Web Based Simulation Environment for Water Resource Management

Antoine Cheron, Zengularity, France,
Quentin Courtois, Univ Rennes, CNRS, OSUR, France,
Johann Bourcier, Univ Rennes, Inria, CNRS, IRISA, France,
Benoit Combemale, Univ. Toulouse, France,
Jean-Raynald de Dreuzy, Univ Rennes, CNRS, OSUR, France

Key words: Model Simulation, Software Engineering, Domain-Specific Languages, Lightweight Containers, Web-Based Application

Challenges in Water Resource Management Simulation

Environmental issues on water resources and water quality are challenging multidisciplinary questions both for the scientific and stakeholders communities. Some issues like artificial aquifer recharge involve strong couplings and feedbacks between injection, chemical reactions and bacterial growth. These questions have been well identified by applied mathematicians and advances in coupling methods and software are continuously proposed to face increasingly complex situations.

Many other issues like catchment to regional water quality issues face other challenges of integrating contributions from different part of the catchments or from different disciplines like hydrology and biochemistry with only weak interactions. For example, water quality requires to integrate information on contaminant inputs, models of transfer and reactivity, calibration and predictive capacities at the scale relevant to the targeted sampling. Relations between data, models and prediction is especially challenging.

While black box models may increasingly provide immediate solutions for using the growing flow of observations produced by cheaper and more connected sensors, we propose to reach a balance between process-based models and quantity of data in a flexible and yet efficient development framework. We argue that progresses in software architecture can provide a shift of the hydrological development capacities to more readily accessible solutions for simulation, prediction and use of existing information.

On the Needs for Abstractions, Separation of Concerns and Socio-Technical Coordination

The complex problems addressed by computational sciences are more and more benefiting from the progress of computing facilities (simulators, libraries, accessible languages, etc.) [1]. Nevertheless, scientists, and other stakeholders wishing to interact with complex models, still face various difficulties that would benefit from advanced software engineering principles.

First, the execution of complex simulation processes requires to coordinate various simulation activities. In this context, **an explicit software architecture** enables the seamless integration of time consuming tasks, and provides the support to automate complex tasks such as calibration.

Moreover, complex simulations require complex deployment in order to finely tune and optimize the underlying execution platform, and eventually support efficient execution and possibly the seamless exploration of alternatives. In this context, domain-specific languages support the **separation of concerns** between the domain experts who use the abstractions for expressing complex situations in terms of the problem, and the software engineers who implement the associated generative approaches that bridge the gap to the solution space, automating the deployment that manage scalability and distribution over the accessible execution platform.

Finally, the study of complex problems is a collaborative effort involving many scientists worldwide, and requires the coordination of various artifact (models, data, etc.). It is thus of uttermost importance to provide an environment supporting the required **socio-technical coordination**, such as collaborative modeling, deployment into the cloud, and Web-based applications.

Applicability of Software Engineering Principles

Our goal is to provide a tool to ease the design, calibration and execution of scientific simulation processes for Water Resource Management. To deal with the problems mentioned in the previous section, our approach is based on the three following pillars.

A domain specific approach. In our approach, scientists design their simulation processes using specific concepts from the water resource management domain. We provide a domain specific language (DSL) to design complex simulation by describing a simulation process. A simulation process is composed of various simulation activities chosen among the following categories: *Data importation/acquisition, Data pre-processing, Simulation,*

Data post-processing, Visualization. Each activity has input and output data. The whole simulation consist in executing the process following the partial order of the activities.

Seamless deployment and scalability techniques. Our approach has been devised to ease the portability and deployment process of the whole system. We therefore rely on a set of container images which can be easily deployed on demand. The back-end technology used to perform each simulation activity allows parallel execution of different activities. This property combined with the ability to easily deploy on demand our system enable seamless scalability of the whole system.

A collaborative and integrated approach. In our approach, scientists are provided with a unique tool to perform all simulations activities. This tool is available on-line in the form of a web application which provides both the ability to collaborate on-line between researchers and the ability to perform all simulation activities on a single tool. Therefore the tool provides a graphical way to design the simulation process and integrated code editors to specify the behavior of each simulation activity. The collaboration between several scientists is handled by the integration of a collaborative asynchronous protocol to synchronize the server and all connected clients.

Towards a Web-Based Domain-Specific Simulation Environment

We have built a prototype implementing the approach defined in the previous section. Our prototype is accessible through a web based interface enabling the design, manual calibration and execution of various water resource management simulations. Our prototype is open-source[1] and provides the implementation of the server and client parts together with all required synchronization protocols. The server part has been implemented using docker[2] containers to ease the deployment process. Our current implementation heavily relies on Jupyter[3] to perform the simulation.

Our server implementation is currently limited to simulation activities written in Python, but the support for other languages can be easily added. Moreover, the automatic calibration is not yet implemented in our prototype, but the proposed architecture and the current implementation have been designed to easily incorporate this new feature, and test various calibration techniques. In addition, the automatic scaling to take benefit from a pool of computers is not yet implemented, but the docker container approach allows us to benefit from the docker ecosystem to deal with this problem.

The prototype (see figure 1) has been tested and used by hydro-geologists and their feedback is really positive. They have been able to install and use our tool to design and execute a simulation.

Conclusion and perspectives

We present a web-based domain-specific environment for designing simulation processes, editing simulation codes, and automatically deploying them over distributed containers.

Our first experiments are very promising, and open many perspectives. In particular, we plan to investigate the way to support automatic calibration from the explicit definition of the simulation process, to increase the possible scalability using a deployment on top of Docker Swarm, and to rely on existing collaborative workspace management to support fine-grained model sharing and IP management. Finally, we also look at other application domains from computational sciences that water management, where such an approach would be instantiated to provide domain-specific environment for scientific simulation.

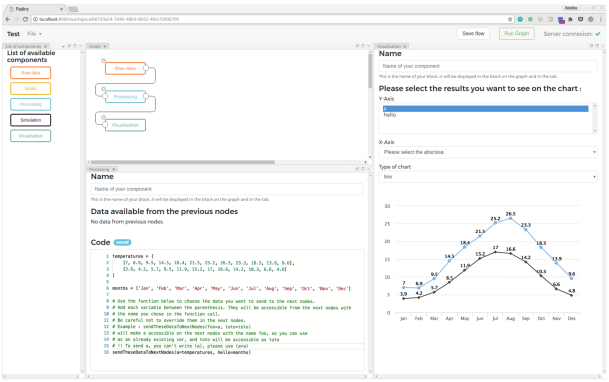

Figure 1: A Screenshot of our web based Domain-specific Simulation environment

References

[1] Jean-Michel Bruel, Benoit Combemale, Ileana Ober, and Hélène Raynal. MDE in Practice for Computational Science. In *International Conference on Computational Science (ICCS 2015)*, Reykjavík, Iceland, June 2015.

[1] Source code available at `https://github.com/AntoineCheron/padins-backend`
[2] Cf. `https://www.docker.com`
[3] Cf. `http://jupyter.org`