Formal Modeling of Social-Ecological Systems

Christophe SIBERTIN-BLANC[a], Olivier THEROND[b,c], Claude MONTEIL[d,e] and Pierre MAZZEGA[f]

[a.] Université Toulouse 1- Capitole, IRIT, F-31042 Toulouse cedex, France
sibertin@irit.fr
[b.] INRA, UMR 1248 AGIR, F-31326 Castanet Tolosan, France
olivier.therond@toulouse.inra.fr
[c.] Université de Toulouse; ENSAT; UMR 1248 AGIR, F-31326 Castanet Tolosan, France.
[d.] Université de Toulouse; ENSAT; UMR 1201 DYNAFOR, F-31326 Castanet Tolosan, France
monteil@ensat.fr
[e.] INRA; UMR 1201 DYNAFOR, F-31326 Castanet Tolosan, France.
[f.] LMTG / GET UMR5563, IRD – CNRS – Université P. Sabatier Toulouse 3, F-31400, France
pierre.mazzega@lmtg.obs-mip.fr

Abstract. The success of Integrated Assessment and Modeling of social-ecological systems requires a framework allowing the members of such a process to share and gather their respective knowledge about the system under consideration and to get confidence into the reliability of the software that implements the system's model they have produced. To this end, this paper presents an Entity-Process meta-model of SESs and outlines its use.

1 Introduction

Social-ecological systems (SES) are considered as complex adaptive systems that are characterised by self-organisation and distributed control. They encompass multiple actors with diverse and contrasting management interests and objectives, acting at different spatial and temporal levels (Reed, 2008; Pahl-Wostl, 2007; Giampietro 2002). In such systems, social, economic and ecological processes, in interaction at many organization levels, give rises to emergent structures and functions at the whole SES level.

During the last decades integrated assessment (IA) has been playing an increasing role to better understand how complex systems function. Rotmans and Asselt (1996) define integrated assessment as an interdisciplinary and participatory process to combine, interpret and communicate knowledge from diverse scientific disciplines. IA approaches to assess system responses to scenarios of change have been commonly used to design sustainable management and development strategies. Mainstream in IA is based on modeling and simulation. This Integrated Assessment and Modeling (IAM) approach aims at describing quantitatively the causal relationships and the interactions between the various components of the investigated system. According to the problem at hand, IAM approach assembles the relevant information from a wide range of scientific disciplines and put them into a "policy oriented context" in order to analyze complex system behaviour (Toth and Hizsnyik, 1998). The potential of model-based methods is well-established for handling management and policy problems (Sterk et al., 2009; McIntosh et al., 2007; Jakeman et al, 2006; Oxley et al., 2004).

When applying IAM to social-ecological systems, one of the main challenges lies in the integration of many different disciplinary viewpoints and forms of knowledge. One fruitful way to reach this objective consists in building up a well-funded and shared ontology of the SES, i.e. a formal specification of the concepts and relationships among these concepts (Beck et al., 2010).

Another challenge is the confidence of the participants of the modeling process in the simulation results. This confidence is problematic if, once they have built the model together, they lose the control on the following of the process, that is the implementation by the computer scientists of the model as a piece of software. The use of a meta-model allows to produce a well-structured model whose implementation suffers from no technical distortion.

This paper presents an ontology expressed as a meta-model, intended to provide scientists and stakeholders from various competencies and backgrounds with a generic and formal framework to coherently arrange their respective knowledge about SESs. It guides collective modeling processes aiming to assess SESs' scenarios by using simulation models. Examples of models and corresponding specific diagrams (with a well-defined semantics) used for their graphical representation are presented in another paper (Therond et al., 2011).

2 An Entity-Process Meta-model for SESs Modelling

Working in an interdisciplinary environment requires a shared conceptual framework for a collective interpretation of the concepts put forward by one another. Accordingly, our meta-model uses terms understandable and usable both by stakeholders and by a large scientific community, rather than referring to domain specific terminologies.
The primary function of the meta-model is to provide a formal framework facilitating the coherent description of SESs as formal models. The meta-model identifies the types of the constitutive elements that are considered in the design of SESs' models and how they are related, as shown in Fig. 1 represented as an UML class diagram. According to this meta-model, the model of a SES includes a set of entities and a set of processes, together with relations between elements of these sets. The *structure* of a system is composed of entities and relations between entities, both being perceivable and more or less persistent. The dynamic character of SESs is handled through processes that involve one or more entities of the system, and make the state and also the structure of the system to evolve. Finally, we account for interactions between a SES and its environment.

