International Environmental Modelling and Software Society (iEMSs)
2012 International Congress on Environmental Modelling and Software
Managing Resources of a Limited Planet, Sixth Biennial Meeting, Leipzig, Germany
R. Seppelt, A.A. Voinov, S. Lange, D. Bankamp (Eds.)
http://www.iemss.org/society/index.php/iemss-2012-proceedings

Tools for uncertainty propagation in the Model Web using Monte Carlo simulation

Benjamin Pross[a], Lydia Gerharz[a], Christoph Stasch[a], Edzer Pebesma[a]
[a]*Institute for Geoinformatics, University of Münster, Weseler Straße 253, 48151 Münster, Germany ({benjamin.pross, lydia.gerharz, staschc, edzer.pebesma}@uni-muenster.de)*

Abstract: In this paper, we describe an approach for integrating Monte Carlo simulation in the Model Web to propagate uncertainty in model inputs and processes. In our approach, the models and model services are not capable to handle uncertainties. Therefore, we developed separate Web service components that can be used to manage uncertainties in the model workflow using Monte Carlo simulation. This allows flexible application of the developed uncertainty services with existing model services to quantify uncertainties propagated from the model inputs to the outputs. The approach is evaluated in an air quality modelling scenario where AUSTAL2000, a local air quality prediction model, is connected to the Model Web and uncertainty-enabled with the tools presented.

***Keywords**:* Uncertainty propagation; Environmental model; Web service; Monte Carlo simulation

1 INTRODUCTION

Environmental models can be complex and are usually not designed in a web-accessible way from the beginning. The Model Web envisions an infrastructure for standardised access to models and their results (Geller [2008]). This infrastructure should enable the coupling of models and data conveniently and thus enhance the reproducibility and transparency of scientific work. This poses numerous conceptual and practical challenges to permit web-enablement of environmental models. Key issues are uncertainty and trust in model results as well as processes that are available to heterogeneous users in the Web. Environmental models always include modelling errors and simplifications (Chang and Hanna [2004]) Therefore, when chaining (model-) services of varying or unknown quality, the uncertainty in the outputs needs to be propagated to allow rational decision making based on the final product. The European FP7 research project UncertWeb aims to develop and apply tools and methods to create an uncertainty enabled Model Web (Pebesma [2010]). Different application scenarios of the project raise different challenges and requirements in terms of security, performance, and tools and technologies required for uncertainty propagation. In this paper we will present two concepts to manage uncertainties in model inputs and processes propagated to the output by using Monte Carlo simulation (Metropolis [1949]). We realised these concepts as two Web services handling uncertainty for environmental model workflows in the Web.

1.1 Background

The OGC web processing service (WPS) specification (Schut [2007]) defines a standard for Web based (geo-) processing. The communication between clients and server is largely based on XML. Also, the metadata (i) of the whole Web service (e.g. version, service provider, list of processes) and (ii) of a single process (model) provided by the service (e.g. identifier, input and output types) is described

in XML. The type of models as well as the input and output types are not restricted in any way. That makes it difficult to develop clients that are able to execute processes on different WPS. Therefore profiles should be defined, restricting input and output types of a process to optimise interoperability (Baranski [2009]).
The Uncertainty Markup Language (UncertML) is used to encode probabilistic uncertainties in the Web. After UncertML 1.0 (Williams [2009]) was defined by an XML schema in its first version within the INTAMAP project (Pebesma [2011]), UncertML has recently been updated to version 2.0 by developing a conceptual model for uncertainties and defining additional encodings such as XML or JSON[1].
The Observations & Measurements (O&M) format (Cox [2007]) defines an interoperable model and encoding for spatio-temporal vector-based observations in the Web. To integrate uncertainty in the O&M format, the UncertWeb O&M (U-O&M) profile has been defined that reduces the complexity of the core standard and defines two ways to add uncertainty in observations.

