation process allowed to highlight some limitations of the ontology to represent spatial concepts. We are now working on incrementing the required spatial references.

## 6 Expected contribution

By providing a DSML to the ABM/LUCC community we aim at providing its users with a common platform to represent and communicate the systems and sub-models they are working on. We also aim to contribute to the validation and reuse of the sub-models by reducing the gap between conceptualization and implementation. We also expect that the choice we made about using a foundational ontology as a basis for our domain ontology will facilitate the adoption and extension of our DSML to other related domains.

## 7 conclusions and further work

In this paper, we have introduced transparency issues in ABM/LUCC domain. We have proposed the development of a Domain Specific Modeling Language for this domain and described the methodology we will follow to respond to the research questions we have raised. We also presented a domain model based on a domain ontology. The next step in our work will consist in validating the domain ontology by confronting its capacity to represent MAS/LUCC models existing in the literature. After this validation, we will pursue by designing a UML profile for the ontology domain axioms and formalize them as Object Constraint Language (OCL) constraints. We will also implement an editor in Eclipse that will enable the design of conceptual schemas in conformity with the UML profile.



## References

1. F. Bousquet and G. Trébutil. *Introduction to companion modeling and multi-agent systems for integrated natural resource management in Asia*, pages 1–17. IRRI Publications, Los Banos, Philippines, 2005.
2. I. Ceh, M. Crepinsek, T. Kosar, and M. Mernik. Ontology driven development of domain-specific languages. *Comput. Sci. Inf. Syst.*, 8(2):317–342, 2011.
3. S. Christley, X. Xiang, and G. Madey. An ontology for agent-based modeling and simulation. *Agent 2004 Conference on Social Dynamics: Interaction, Reflexivity and Emergence*, 2004.
4. A. v. Deursen, P. Klint, and J. Visser. Domain-specific languages: An annotated bibliography. *ACM SIGPLAN Notices*, 35(6):26–36, June 2000.
5. H. Fatemi, M. J. van Sinderen, and R. J. Wieringa. A trust ontology for business collaborations. In K. Sandhu, U. Seigerroth, and J. Stirna, editors, *Short Paper Proceedings of the 5th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling, Rostock, Germany*, volume 933 of *CEUR Workshop Proceedings*, page 5, Aachen, November 2012. CEUR-WS.org.
6. F. Gailly and G. Poels. Conceptual modelling using domain ontologies: Improving the domain-specific quality of conceptual schemas. Working papers of faculty of economics and business administration, Ghent University, Faculty of Economics and Business Administration, Belgium, 2009.
7. V. Grimm, U. Berger, F. Bastiansen, S. Eliassen, V. Ginot, J. Giske, J. Goss-Custard, T. Grand, S. K. Heinz, G. Huse, A. Huth, J. U. Jepsen, C. J  rgensen, W. M. Mooij, B. M  ijller, G. Pe    zer, C. Piou, S. F. Railsback, A. M. Robbins, M. M. Robbins, E. Rossmanith, N. R  ijger, E. Strand, S. Souissi, R. A. Stillman, R. Vab    , U. Visser, and D. L. DeAngelis. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198(1    2):115 – 126, 2006.
8. V. Grimm, U. Berger, D. L. DeAngelis, J. G. Polhill, J. Giske, and S. F. Railsback. The odd protocol: A review and first update. *Ecological Modelling*, 221(23):2760 – 2768, 2010.
9. C. Grueau and J. Araujo. Towards a domain specific modeling language for agent-based models in land use science. In *Proceedings of the 28th Symposium On Applied Computing*, Coimbra, Portugal, 2013.
10. N. Guarino. *Formal Ontology in Information Systems: Proceedings of the 1st International Conference June 6-8, 1998, Trento, Italy*. IOS Press, Amsterdam, The Netherlands, The Netherlands, 1st edition, 1998.
11. G. Guizzardi. *Ontological foundations for structural conceptual models*. PhD thesis, University of Twente, Enschede, The Netherlands, October 2005.
12. G. Guizzardi and G. Wagner. Towards an ontological foundation of agent-based simulation. In *Winter Simulation Conference*, pages 284–295, 2011.
13. C. Hahn. A domain specific modeling language for multiagent systems. In *Proceedings of 7th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 233–240, 2008.
14. A. Heppenstall, A. Crooks, L. See, and M. Batty, editors. *Agent-based Models of Geographical Systems*. Springer: Dordrecht. Publication, 2012. <http://www.springer.com/social+sciences/population+studies/book/978-90-481-8926-7>.
15. A. R. Hevner, S. T. March, J. Park, and S. Ram. Design science in information systems research. *MIS Quarterly*, 28(1):75–105, 2004.

16. B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman. Systematic literature reviews in software engineering - a systematic literature review. *Inf. Softw. Technol.*, 51(1):7–15, Jan. 2009.
17. A. Lee. Action is an artifact. In N. Kock, editor, *Information Systems Action Research*, volume 13 of *Integrated Series in Information Systems*, pages 43–60. Springer US, 2007.
18. D. C. Parker, D. G. Brown, J. G. Polhill, and P. J. M. Deadman. *Agent-Based Modelling in Natural Resource Management*, chapter Illustrating a new conceptual design pattern for agent-based models of land use via five case studies: the MR POTATOHEAD framework, pages 29–62. Valladolid, Spain, 2008.
19. D. C. Parker, S. M. Manson, M. A. Janssen, M. J. Hoffmann, and P. Deadman. Multi-agent systems for the simulation of land-use and land-cover change: A review. *Annals of the Association of American Geographers*, 93(2):314–337, 2003. <http://dx.doi.org/10.1111/1467-8306.9302004>.
20. J. Polhill and N. Gotts. Ontologies for transparent integrated human-natural system modelling. *Landscape Ecology*, 24(9):1255–1267, Nov. 2009.
21. J. G. Polhill, D. Parker, D. Brown, and V. Grimm. Using the odd protocol for describing three agent-based social simulation models of land-use change. *Journal of Artificial Societies and Social Simulation*, 11(2):3, 2008.
22. J. G. Polhill, E. Pignotti, N. M. Gotts, P. Edwards, and A. Preece. A semantic grid service for experimentation with an agent-based model of land-use change. *Journal of Artificial Societies and Social Simulation*, 10(2):2, 2007.
23. S. Seedorf, F. F. Informatik, and U. Mannheim. Applications of ontologies in software engineering. In *2nd International Workshop on Semantic Web Enabled Software Engineering (SWESE), held at the 5th International Semantic Web Conference (ISWC)*, 2006.
24. R. J. Wieringa and A. Morali. Technical action research as a validation method in information systems design science. In *Design Science Research in Information Systems. Advances in Theory and Practice 7th International Conference, DESRIST 2012, Las Vegas, USA*, volume 7286 of *Lecture Notes in Computer Science*, pages 220–238, London, May 2012. Springer Verlag.