2.1 System structure: entities and relations

The meta-model distinguishes three categories of *entities*: material resources, cognitive resources and actors. Instances of entities are the considered individual resources and actors that may appear or disappear over time, or experience changes of their state. Entities are characterized by *properties* whose values constitute the *state* of an entity instance. They are endowed with *operations* that process these property values.

There are two types of *relation* between entities. A structural relation permanently associates entities related by their very nature (e.g. field – soil type). A non-structural relation puts in association entities as the result of some action (e.g. farmer ([wner of an acquired] field). A *link* is an instance of a relation; it ties instances of the related entities.

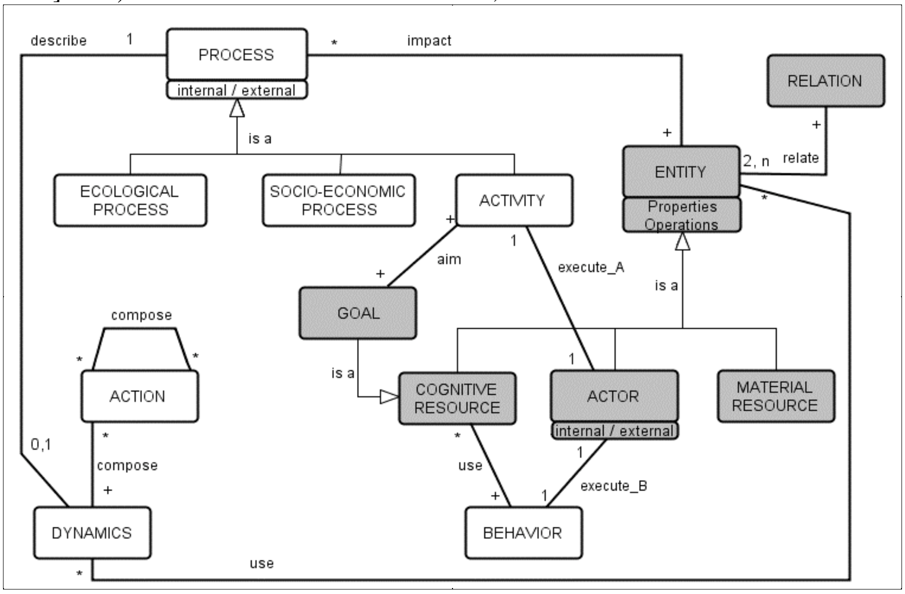


Fig. 1: A meta-model for the formal modeling of social-ecological systems represented as a UML class diagram. Cardinalities use the following conventions: *: any number of elements (including none) ; +: at least one element ; n: exactly n elements.

Actors are human agents, be they an *individual*, a *population* of similar individuals or a *group* such as an organization, association, committee, enterprise, etc. Each actor *executes* at least one activity.

Resources are objects of uses, or objects for uses. *Material resources* are physical resources spatially and temporally distributed (e.g. a water body, a field plot, a crop). *Cognitive resources* stay in the minds of human beings: they are information, believes or expectations about facts, procedures, values or aims, that humans use or consider in the activities they undertake, in designing strategies or in the formation of goals. A cognitive resource is for a single actor (e.g. the cropping system of a farmer) or shared by several actors (e.g. the weather forecasts). Many cognitive resources have a material counterpart – for example a legal norm (cognitive resource) has an official publication support (material resource) – which is most often not relevant to consider. It should be stressed that any knowledge about a material resource used by an actor or a process implies the existence of an own cognitive resource; such a cognitive resource will be considered in the model only if required (e.g. if its proprieties do not correspond to those of the material resource). Every resource must be either impacted or used by at least one process (this statement is the main integrity constraint of the meta-model, see 3.1 below).

At each time the *structure* of a system is defined as the sets of its entity instances and of its structural links (the instance of the structural relations). At each time the system's *state* is defined as the states of these entities (i.e. the values of their properties) together with the existing non-structural links.