1.2 Related Work

For the coupling of model and model services, numerous approaches exist. Jagers [2010] elaborated on the differences of a number of them. Two widely adopted workflow engines are Taverna (Hull [2006]) and Kepler (Altintas et al. [2004]). With Taverna Server the Taverna system provides Web access to model workflows and Kepler workflows can be exposed as web processing services (Pratt et al. [2010]). Uncertainty propagation, however, is not addressed in a standardised way by the workflow tools. The OpenMI standard (Greggsen et al. [2007]) enables coupling of models in complex workflows and executing them accordingly. It also provides support for legacy models as well as Web services. Handling the uncertainty in the model workflows is also not a core feature of the standard.
The remainder of the paper is structured as follows: First we introduce the concepts of uncertainty propagation in the Model Web. Then the two Web services for uncertainty propagation are described (section 2). Afterwards we evaluated these services in an air quality modelling case study which is described in section 3. In section 4 we will present the results of our work and end in section 5 with a conclusion and outlook.

2 UNCERTAINTY PROPAGATION IN THE MODEL WEB

Basically there exist two approaches to quantify model output uncertainty (Bastin et al. 2012). In the first approach the model code itself is modified to enable uncertainty propagation. Although this is a reasonable approach, there are a couple of issues: Especially, one needs access to and deep knowledge of the source code of the model. The second approach is to wrap the unmodified model with a system that manages the uncertainty propagation. We follow the latter approach and develop a concept that utilises Monte Carlo simulation. Figure 1 gives an overview of the architecture. In our approach the model has to be wrapped in a Web service (model service). This service merely enables the sending/receiving of the standard model inputs/outputs via the web. In front of the model service we place another Web service, the Monte Carlo service. This service accepts uncertain inputs and is able to execute Monte Carlo simulations. The third service is called (uncertainty) transformation service. It is able to transform different uncertainty representations (e.g. Normal Distribution, Samples) between each other. A detailed description of the architecture and the involved services is given in the following sections.

[1] The Java Script Object Notation is an encoding format that allows encoding JavaScript objects in a common way.

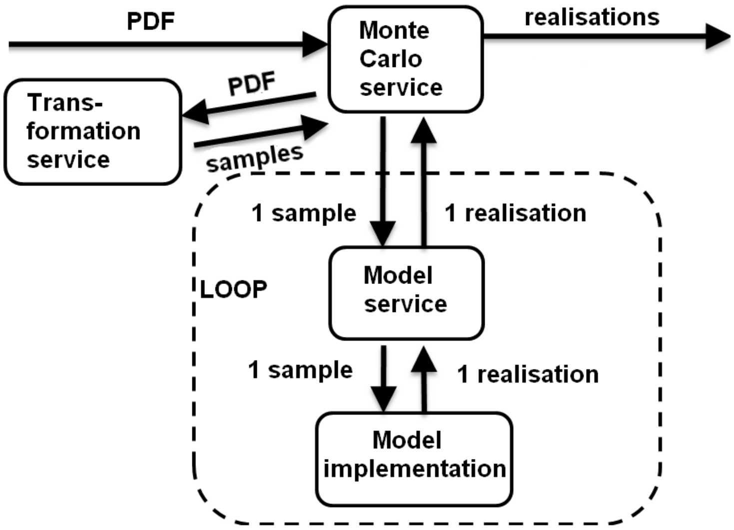

Figure 1. Uncertainty propagation architecture overview.

2.1 General Execution Patterns

There are two general patterns for the Monte Carlo execution: In the first one (the simple workflow - see figure 2) a set of samples serves as the uncertain process inputs for the process. These inputs have to be marked with the prefix “u_”. The uncertainty type (here: sample) of the uncertain inputs is resolved by the Monte Carlo service and treated accordingly. In this case the model service will loop over the samples and will create requests containing the sample input as well as the static model inputs. The request is sent to the specified model service. Once the simulation is finished, the resulting output samples are put together and sent back to the user.

