Community-based software tools to support participatory modelling: a vision

Alexey Voinov

David Arctur

Ilya Zaslavskiy

Saleem Ali

iEMSs 2008: International Congress on Environmental Modelling and Software
Integrating Sciences and Information Technology for Environmental Assessment and Decision Making
4th Biennial Meeting of iEMSs,
http://www.iemss.org/iemss2008/index.php?n=Main.Proceedings
M. Sànchez-Marrè, J. Béjar, J. Comas, A. Rizzoli and G. Guariso (Eds.)
International Environmental Modelling and Software Society (iEMSs), 2008

Community-based software tools to support participatory modelling: a vision

Alexey Voinov[a], David Arctur[b], Ilya Zaslavskiy[c], and Saleem Ali[d]

a – Chesapeake Research Consortium Community Modelling Program, Johns Hopkins University, USA, alexey.voinov@uvm.edu
b – OGC Interoperability Institute, Austin, Texas, USA
c – San Diego Supercomputer Center, CA, USA
d – Rubenstein School of Natural Resources, University of Vermont, USA

Abstract: Environmental management depends on analysis of complex dynamics and spatial relationships of ecological and socio-economic systems. Modelling, when used to conduct such analyses, is recognized as an effective decision support tool in environmental management. Modelling conducted in a participatory fashion, involving stakeholders in various stages of model building and data processing has evolved as an efficient method for conflict resolution and decision-making. However, successful participatory modelling efforts require specific software and computer tools that are not available or accessible for stakeholders. There is a clear need for specialized modelling and data processing infrastructure that would allow comprehensive environmental simulations, based on limited computer programming skills, computer power, and data availability. We are developing a software framework of model and data modules to enable various stakeholders to tap into the recent and ongoing advances in environmental modelling, and high-quality data available on the Internet. The proposed framework would allow managers and planners to run simulations of policy scenarios and utilize state-of-the-art algorithms to develop and evaluate policy alternatives.

The web-based modelling framework is based on the following components:

- A web-based domain-specific interface which facilitates the development, configuration, and execution of models applicable to region-specific watershed issues;
- A data-finder and transformer unique to the landscape modelling framework that lever-ages relevant Open GIS catalogue, RDF, and GRID resource discovery standards;
- A module composer that uses a module pool and guided composition of modules based on expert rules, which are either automatically acquired or input from human users, to guide the simulation-modelling process; and
- A semi-automatic model calibrator and verifier to deliver high quality simulation models.

The framework's core components, i.e. model composition, data finder, and geospatial data processing, serve the needs of a wide range of applications. The framework should be designed to be configurable for multiple domain specific user groups, enabling applications as diverse as agricultural forecast, real estate market analysis, transportation planning, etc.

Keywords: Modularity, interoperability, conflict resolution, environmental management

1. INTRODUCTION

Environmental management of natural resources has become an important responsibility of regional environmental agencies and planning departments around the country. Dealing with issues of natural resource management is quite challenging for five principal reasons:
(1) Public and private interests are in dispute, resulting in mutually exclusive alternatives;
(2) There is political pressure to make rapid and significant changes in public policy;
(3) Private and public stakes are high with substantial, often irreversible, costs and risks;
(4) The technical ecological and sociological facts are highly uncertain;
(5) Policy decisions have effects outside the scope of the problem.

Due to its complex, multi-disciplinary nature, and long-term effects on the integrity of both natural resources and socio-economic systems, environmental management has necessarily become a participatory exercise, requiring feedback from many stakeholders. This requires that all participants understand the complex interactions and processes in natural and social systems over space and time. Dispute resolution professionals have traditionally relied on the exchange of information at forums and tended to reduce the complexity of the problem to make it cognitively more suitable for resolution. The use of computational techniques was relegated to optimization algorithms that presented multiple criteria for maximizing benefits and minimizing costs in a matrix from which various options were ranked and presented to stakeholders. However, stakeholders themselves were not part of the epistemic process and hence were often quite sceptical of the decision matrices. The use of cyber-infrastructure in eliciting and processing information from dispute stakeholders can thus be both empowering as well as providing clarity through synthesis of complex data sets [Voinov, Costanza, 1999; van den Belt, 2005]. Environmental parameters that can be presented in spatial terms have also been shown to enhance the potential for dispute resolution [Kyem, 2005].