2.2 System dynamics: processes and behaviors

The *processes* generate the observed phenomena and make the structure or the state of the system to change. Each execution of a process *impacts* entities by changing their property values, or by creating or destroying entity instances or links. We distinguish three types of processes: activities, socio-economic processes and ecological processes.

An *Ecological process* corresponds to an enactment of a biophysical law. It determines the evolution of the states of material resources (and possibly of actors).

A Socio-economic process generates phenomena resulting from the human beings' economic or social activities. Only the effects of a socio-economic process is described in the model, considering that it is not necessary to represent the actors' activities at the origin of this phenomenon, that is how it is generated.

An *Activit*y is a process *executed* by an actor intending to achieve some *goal*s. A goal is linked to the activity *aiming* at its achievement.

Each actor is associated with one *behavior* that describes how he orchestrates his various activities, solves the conflict that could appear between them, and decides to execute (or not) at a given time an action rather than another one.

The description of a process is broken down into *actions* that can themselves be decomposed into *sub-actions*. The *dynamics* of a process describes the scheduling of actions executions as a system of equations, an automaton, an algorithm, or a set of (event-condition-action) rules, according to the most appropriate representation for the considered process. A process dynamics *uses* entities (resources or actors) whose states condition its development.

Processes *interact* by means of entities. Indeed, it is through changes in the states of the impacted entities that a process affects the course of the processes that depend on these entities.

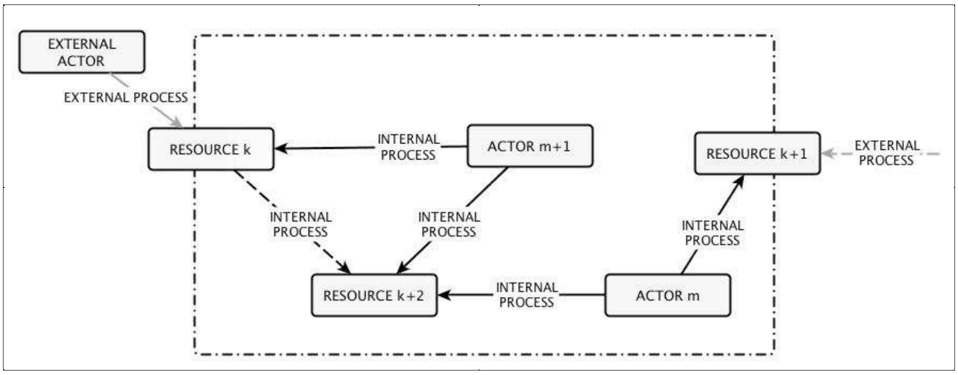


Fig. 2: The interactions between a SES and its environment (outside the dashed-dotted line box).

2.3 Interactions between the system and its environment

Social-ecological systems are open systems that interact with their *environment*, that is the entities and processes of the reality that are outside the system while having an influence on it. The meta-model considers these interactions through the setting of resources at the interface of the system. We call *external process* a process whose execution is not affected by the state of the modeled system and which changes the state of some entities that are, from this fact, qualified as *interface entities* (see Fig. 2). An interface entity may be impacted by both external and internal processes and it is similar in nature to other entities. Classically the dynamics of an external process is not represented but rather its impacts. These impacts are commonly framed as a (spatio-) temporal series, representing for each time step of the simulation, the new values (or the variations) of interface entities' properties (e.g. climate data).

We call *external actor* an actor who performs only external activities.

Outputs of the system to the environment are considered only whether they constitute indicators to be considered.

3 The Modeling Process

Any interdisciplinary modeling of a SES requires, in addition to a conceptual framework such the one presented in the previous section, a methodological framework to structure and support the modeling process. The structure of the model to be built guides the structure of the modeling process by defining the objects it must produce. We here mainly address the main issues related to a tractable representation of models conforming with our meta-model, in order to illustrate its practical usability.

The meta-model described in section 2 is a tool for supporting integrated modeling of a given *reference system*. By "reference system" we mean the collective representation that observers build of the knowledge they share about the reality they investigate regarding a given question (see Fig. 3). The first step of the modeling process consists in an instantiation of the meta-model into a *conceptual model* that identifies relevant entities (actors, material resources and cognitive resources), relations and processes when considering the question the model is intended to answer.