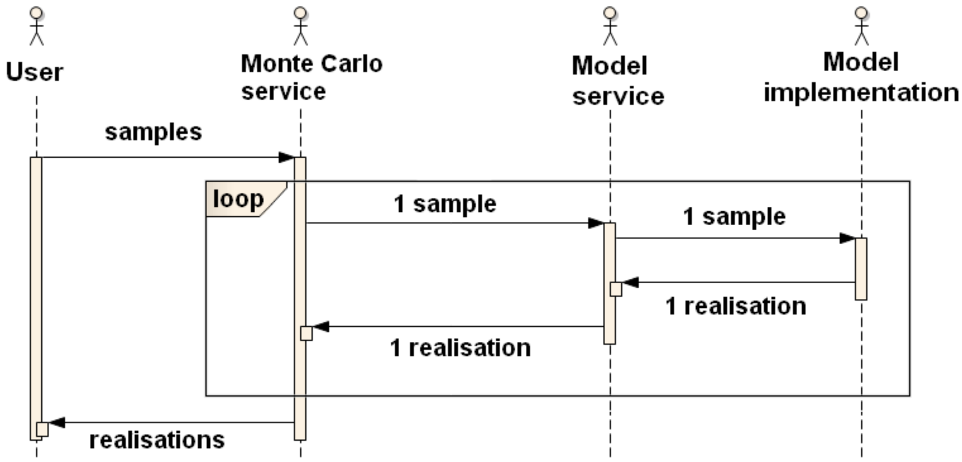

Figure 2. Simple Monte Carlo simulation workflow.

In the second pattern (the extended workflow – see figure 3) the uncertain input is encoded as a probability distribution function (PDF). In an additional step during the process, the PDF is transformed into a set of samples using the transformation service (see Section 3). The Monte Carlo service will iterate over these samples as in the simple workflow pattern. The resulting output samples are transformed into a PDF via the transformation service again and sent back to the user.

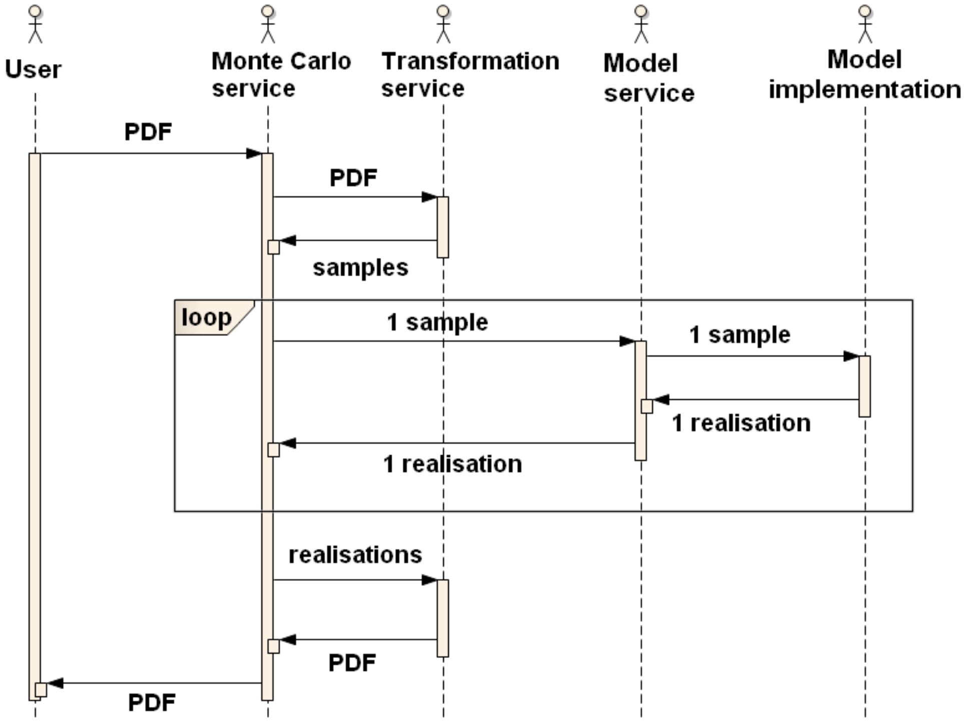

Figure 3. Extended Monte Carlo simulation workflow.

The application of these general patterns (i.e. the internal workflow of the Monte Carlo service) depends on the model service request. An example for the extended workflow is given in the case study described in section 4.

2.2 Uncertainty Transformation Service

The transformation service allows the conversion between different uncertainty types. These can be simple transformations from distributions to samples or from samples to statistics like the mean and standard deviation. Additional the service allows conversions taking into account further assumptions, for example going from a set of realisations to a distribution. The service creates spatial and temporal realisations. It is able to parse and generate UncertML, NetCDF-U (Bigagli and Nativi [2011]) and U-O&M, and it links to R[2] or Matlab[3] for the conversion of uncertainty types.

