Geophysical Research Abstracts
Vol. 13, EGU2011-9753, 2011
EGU General Assembly 2011
© Author(s) 2011

Constructing integrated models: a scheduler to execute coupled components

Oliver Schmitz (1,2), Derek Karssenberg (1), Kor de Jong (1), and Jean-Luc de Kok (2)

(1) Department of Physical Geography, Utrecht University, The Netherlands, (2) Flemish Institute for Technological Research (VITO), Belgium ({O.Schmitz|D.Karssenberg|K.deJong}@geo.uu.nl, Jean-Luc.DeKok@vito.be)

In integrated modelling, multiple interacting components of various domains from environmental, social and economic systems are assembled to model complex system behaviour. The integrated model thereby serves as a means to increase the scientific understanding as well as a base for policy relevant research. Tools for integrated modelling need to support construction and coupling of model components, and preferably assistance to assess models. However, present software frameworks are either tailored to component construction or to component coupling, whereas a consolidated environment is desired. Existing frameworks also require profound knowledge in system programming languages, and offer limited generic support for model assessment. As a result, emphasis in the model development is currently on construction rather than on testing and assessment.

To resolve the addressed technical hassle in the model construction process, we propose a layered software architecture for straightforward construction, coupling, execution and analysis of model components. Here, we primarily focus on the treatment of temporal interactions between components. We strive to support different modelling scenarios such as a combination of components holding continuous discrete time steps, for example in field-based modelling, and event-based components used in individual based modelling.

The layered architecture of the software framework is designed as follows: in the first layer, the model description layer, a model developer constructs components with the help of spatial and temporal building blocks, and describes the component relationships. By following the framework guidelines, components can consist of fixed or variable time step lengths, or confined lifetime within the model run. Also, adapter that aggregate over time can be specified. These adapter can be used for example to average output of a component with a daily time step in order to feed a component with a monthly time step. In the second, the schedule generation layer, the information of components and their relationships are processed to determine the order of component execution. The third, the execution layer, organises the actuating of components and adapter as well as the accounting of component results for post-run analysis purposes.

We provide the functionality of the framework as modules for the high-level scripting language Python. Therefore, the model builder is able to use a platform-independent development environment with a strong support for scientific computing. Moreover, non-software developers are able to conduct exploratory model construction and analysis after a short familiarisation phase.

We present a prototype version of the software framework in connection with spatio-temporal operations from the PCRaster library. We demonstrate how a model developer can apply the framework to construct complex models with diverse spatial and temporal discretisations. A case study is used to illustrate the interaction of components with different time step lengths and process representations.