Practically, several conceptual models are produced in the course of a progressive and iterative modeling process, along with the improvement of the understanding of the question at hand. They differ by their levels of granularity or of generality, their focus on the core of the system, or their boundary. The conceptual model to be simulated must be internally consistent in terms of (spatial, temporal, sociological, physical, etc.) granularity and ontological level of the represented entities and processes.

The structural part of the conceptual model is in turn to be instantiated into a *concrete model* of the reference system. Indeed, the entities and relations of the conceptual model are just the *types (or classes)* of the reference system's entities and relations to be considered. For example, if the conceptual model includes a *"Water Agency"* actor, the concrete model will include a single instance of this type whose properties values describe the particularities of the Water Agency at hand; a cognitive resource "*Weather forecast"* will be instantiated as many different instances (one for each time periods of the simulation); a material resource "*Field plot"* will be instantiated into the number of concrete plots necessary to cover the agricultural territory of the reference system. The same holds for the relations between entities: a relation "*owns*" between an actor "*Farmer"* and a resource "*Field plot"* in the conceptual model will give rise in the concrete model to concrete *owns* links between each *Farmer* instance and the field plot instances he posses.

Finally, the concrete model will be implemented by the development of a computer program, the *simulation model*.

The conceptual model of a SES includes too many elements to be represented at one single glance. Thus, its representation must be broken down into several *diagrams,* each one focused on an aspect and using specific notations. All these diagrams must be documented with the explanations necessary for their unambiguous understanding by all the participants of the modeling process. Indeed, the comprehensive view of the system's representation that is embedded into the model is necessary for the correct interpretation of the simulation results, and thus for the achievement of the modeling process's goals. All the diagrams regarding the system's structure and dynamics must be carefully aligned to ensure the global coherence of the system's conceptual model. Accordingly, this alignment is obtained through back and forth steps between the various diagrams.

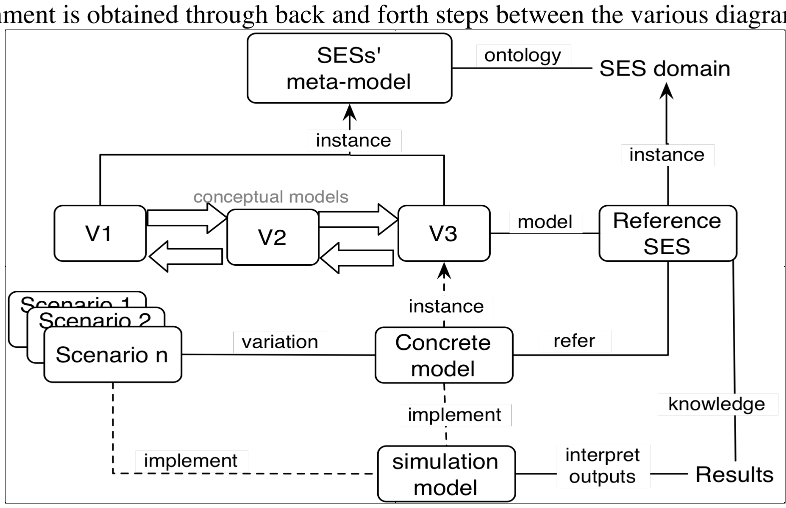


Fig. 3: The different models and their relations to the reference system.

3.1 Actor-resource diagrams and integrity constraints

The *actor-resource diagram* shows the actors and the resources of the conceptual model together with their relations, that is the structural part of the model.

It may be represented by means of the widely used UML notation for class diagrams (OMG, 2005), but it could be represented using the Entity-Relationship formalism as well (Chen et Pin-Shan, 1976)**.** To distinguish the actors, material resources, cognitive resources and interface entities, their surrounding boxes may have different colors or shape (see Plate 1 (a) for the conventions we use).

It is likely that the readability constraints of the actor-resource diagram do not agree with a layout on a single page. Helpfully the diagram can be split into several sub-diagrams, leading to the identification of sub-domains of the system's model.

An actor-resource diagram must be accompanied with the statement of its *integrity constraints*. Integrity constraints are restrictions on the instantiation of entities and relations that ensure the possibility to interpret the concrete model as a description of a coherent and feasible world. Cardinalities of the relations express some of such constraints, e.g. in the UML class diagram of Fig. 1, each process has to impact at least one entity. Other constraints deserve to be explicitly stated, which bear on the values of properties of an entity or of linked entities, or on the existence of links or entities. For example, in a conceptual model including actors "*Farmer*", the value of the property "*income*" of any farmer must be the sum of the incomes of the fields crop he works.