2.2.1 Implementation

The transformation service is realised as a profile of the WPS based on the 52°North implementation[4]. It uses Rserve[5] to link to R where the actual conversion is performed. In Table 1 currently available processes are described.

Table 1. Processes offered by the transformation service.

Process	Description	Supported input types	Supported output types

[2] http://www.r-project.org/
[3] http://www.mathworks.de/products/matlab/
[4] http:// 52north.org/wps
[5] http://rosuda.org/Rserve/

Gaussian2Samples	Creates random samples from a normal distribution.	NetCDF-U, U-O&M, UncertML	NetCDF-U, U-O&M, UncertML
MultivariateGaussian2Sample	Creates random samples from a multivariate normal distribution.	U-O&M, UncertML	U-O&M,UncertML
Lognormal2Sample	Creates random samples from a lognormal distribution.	NetCDF-U, U-O&M, UncertML	NetCDF-U, U-O&M, UncertML
Samples2Statistics	Calculates statistics for a set of samples	NetCDF-U, U-O&M, UncertML	NetCDF-U, U-O&M, UncertML

In the following we describe the workflow of the process "org.uncertweb.wps.Gaussian2Samples" The transformation service parses the input UncertML file and passes the mean and variance values via Rserve to R where samples are drawn from the Gaussian distribution. The resulting sample values are gathered and written to a new UncertML or NetCDF-U file and returned.

2.3 Monte Carlo Service

The Monte Carlo service serves as a proxy for a Web service that encapsulates a model. The model is not able to use uncertainties, e.g. distributions, in its input information. Thus, the Monte Carlo service is managing the uncertainty propagation using Monte Carlo simulation. Thereby, a large number of samples are drawn from the input distributions and each sample is used for a single model run. The outputs can be summarised again as either a distribution or as statistics.
The identifier of the processes in the service offers follow the specific naming scheme "org.uncertweb.ups.UPSModelProcess", where "Model" should be replaced by the name of the model. The processes have a well-defined set of inputs and outputs. In the following table (Table 2) the input and output types are described:

Table 2. Input and output descriptions of a Monte Carlo simulation process.

Identifier	Description
IdentifierSimulatedProcess	The identifier of the process with which the Monte Carlo simulation will be run.
u_InputName	Inputs for the process that will change with every Monte Carlo run. Encoded for example in UncertML or O&M.
InputName	Inputs for the process that will not change during the Monte Carlo simulation. These inputs are passed unmodified to the model WPS.
ServiceURL	The url of the WPS that exposes the model process.
OutputUncertaintyType	Besides the output format we need to specify the output uncertainty type. For example, samples or normal distribution.
NumberOfRealisations	The desired number of samples, i.e. Monte Carlo runs.
UncertainProcessOutputs	The result with uncertainty encoded in the specified format. For example UncertML or U-O&M.

2.3.1 Implementation

Similar to the transformation service, the Monte Carlo service was implemented as a profile of the WPS based on the 52°North implementation. As described in the previous section, the processes the Monte Carlo service offers will differ from model to model. Therefore we implemented a template process including convenience methods e.g. to handle the uncertain inputs and to gather the results from the model service.

3 CASE STUDY

In this case study, the developed tools are tested in an air quality model scenario. Therefore, a dispersion air quality model was (i) brought to the web as a model service and was (ii) uncertainty-enabled using the developed services. A simple chain of necessary components was implemented to test the practicality of the components within a "real-world" scenario.

3.1 Scenario

For this case study, the PM_{10} (particulate matter with aerodynamic diameter smaller 10 µm) concentration produced by local urban emissions, such as traffic, is modelled using the dispersion model AUSTAL2000[6] (Janicke [1983]). AUSTAL2000 takes emission strengths of local sources, meteorology and land use as inputs to estimate the dispersion of particles over the study area on an hourly resolution. Input uncertainties are quantified for street traffic emissions and for wind speed and direction using sample measurements. As the model itself is deterministic and does not account for model error, the quantified uncertainties in the output will be caused by propagation from uncertain input. Using the Monte Carlo and transformation service, we will be able to model PM_{10} concentrations with uncertainties for point locations, e.g. tracked via the Global Positioning System (GPS).