After several decades of intense development, environmental simulation modelling has established itself as a powerful paradigm for understanding and forecasting environmental processes. Computer simulation has become an invaluable method for environmental decision-support, and has spawned a wide variety of tools developed by EPA, US Army Corps of Engineers, USGS, universities, and private industry. Despite the advancement of such tools, resource managers still must overcome four major obstacles in order to routinely benefit from goal-driven environmental modelling:

- Each region requires unique spatial and temporal data. These data are often difficult to find and require much effort to acquire and pre-process, due to a wide variety of formats, different sampling procedures, instrumentation, semantic conventions, and varying quality.
- Extensive computer programming and environmental science expertise are required to select appropriate models, link them together, import necessary data, and execute scenarios. Regional agencies rarely possess the necessary computer programming skills.
- The complex behaviour of ecological and social systems is often sensitive to small perturbations, and requires large amounts of computing power to run several models in concert to simulate environmental dynamics in adequate detail.
- Different environmental stakeholders often employ conceptual schemas and simulation models which describe a narrow set of processes, are hardly compatible, while procedures for assimilating different modelling results in informed model-based consensus-building are either confusing or absent.

Instead of contracting difficult modelling tasks to external experts, we advocate direct involvement of stakeholders in the modelling exercise. To facilitate stakeholder participation there is a clear need for an advanced interactive simulation modelling system tuned to the needs of environmental dispute resolution and providing resource managers with easy access to data and modelling tools, expert model composition rules, previous model runs, and visualizations of model outputs. This system can facilitate an approach that we call participatory environmental management. The target users of the system form a large user community, found in local and state governments, regional environmental agencies and planning offices. These users provide critical information to citizens and decision makers, and are positioned between the citizen and modelling 'experts'. They typically do not have experience in computer programming or model development, but do have experience and training in the environmental management domain.

So far there are no integrated modelling systems specifically developed to support the stakeholder participatory modelling process. Two types of modelling tools have been used in this context. General-purpose modelling systems such as Stella [2007], Microsoft Excel, Extend [2007], or Simile [2007] are simple enough for the stakeholders themselves to handle. More complex modelling tools, such as the LMF [2007] or Cormas [2007], have been used in a participatory context, but require teams of researchers to operate and support throughout the process. Neither of these software tools has been developed for multi-stakeholder decision-making and dispute resolution in a participatory setting, and lack essential functionality to integrate existing more complex models or access online environmental data repositories. In addition, most existing modelling environments are proprietary and are not designed for extensive customization needed to support participatory modelling.

There is a clear need to integrate simulation modelling and data processing tools to empower citizens and decision makers with the ability to evaluate management tradeoffs in an accurate, cost-effective, transparent, and publicly accountable manner. Any dispute can be treated as a clash of different models. Stakeholders contributing to a dispute resolution exercise come to the table with their different models, qualitative and quantitative, of the system at stake. The dispute evolves because of the inconsistencies and controversies between the different models. We hypothesize that by harmonizing the models for use in a common framework, much of the conflict can be resolved. In a way participatory modelling is a mechanism of joint fact finding and understanding when data and knowledge are shared among stakeholders in attempts to build a common model. When the participants mutually educate each other about the models they use, and arrive at a shared model of a system there remains less reason for conflict and dispute.

2. THE SYSTEM

We envision the architecture of the system as follows. The central components of the system are the "simulation modules" and the "composition metadata and rules" governing how the modules can be composed into comprehensive environmental models and workflows. The composition metadata consists of rules for composition, and quality indices for individual modules as well as rules to derive quality indices when modules are composed and used. Module metadata and the model composition rules are organized as a database that users can browse, query and annotate using a web browser. In addition, the system contains a registry of previous simulation model runs representing "best practices" of model applications for different scenarios. The main task of the "module composition wizard" is to guide users through the assembly of simulation models from the available modules. In a typical session, users will pick modules from the module registry, which best address the issue in dispute. If there is a history of previous use of these modules they will come already linked to necessary other supporting data and modules. When this information is missing, users are guided to compose modules into a modelling chain and, using model composition rules, define how data are exchanged between the modules and between online data sources and the modules. This compilation, saved as an XML document, is then interpreted by the model execution pre-processor responsible for retrieving data from online sources and converting them into model inputs. The next step is orchestrating a simulation run over the retrieved data, and calibrating model parameters. The