It is often the case that a system' model requires to refer to different theories that represent a same item of the reality by different entities (Quesnel et al., 2007). For example, the representation of water flow and plant growth processes often needs to represent the investigated area by resources that differ with regard to their properties or spatial delineation. Whether the area is represented through different resources by the two processes, the state of resources used and impacted by the plant growth process must take into account the effect of the water flow process and conversely. Integrity constraints allow specifying how the relations between the states of the two types of resources have to be handled. The UML proposes to formally express these constraints using the Object Constraints Language (OMG, 2010) but any means for unambiguous statement may be used.

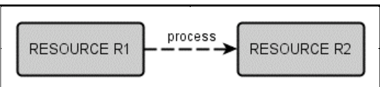
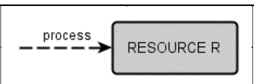
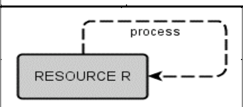
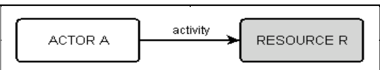
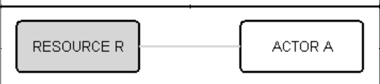
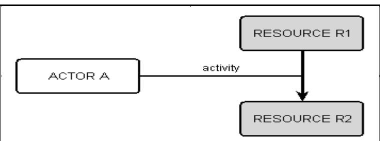
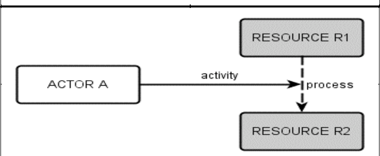
(a)		Graphical conventions for the entities (actor, material resource and cognitive resource respectively)
(b)		An ecological or socio-economic process that impacts resource R2 is dependent on the state of resource R1.
(c)		An ecological or socio-economic process impacts resource R.
(d)		An ecological or socio-economic process being dependent on the state of the resource it impacts.
(e)		Actor A performs an activity that impact the resource R.
(f)		Actor A uses the resource R to execute at least one of his activities; the relation can be labeled or not.
(g)		Actor A performs an activity that simultaneously impacts resources R1 and R2, their respective states being inter-dependent.
(h)		Actor A performs an activity that changes the realization of a process being dependent on the state of resource R1 and impacting resource R2.

Plate 1: Graphical conventions for entities are given in panel (a). In panels (b) to (h) resources can be indifferently material or cognitive.

3.2 Process diagrams

The *process diagram* displays the processes and the entities they impact, the actors performing the activities and the most relevant entities used by their dynamics. For readability, it may be broken down into several sub-diagrams. To get a synthetic representation, Plate 1 suggests graphical conventions for basic patterns.

To be illustrative, Fig. 4 shows the process diagram of the quantitative water management in a French association of irrigators. In such associations the water withdrawals are generally carried out through collective equipment (pump) and water is distributed to irrigators' fields by water pipes. Association often withdraws water in several water resources like reservoir hills and rivers. Water withdrawals in river are generally limited by a water quota attributed to the association. The association defines its management rules to distribute available water resources among association's members. The president of the association is generally one of the irrigators and often plays the role of coordinator and regulator. The process diagram of Fig. 4 presents is aimed at simulating impacts of management rules scenarios on different indicators regarding water withdrawals (quantity and dynamics), nitrogen pollution and farmer income. The considered time horizon of scenarios is a short term one (about 15 years).