3.2 Architecture

The air quality use case architecture basically follows the general architecture described in section 2. The AUSTAL2000 model software is accessed through a service interface which takes emission and meteorology inputs (without uncertainties), converts them to the AUSTAL2000 specific format, executes AUSTAL2000 and parses the output into O&M samples at the GPS point locations. The Monte Carlo service takes the uncertain inputs (emissions and meteorology) as distributions encoded in U-O&M from a client. The inputs are sent to the transformation service to draw samples which are used to execute the air quality model in a loop. Due to the model run time, only a small set of samples is executed to yield a set of output samples. The Monte Carlo service collects the samples and combines them as an U-O&M output.

3.2.1 Model Service

The AUSTAL2000 model process is deployed in the Web through the WPS OGC interface. The WPS implementation offers the "Austal2000algorithm" process to produce the required input files from the service inputs and execute the AUSTAL2000 software. Mandatory inputs are study area parameters, meteorological data, emission data and the receptor points, i.e. the locations that the PM_{10} concentration should be estimated for. After running AUSTAL2000, the service parses the results for the receptor points and sends them back as O&M encoded observations.

3.2.2 Transformation Service

The transformation service requires two processes for this use case: MultivariateGaussian2Samples and Gaussian2Samples. As no spatial sampling is required, the service gets the U-O&M documents containing the distribution

[6] http://austal2000.de/en/home.html

parameter and uses simple sampling methods in R based on the distribution parameters for each feature and time step separately.

3.2.3 Monte Carlo Service

The Monte Carlo service offers an "org.uncertweb.ups.UPSAustal2000Process" process that takes the static and uncertain process inputs as well as the service URL for the Model service. The service calls the transformation service for sampling of the meteorology and street emission inputs. With the samples, the Model service is executed for the number of realisations requested.

4 RESULTS

The architecture was successfully applied to model a number of point locations in the city of Münster. Results for 10 samples are shown in figure 4. The considerable long run time of the AUSTAL2000 model makes an execution in a Monte Carlo simulation resource intensive. However, the results show large differences in the model outputs for some points and time steps proving that the inclusion of input uncertainties in the model workflow is useful.
The execution in the Web allowed a seamless integration with other Web-based tools such as a visualisation client for uncertainty visualisation used to produce figure 4.

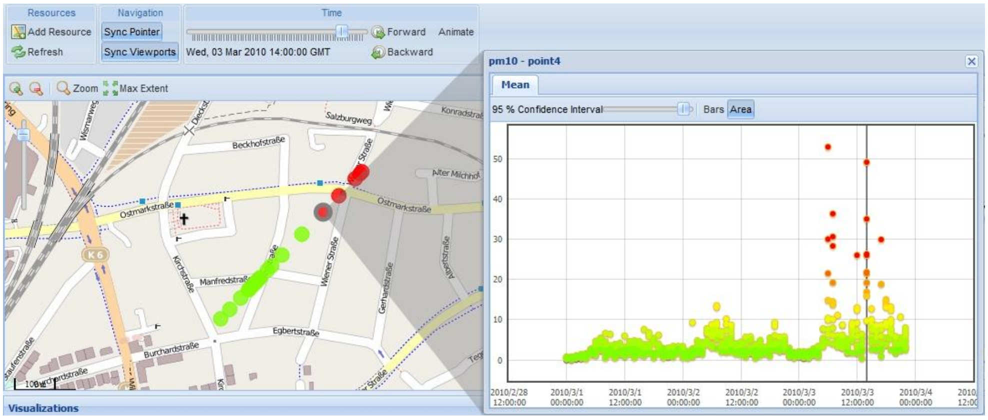

Figure 4. Example results for hourly PM_{10} concentration modelled by AUSTAL2000 at point locations in Münster. On the left side a map with mean concentrations per point is shown, on the right side the sample values for the whole time series at one point are visualised.

5 CONCLUSIONS AND OUTLOOK

This paper introduces tools for Monte Carlo simulation in the Model Web. Standardised Web service interfaces have been developed for software which was previously only accessible on a local machine, and could only understand proprietary data types. Both components in the current workflow have been uncertainty enabled, with uncertainty propagated from the first model output to the second model input.
Using the Monte Carlo service as a proxy service enabled us to reuse an existing implementation of the AUSTAL2000 model service without further implementation work necessary. All uncertainties are processed by the Monte Carlo service and hidden from the model service. The created workflow was successfully used to model air quality for point locations including input uncertainties.
As an advantage, this concept also allows the parallelisation of model runs as the Monte Carlo service manages the loop. Thereby, the service can distribute the different input samples which are then run independently on different model service implementations. This has a special advantage as environmental models might

have a considerable running time (in this case several minutes) which makes a parallel execution more appealing.

ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no [248488], UncertWeb.

REFERENCES

Altintas, I., C. Berkley, E. Jaeger, M. Jones, B. Ludascher and S. Mock. Kepler: an extensible system for design and execution of scientific workflows. Scientific and Statistical Database Management. Proceedings. 16th International Conference on. 423-424pp 2004.

Baranski, B. (Ed.) OWS-6 WPS Grid Processing Profile Engineering Report. OGC 09-041r3, Open Geospatial Consortium, Inc. 2009.

Bastin, L., D. Cornford, R. Jones, G.B.M. Heuvelink, E. Pebesma, C. Stasch, S. Nativi, P. Mazzetti and M. Williams. Managing Uncertainty in Integrated Environmental Modelling: The UncertWeb framework. *Environmental Modelling and Software* (in press) 2012.

Berre, A. and D. Roman. Environmental Service Infrastructure with Ontologies (ENVISION), In Towards a Service-Based Internet, Vol. 648, pp. 211-212 2010.

Bigagli, L. and S. Nativi. “NetCDF Uncertainty Conventions (NetCDF-U)”. OGC Discussion Paper 2012.

Chang, J. C., Hanna, S. R. Air quality model performance evaluation. *Meteorology and Atmospheric Physics,* 87, 167-196pp 2004.

Cox, S. (Ed.). Observation and Measurements - Part 1 - Observation Schema. OGC 07-022r1. Open Geospatial Consortium, Inc., 86 pp. 2007.

Geller, G.N. and F. Melton. Looking Forward: Applying an Ecological Model Web to assess impacts of climate change. *Biodiversity* 9, 79-83 2008.

Gregersen, J.B., P.J.A. Gijsbers and S.J.P. Westen. OpenMI: open modelling interface. *Journal of Hydroinformatics*, 9, 175–191pp 2007.

Jagers, H.R.A.. Linking data, models and tools: an overview. iEMSs 2010. Proceedings of The Fifth Biennial Conference Of The International Environmental Modelling And Software Society 2010.

Janicke, L.: Particle simulation of inhomogeneous turbulent diffusion. In: Weber (Ed.) Air Pollution Modeling and its Application II. New York: Plenum Press, 527-535pp 1983.

Hull, D., K. Wolstencroft, R. Stevens, C. Goble, M. Pocock, P. Li, and T. Oinn. "Taverna: a tool for building and running workflows of services.,"*Nucleic Acids Research*, vol. 34, iss. Web Server issue, 729-732pp 2006.

Metropolis, N., S. Ulam. "The Monte Carlo Method". *Journal of the American Statistical Association,* 44 (247), 335–341pp 1949.

Pebesma, E., D. Cornford, S. Nativi, C. Stasch. The uncertainty enabled model web(UncertWeb) In Proceedings of the Workshop "Environmental Information Systems and Services - Infrastructures and Platforms" (envip'2010), Bonn, Germany, October 6-8 2010.

Pebesma, E., D. Cornford, G. Dubois, G.B. Heuvelink, D. Hristopulos, J. Pilz, U. Stöhlker, G. Morin, J.O. Skien. INTAMAP: The design and implementation of an interoperable automated interpolation web service. *Computers & Geosciences,* 37, 343-352pp 2011.

Pratt, A., C. Peters, S. Guru, B. Lee and A. Terhorst. Exposing the Kepler Scientific Workflow System as an OGC Web Processing Service, iEMSs 2010. Proceedings of The Fifth Biennial Conference Of The International Environmental Modelling And Software Society. 1554-1561pp 2010.

Schut, P. OpenGIS® Web Processing Service: OGC 05-007r7, Open Geospatial Consortium, Inc. 2007.

Williams, M.; Cornford, D.; Bastin, L., Pebesma, E. (Ed.) Uncertainty Markup Language. OGC 08-122r2. Open Geospatial Consortium, Inc., 61pp. 2009.