"Module building assistant" augments the wizard and contains routines that find and suggest new modules, data, and model composition rules that are not yet part of the system. This will guide future modelling exercises and provide tools to enhance the framework. In many cases this will require additional effort to "wrap" models and data making them compliant with the rest of the system. In this way, each module will be accompanied by a detailed and expanding use history to help stakeholders learn from the experience of others and share their own experience. A hierarchy of user access will provide several tiers of functionality for different groups of stakeholders, depending on their role in the project.

The "Optimized simulation execution engine" is responsible to run the models with the available data ("Locally archived data"). The model runs are optimized with this engine to facilitate faster response in support of interactive modelling sessions. Finally, the "Module calibrator and validator" provides necessary quality information about the simulation modules available in the system. Once the model is tuned to the available data, the calibrated parameter values are recorded and annotated, and model output is stored, analyzed and annotated using the online mapping, charting and annotation components of the modelling system.

One of the main reasons that existing modelling tools are hard to use is that data preparation is a hard and cumbersome task. The objective of the Locally Archived Data is to supply data to the simulation modules in the module pool. Data come from various archives in various formats. The Open GIS Consortium (OGC) standards on data interoperability are being implemented in many public agencies, driven by the Federal Geospatial One-stop initiative [OGC, 2007]. Relevant OGC standards for data and geospatial services should be adopted and taxonomic and ontology-driven discovery mechanisms should be used based on model requirements [OGC, 2007a].

2.1 Web-based User Interface (the Guru)

This component will serve as the entry point for users to interact with the system. System functions will be delivered to the user through this user interface driver, which will also provide essential tutoring and guidance. WEB technology will be used to create the user interface. The tasks that a user can perform from this interface are as follow:

- Search for useful modules and models;
- Compose simulation and data modules to form task-specific models;
- Test models;
- Run the model to obtain output data (delivered in different formats, plots or data).

The concept of workspace will be used to allow users to localize their own modules and models. A particular use of a module in this space is to construct a data module that actually takes data from the users' private holding. This will be especially useful for testing watershed questions that require more detailed or specific data inputs. The workspace will also allow users to upload their data to be used for their own simulation tasks. However in most cases users will be encouraged to make their data simply available over the Internet. This will be sufficient for the system to access them seamlessly.

The Guru will also serve the following maintenance tasks: (1) Compose simulation and data modules and insert them back to the Module Pool; and (2) Add relevant metadata.

Finally, results delivery is part of the web-based user interface. An important task is data visualization. These visualization modules are also in the module pool (see below) for the user to pick and choose.

2.2 Module Composition Wizard (the Composer)

The wizard first takes the user through a question-and-answer session, which automatically generates a draft model from existing modules. If the resulting model is not uniquely determined, the system will present various options from which the user must select. The user is guided through this with a series of explanations for the presented options, based on the module quality indices and composition rules.