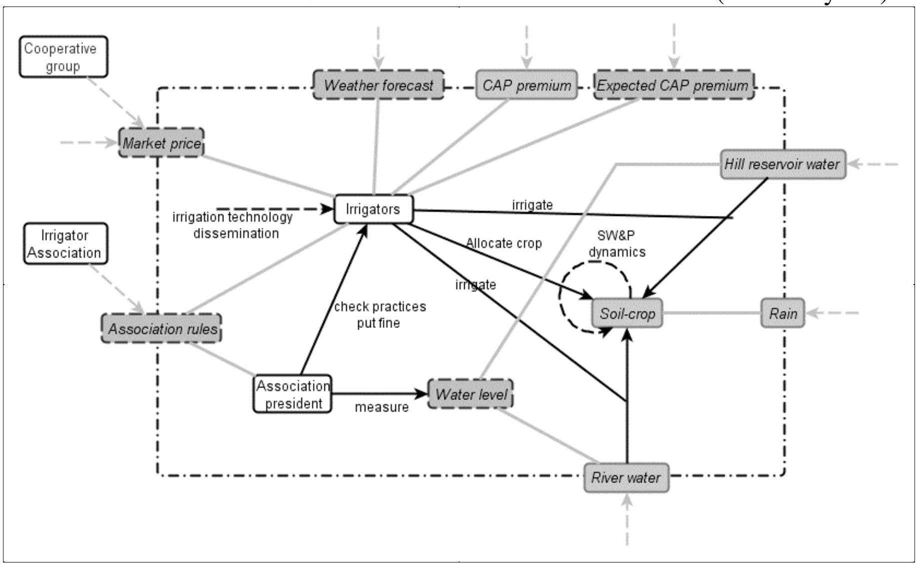


Fig. 4: A process diagram of water management at the scale of an irrigator association in France. SW&P means "Soil Water and Plant". See Plate 1 for the graphical conventions. Resources positioned on the dashed-dotted boundary line are interface resources.

In the model presented in Fig. 4 two types of actors perform activities: irrigators and the association president (AP). "Cooperative group" and "Irrigator association" are external actors. They are represented to ensure that modelers share understanding of the nature and origin of interface resources. Irrigators execute two activities: "allocate crop acreage" and "irrigate". The AP checks the compliance of irrigators' behaviors with the association rules and if necessary puts fine to irrigators to regulate their behavior. The resource "Water level" is generated by the AP's activity "measure". Both the AP and irrigators use "Association rules" as cognitive resources. The definition of association is provided by external process scenarios. The relations between the cognitive resource "Water level" and the material resources "River water" and "Hill reservoir water" express integrity constraints between these entities: "Water level" directly corresponds to "River water" and "Hill reservoir water". To execute the activity "allocate crop acreage" irrigators use "Market price", "Expected CAP premium" as resources. To perform "irrigate" they use the cognitive resource "Weather forecast".

Only one internal ecological process is considered in this model: "soil water and plant dynamics" (SW&P dynamics). It allows simulating crop growth, water and nitrogen cycle and crop yield in each irrigated field plot. The graphical features of plate 1 allow representing that the activity "irrigate" performs a water transfer from rivers or hill reservoirs to soil-crop resources (i.e. plot field).

Only one internal socio-economic process is considered in this model: irrigation technology dissemination. It allows representing evolution of agricultural irrigation practices over the temporal extent of scenarios without considering by who and how it is performed.

3.3 Dynamics and Behavior diagrams

A *dynamics diagram* describes the dynamics of one process, so that there is a dynamics diagram for each internal process. It describes the actions of the process and which ones must be performed at each time step of the simulation. As there are many ways to formalize the dynamics of a process, no standard representation is proposed. What about the multiplicity of the entities instances of the concrete model impacted by the execution of a process at each time step of the simulation? The better is to consider that each process occurs only once and applies to every (concrete) instance of the impacted entities. As an example in the model shown in Fig. 4, every occurrence of the *Soil Water & Plant dynamics* process applies to each instance of the *Soil crop* resource. The correct time and spatial resolutions of the model should be the finest one making sense for all the processes.

The processes interact through the entities of the model's structure. To ensure that they act within the same structure, there must be no ambiguity about the identities and the properties of the entities involved in their respective executions. They must also specify the entities' operations they need for their processing, in order to complete the actor-resource diagram. A dynamics diagram is to be precise enough so that it can be implemented as a computer routine without ambiguity.

An actor's *behavior diagram* describes how an actor having several activities to execute selects, among the actions of these activities enabled by the current system's state, the ones to be effectively executed. The various activities of an actor should be described using the same formalism, which can also be used for the description of the actor's behavior diagram.

