

Accepted Manuscript

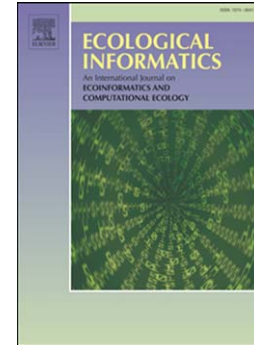
A software framework for process flow execution of stochastic multi-scale integrated models

Oliver Schmitz, Jean-Luc de Kok, Derek Karssenber

PII: S1574-9541(16)30004-8
DOI: doi: [10.1016/j.ecoinf.2016.01.009](https://doi.org/10.1016/j.ecoinf.2016.01.009)
Reference: ECOINF 658

To appear in: *Ecological Informatics*

Received date: 12 October 2015
Revised date: 25 January 2016
Accepted date: 27 January 2016



Please cite this article as: Schmitz, Oliver, de Kok, Jean-Luc, Karssenber, Derek, A software framework for process flow execution of stochastic multi-scale integrated models, *Ecological Informatics* (2016), doi: [10.1016/j.ecoinf.2016.01.009](https://doi.org/10.1016/j.ecoinf.2016.01.009)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

A software framework for process flow execution of stochastic multi-scale integrated models

Oliver Schmitz^{a,b,*}, Jean-Luc de Kok^b, Derek Karssen^a

^a*Department of Physical Geography, Faculty of Geosciences, Utrecht University,
Heidelberglaan 2, PO Box 80115, 3508 TC Utrecht, The Netherlands*

^b*Flemish Institute for Technological Research (VITO), Unit Environmental Modelling,
Boeretang 200, 2400 Mol, Belgium*

Abstract

Dynamic environmental models use a state transition function, external inputs and parameters to simulate the change of real-world processes over time. Modellers specify the state transition function and the external inputs required in the process calculation of each time step in a component model, a self-contained numerical module representing an individual spatio-temporal process. Depending on the application case of a component model—such as standalone execution or in an integrated model—the source of the external input needs to be specified. The required external inputs can thereby be obtained by a file operation in case of a standalone execution. Alternatively, required inputs can be obtained from other component models, in case the component model is part of an integrated model. Using different notations to specify these input requirements, however, requires a modification of the state transition function per application case and therefore would reduce the generic applicability of a component model.

To address this problem, we propose the function object notation as a means to specify the input requirements of a component model. This function object notation provides modellers with a uniform syntax to express the input requirements within the state transition function. During component initialisation, the function objects can be parameterised with different external sources. In

*Corresponding author. Tel.: +31 30 253 9363; fax: +31 30 253 1145. E-mail address: o.schmitz@uu.nl

Download English Version:

<https://daneshyari.com/en/article/6295848>

Download Persian Version:

<https://daneshyari.com/article/6295848>

[Daneshyari.com](https://daneshyari.com)