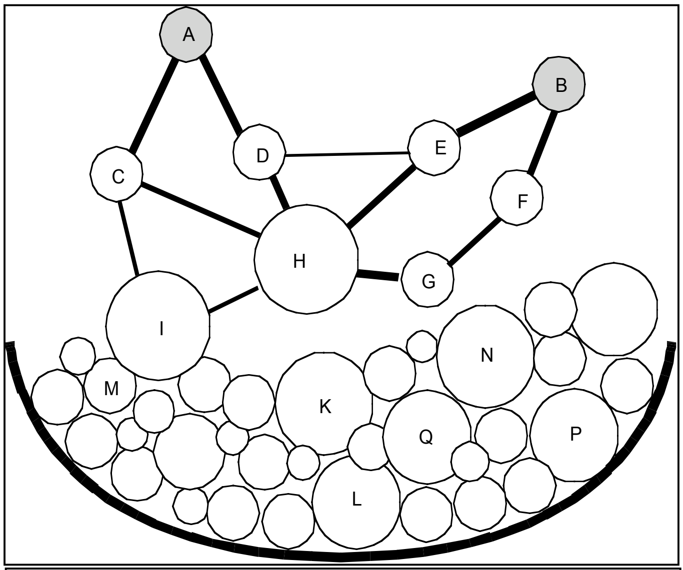

Figure 1: From the pool of available modules the user picks a few focal modules that respond to the issues at stake (say, A and B). These modules in turn are linked to other modules in the pool that provide input to the focal modules. The complexity of a module is indicated by the size of the circle, and the thickness of the link represents its reliability. There may be various minimal configurations that provide functionality to the model thus formulated (say, ADEB is the simplest model, whereas ADHEB is probably the most reliable one in terms of quality of links; ACHGFB may turn out to be the optimal one in terms of module quality). The users can choose the configuration that best suits their needs

The building block in our modelling approach is a module, which can be thought of as a function with inputs and outputs, and usually has a mathematical description. We allow different kinds of modules to co-exist in the module pool. For example, there might be modules described in XML-based Modular Modelling Language [Maxwell, Costanza, 1997], C++ code pieces, data sets, and Stella models. These simulation modules can be quite complex describing temporal or spatial dynamics of several variables, whereas others may be simple unit translators between output from one module and input to another. Some modules are "conduits" that "pipe" data sources from the Web, or in the local store, to other modules for a simulation task. Data modules are simple datasets that have only an output function. In short, the module pool contains simulation and data resources that are useful for watershed modelling, and can accept new modules as they become available.

In the simplest terms, a modelling task involves linking modules, connecting output from some modules to the input of others, and ultimately generating results that are useful for visualization and understanding of watershed processes. A general problem is how to describe the inputs and outputs of the modules so that only the appropriate connections are engaged. This, in general, is a very difficult task, akin to the code "reuse" problem in software engineering. However, we are not solving a general software engineering problem, since we are focused on a specific subset, that is, spatially explicit, coupled ecological, hydrologic, and socio-economic models. When we design a module for the module pool, we can determine which other modules will be connected (through input and/or output). A module design interface will be provided to help this connection task. This ensures that any future additions to the module pool are already linked to all the other possible sources/sinks of information and such that in future modelling tasks, the modules are used appropriately according to the domain knowledge inherent in such pre-built connections.

A conceptual view of the module pool is shown in Fig. 1. When a modelling task is started, the user identifies (through the composition wizard and ontology search tools) the few "focal" modules that generate the output essential to their task. Due to the pre-wiring, when these modules are "pulled" out of the pool, they "drag" out a whole vine of other modules that are connected to the focal modules. This pre-wiring defines all of the "possible" connections, including those that are critical for the focal module and those that are supplementary. It also determines the initial conditions, the forcing functions, and the boundaries of the system. Oftentimes, alternatives exist for many of the supplementary modules. For example, similar data may be available from different sources (represented by different data modules). If real data is limited, a simulation module might be used to generate approximate data. To make the right choice for a particular model goal the "composition wizard" will guide the modelling process through the choices between alternative modules. To be successful, the wizard needs to know the tradeoffs between different choices and be clear which modules are essential for analysis determined by the focal module.

Several indices can be helpful:

- Simulation/data quality indices, assessing the quality of the modules for various applications;
- Composition quality index, rating the criticality of particular linkages between modules;
- Model quality index, assessing the overall quality of a complete model.

The users can specify a number of criteria that are important for them, including performance, reliability, and sensitivity. Based on these, the final configuration of modules is determined, and the model is generated. A specific challenge in creating the Composer is the heterogeneity of data and modules. A study of metadata, matching of modules, and conversion module generation will be needed.

2.3 Module Calibrator and Validator (the Tester)

Quality indices of the modules are particularly important for the system. Calibration and validation of modules requires both guided and automated processes. An important research question is how to measure the quality of a module or model based on its previous performance.