3.4 Scenarios and indicators

Scenarios intend to study the effect upon the behavior of the system of various hypothesizes. They are expressed as variations of the concrete model. They can encompass changes into (i) the dynamics of internal processes, notably their parameters

(ii) the impact of external processes upon interface entities, or (iii) the initial states of entities and links of the concrete model. In our example on water management (see Fig. 4), the scenarios of rainfall over several decades are introduced in the form of spatiotemporal series on the hydrological.

The integrated assessment of scenarios is achieved by analyzing the value of *indicators* derived from the final value or from the evolution, in the course of the simulation, of the state of entities of interest for scientists and stakeholders.

3.5 Documentation of the concrete model and scenarios

The system's *concrete model* describes the initial structure and state of the system to be simulated that is the instances of the entities and their state, together with their links. It includes too many elements to be displayed as diagrams. A comprehensive documentation of the concrete model must describe:

1. the concrete instances of the model's entities, including the values of their properties and their links, together with the sources of the data used to define these instances; if the instances of some entities are very numerous, they are not described individually but by a probabilistic law that distributes the values of their properties;
2. the sources of the data that are used to feed the external processes and how they are processed, if necessary, to fit the structure of the interface entities;
3. an implementable specification of the dynamics and the actions of each process;
4. the algorithms allowing to calculate the values of indicators.

For each scenario, the changes have to be documented in the same way.

3.6 Simulation model

The *simulation model* is an implementation of the concrete model (or one of the scenarios) as a software[3]. Classically in IAM approaches the simulation model looks like a black box for most of the participants to the modeling process (Resnick et al., 2000). They do not know to what extent the simulation model is a faithful implementation of the concrete one. This lack of comprehensive view of the simulation model often leads to little or even no confidence in the simulation results.

Using the entity-process meta-model, the structure of the simulation model can mirror the one of the concrete model and so prevent the occurrence of distortions between the two models. Using an object-oriented language such as Java, each entity of the conceptual model is implemented as an object class provided with the corresponding attributes and methods. The relations are implemented as entities' attributes, according to the navigability[4] of the relations that can be deduced from the processes' actions. These classes will be instantiated as objects according to the concrete model. Each process will be implemented as a class including class attributes and a method that

[3] According to the terminology of logic, the concrete model would be considered as a theory, and the SES and the simulation model as two models (or interpretations) of this theory in the worlds of perceivable reality and software respectively.

[4] The navigability of a relation between entities specifies which of the related entities needs to access to the other.

implements its dynamics. Each actor class will be in addition provided with a method that implements its behavior.

The *simulation model* can be developed using a simulation platform providing useful services such as Cormas (Bousquet et al. 1998), Mimosa (Müller, 2010), Record (Bergez et al., submitted), Gamma (Taillandier et al., 2010), Repast (North et al. 2006) or many others - see the simulation platforms reviewed in (Schreinemachers et Berger, 2011).

Thanks to the structure of the behavior of the model, as a set of processes that interact in an asynchronous way by means of entities, the engine that makes the simulation to progress is quite simple. Assuming that the time scales of all the processes are nested, at each time step it has to carry out the following:

1. for every internal ecological and socio-economic process, activate its dynamics to select the actions to be performed;
2. for each actor, activate its behavior to select the actions of each of his activities that are enabled, and then, according to his behavior, select the action(s) to be performed;
3. perform the update of the state of the interface entities (as a result of the effect of external processes);
4. perform the actions selected by each internal bio-physical process, by each socio-economic process and then by each actor's behavior; in case of conflicts between some actions (e.g. the execution of one action disabled a selected other one), the conflict must be solved either at random or by following well-specified rules.

A *simulation* corresponds to a numerical experiment performed by applying the simulation engine to the simulation model of a given scenario. Each numerical experiment allows assessing indicators. Comparisons of indicator values of a given scenario to the ones of a reference, either the current situation or a reference scenario, allow the integrated assessment of scenarios.

Conclusion

The concepts proposed by the Entity-Process meta-model are elementary, while the proper modeling of complex SESs requires recourse to much more elaborated concepts such as "territory", "public resource" or "policy". We believe this meta-model can be specialized for specific modeling purpose (regarding the type of the investigated reference SES and investigated question) by providing rigorous definitions of higher level concepts, built upon concepts of this meta-model. This is nothing but the "profile" feature of the UML.