We can distinguish between model testing or evaluation (assessing the degree and significance of fit of the model with data or with other models) and calibration (the adjustment of model parameters to improve the fit). Unlike statistical models, there are no universally accepted methods for testing or calibrating complex dynamic, non-linear simulation models [Berk, et al., 2000]. For the initial calibration and validation we can apply a range of visualization techniques and quantitative statistical tests informed by the specific context of the problem [Berk et al., 2001]. The human brain is a powerful pattern processor, and if model output can be presented in appropriate formats, direct visual comparisons of models with data can yield significant insights about model performance [Kuhnert et al., 2005]. In addition, complex models must generally be tested against a broad range of data of varying types, quality, and coverage. For example, for some variables we may have only scattered field measurements, while for others we may have more complete time series data or even maps, while others may have no quantitative data at all, but only qualitative assessments. As discussed above, a system is therefore necessary for ranking or grading the relative quality of the data and the relative importance of the variables to be fit.

The difficulty of calibrating complex models by manually adjusting parameters can be overwhelming, and an optimal parameter set may not even exist [Beven, 1993]. It is therefore more appropriate to recognize the uncertain nature of the process, to focus on finding "good" parameter sets, and to create a better way of defining the quality of parameter sets. However, once we have done the original model testing, we can automate some of the recalibration/validation functions to the Tester that together with the Feeder will keep track of all the newly available information and do a background recalibration of the modules.

There are web sites with high resolution and continuous monitoring data that will be used as test cases. A well watched site will be used to assess the impact of various levels of data availability on the calibration process and on the model results. A research question is how to "incrementally" calibrate modules. That is, instead of running models "from scratch" when new data become available, we try to continuously test the modules, as in continuous queries [Babu, Widom, 2001]. Once statistics of module testing are collected, we can use data mining techniques to find ordinary cases (outliers), and general trends in the model output [Han, Kamber, 2000]. All this is an important part of the ultimate goal of these system components, that is, to constantly evaluate and update the quality indices that are assigned to the modules and links in the module pool.

2.4 Optimized simulation execution engine (the Runner)

This part of the system is responsible to execute the simulation after the user has composed the model (with the help from the Composer and the Expert). In a participatory modelling environment, fast response is of paramount importance, as an execution of a model may be essential for further modification of the model. This is similar to OLAP systems (online analytical process systems) [Sarawagi, et al., 1998]. Since OLAP systems are mainly used in an exploratory fashion, fast response is the main concern. We plan to study techniques used in OLAP systems, such as pre-computation, for our purpose.

The second direction is to use model output from previous runs to accelerate the current model execution. This is realistic due to the fact that when models are constructed in an exploratory way (in a participatory environment), the previous model and the current model may share a lot of common modules. The intermediate results of the previous model execution may be used to expedite model executions. This is similar to using pre-computation to achieve faster response in database systems [Wiekum, 2002]. As in any pre-computation scheme, the critical issue is what computing results to store and how to find the useful results. In our simulation modelling system, to achieve such pre-computation, we can use a self-adaptive method to record computing results. That is, initially, when the storage space is available, we will store all pre-computation results of all intermediate computation (the modular approach of the models in our system makes this easy to implement). As the time goes by and we need to purge storage, we will retain those that have been used most recently while purging away the stored results that are not used recently. Another problem is how to search for available pre-computed results. We can associate the module metadata with pre-computation results. If the exact modules are used (including the same data modules), then the pre-computation result can be used.

4. DISCUSSION AND CONCLUSIONS

There are three important principles that the system is based on:

(1) The use of a module pool and guided composition of modules as a basis for simulation modelling. Modules come in a variety of time and space resolutions and scales. Matching these scales and ensuring consistency in the overall model is not a trivial process and may be difficult to fully automate. As one possible solution we can pre-link the existing modules making sure that they are compatible and can function in concert. Then the users will need to 'cut off' the links that are redundant, choosing the ones that suit their needs best of all. This can be done based on the reliability indices, or history of previous use, or personal preference.

(2) The use of expert rules, which are either automatically acquired or created by human users, to guide the simulation-modelling process. These are the various principles and rules that apply to module and data scale, reliability, compatibility, and relevance.

(3) The intelligent use-tracker, a system that registers previous instances of module uses and success or failure associated with that. This is basically the growing knowledge base of previous experience acquired from using different modules for various applications.

The success of a participatory modelling project depends upon the transparency and usefulness of the model that is created. The model and the modelling system used cannot be overly complex to bar users with low computer skills from accessing and using it. There should be several levels of access that would match the needs of various types of users, from the very general public type, that provides plug-and-play and scenario run functionality, to the sophisticated users who may need to extend the system with their own modules and expert rules.