The meta-model presented in the paper can be used for two different purposes: to support *a priori* the Integrated Assessment and Modeling process of a SES or to make explicit *a posteriori* the conceptual model of an already implemented model or of a quite informal model that does not satisfactorily manage the complexity of the system at hand. In both cases, the formal nature of the meta-model allows a quite easy comparison of models and of their components (Bezivin 2005).

Acknowledgements. This work is funded by the Thematic Network for Advanced Research «Sciences & Technologies for Aeronautics and Space» (RTRA STAE http://fondation-stae.net/) in Toulouse, France, under the MAELIA project.

References

Beck, H., Morgan, K., Jung, Y., Grunwald, S., Kwon, H.-y., Wu, J., 2010. Ontology-based simulation in agricultural systems modeling. Agricultural Systems 103 (7), 463–477.

Bergez, J.-E., Chabrier, P., Gary, C., Jeuffroy, M.H., Makowski, D., Quesnel, G., Ramat, E., Raynal, H., Rousse, N., Wallach, D., Debaeke, P., Durand, P., Duru, M., Dury, J., Faverdin, P., Gascuel-Odoux, C., Garcia, F. RECORD: an open platform to build, evaluate and simulate integrated models of farming and agro-ecosystems. In revision at Environmental Modelling & Software.

Bezivin, J., 2005. On the Unification Power of Models. Journal on Software and Systems Modeling, 4(2):31.

Chen, P., Pin-Shan, P., 1976. The Entity-Relationship Model - Toward a Unified View of Data. *ACM Transactions on Database Systems* **1** (1): 9–36

Giampietro, M, 2002. Complexity and Scales: The Challenge for Integrated Assessment. Integrated Assessment, 3 (2–3), 247–265.

Jakeman, A. J., Letcher, R. A., Norton, J. P., 2006. Ten iterative steps in development and evaluation of environmental models. Environmental Modelling & Software, 21: 602-614.

McIntosh, B. S., Seaton, R. A. F., Jeffrey, P., 2007. Tools to think with? Towards understanding the use of computer-based support tools in policy relevant research. Environmental Modelling & Software, 22: 640-648.

Müller J.P. 2004. The mimosa generic modeling and simulation platform: The case of multi-agent systems, in : SCS. Proceedings of the 5th Workshop on Agent-Based Simulation, Lisbon, 77-86.

OMG 2005, Object Management Group, UML 2.0 Specification, August 2005, http://www.omg.org.

OMG 2010, Object Management Group, Object constraint language. OMG formally released versions of OCL, version 2.2, February 2010. URL http://www.omg.org/spec/OCL/

Oxley, T., McIntosh, B. S., Winder, N., Mulligan, M., Engelen, G., 2004. Integrated modelling and decision-support tools: a Mediterranean example. Environmental Modelling & Software, 19: 999-1010.

Pahl-Wostl, C., 2007. Transitions towards adaptive management of water facing climate and global change. Water Resources Management, 21: 49-62.

Quesnel, G., Duboz, R., Ramat, E., Traoré, M.K., 2007. VLE: a multimodelling and simulation environment, Proceedings of the 2007 summer computer simulation conference, San Diego, California, pp.367-374.

Reed, M. S., 2008. Stakeholder participation for environmental management: A literature review. Biological Conservation, 141: 2417-2431.

Resnick M., Berg R., Eisenberg M. (2000). Beyond Black Boxes: Bringing Transparency and Aesthetics Back to Scientific Investigation, Journal of the Learning Sciences 9(1):7-30.

Rotmans, J. and Asselt, M., 1996. Integrated assessment: A growing child on its way to maturity. Climatic Change 34 (3) 327.

Schreinemachers, P., Berger, T., 2011. An agent-based simulation model of humaneenvironment interactions in agricultural systems, Environmental Modelling & Software 26, 845-859Sterk, B., Leeuwis, C., van Ittersum, M. K., 2009. Land use models in complex societal problem solving: Plug and play or networking? Environmental Modelling & Software, 24: 165-172.

Therond, O., Sibertin-blanc, C., Monteil, C., Mazzéga, P., 2011. A Meta-Model to support Integrated Modeling of Social-Ecological Systems, in revision at Environmental Modelling & Software.

Toth, F. L., Hizsnyik, E., 1998. Integrated environmental assessment methods: Evolution and applications. Environmental Modeling and Assessment, 3: 193-207.