Major steps in developing the system include the following:

- Develop a registry of commonly used and tested modules, and wrap the modules within a common model-coupling framework such as CCA [2007] or OpenMI [2007]. This will allow discovery and guided composition of dynamic, spatially explicit, models.
- In coordination with federal, state and local agency partners, develop or integrate available online data access and conversion services, to ensure that data retrieved from several common federal sources of environmental information are formatted as valid local model inputs (for observations data in particular).
- Create an online mapping interface providing browsing and query access to online metadata repository for the data typically required by the simulation models.
- Integrate the participatory modelling system in a portal environment, to support personalized storing, analyzing, documenting, and annotating models and simulation runs computed for specific locations and scenarios.

Most important, we need to build the community of users and developers who will collaborate on the project and contribute their modules, data sets, methods, applications, case histories, bugs and documentation. Building such a community requires openness and flexibility. One of the reasons that we start with this vision is because we invite input at this very early stage of project development and are prepared to modify the system based on this community input. Our hope is that if the community develops along with the project there are better chances that the product will suit the needs of the users.

REFERENCES

Babu, S. and Widom, J. 2001. Continuous queries over data streamsq. SIGMOD Record. 30(3): p. 109-120

Berk, R.A., Bickel, P., Campbell, K., Keller-McNulty, S., Kekky, E., and Sacks, J. (2000). Workshop on statistical approaches for the evaluation of complex computer models. Los Alamos National Laboratory. LA-UR-00-5034.

Berk, R.A., Fovell, R.G., Schoenberg, F., and R.E. Weiss (2001). The use of statistical tools for evaluating computer simulations - An editorial essay. Climatic Change. 51(119-130)

Beven, K. (1993). Prophecy, reality and uncertainty in distributed hydrological modelling. Advances in Water Resources. 16: p. 41-51

CCA (2007) - http://www.cca-forum.org/, www.llnl.gov/CASC/components/babel.html

Cormas (2007) - http://cormas.cirad.fr/indexeng.htm

Extend (2007). http://www.extendsim.com/index.html

Han, J. and Kamber, M. (2000). Data mining: Concepts and techniques. The Morgan Kaufmann Series in Data Management Systems, ed. Gray, Jim: Morgan Kaufmann Publishers. 500.

Kyem, Peter (2005). Of intractable conflicts and participatory GIS applications: The search for consenus amidst competing claims and institutional demands. Annals of the Association of American Geographers, 94:1:37-57.

Kuhnert, M., A. Voinov, R. Seppelt (2006). Comparing Raster Map Comparison Algorithms for Spatial Modeling and Analysis. Photogrammetric Engineering & Remote Sensing, Vol. 71, No. 8, August 2005, pp. 975–984.

LMF (2007) – Landscape Modeling Framework - http://www.uvm.edu/giee/IDEAS/lmf.html

Maxwell, T. and Costanza, R. 1997. A language for modular spatio-temporal simulation. Ecological Modelling. 103(2, 3): p. 105-113

OGC (2007a). Open Geospatial Consortium (OGC) Standards - http://www.opengeospatial.org/standards

OGC (2007). Open Geospatial Consortium Interoperability Program Policies and Procedures - http://www.opengeospatial.org/ogc/policies/ippp

OpenMI (2007) - http://www.openmi.org/openminew/

Sarawagi, S., Agrawal, R., and Megiddo, N. (1998). Discovery-driven exploration of OLAP data cubes. The 6th International Conference on Extending Database Technology: p. 168-182

Simile (2007) - http://www.simulistics.com/

Stella (2007). http://www.iseesystems.com/

van den Belt, Marjan (2005). Mediated Modeling: A systems dynamic approach to environmental consensus building. Washington DC; Island Press.

Voinov, A. and Costanza, R. (1999). Watershed management and the Web. Journal of Ecosystem Management. 56: p. 231-245

Wiekum, G., Monkeberg, A., Hasse, C., and Zabback, P. (2002). Self-tuning database technology and information services: From wishful thinking to viable engineering. Very Large Data Bases (VLDB) Conference. Hong Kong, China: p. 20-31