

Opportunities for Software Engineering practices in deploying environmental models to cloud computing architectures

F.Samreen, W.Simm, R.Bassett, G.Blair, Ensemble Team, Lancaster University, Lancaster, UK,
P.Young, Lancaster Environment Centre, Lancaster University, Lancaster, UK,

Key words: Cloud Computing, DSL, Environmental Modelling, Model Driven Engineering, Software Frameworks

1 Introduction

Environmental science involves understanding complex systems and modelling their behaviour using computational models. Developing and executing environmental models requires the use of a large number of computational resources, along with the software tools and frameworks to provide supplementary support. Running such models not only requires the understanding of scientific phenomena but also sufficient experience of using operating systems, supporting software and knowledge of computer architectures.

A large number of complex computational applications are taking advantage of the potentially transformative characteristics of cloud computing such as on-demand, scaleable, elastic and heterogeneous services. Environmental modelling, with the exception of a few large community efforts, however has not yet taken advantage of cloud computing or the associated emergent software engineering tools and techniques. This is down to a number of factors including a lack of awareness of the potential of cloud computing, a lack of skills in exploiting cloud facilities and also a lack of support in terms of tools and frameworks to support cloud deployment. In addition, the inherent complexity and ensemble nature of environmental models means deploying them to a cloud infrastructure is not a trivial task. We have identified opportunities from software engineering to simplify this process and abstract away from the underlying architecture complexities.

We believe software engineering practices, emergent software engineering tools and cloud computing architecture could have a transformative effect on environmental modelling. Such a paradigm shift can allow scientists to **a)** spend more of their time concentrating on science rather than systems administration and **b)** run models more times and orchestrate new ensembles to better understand sensitivities and uncertainties.

2 Models in the Cloud

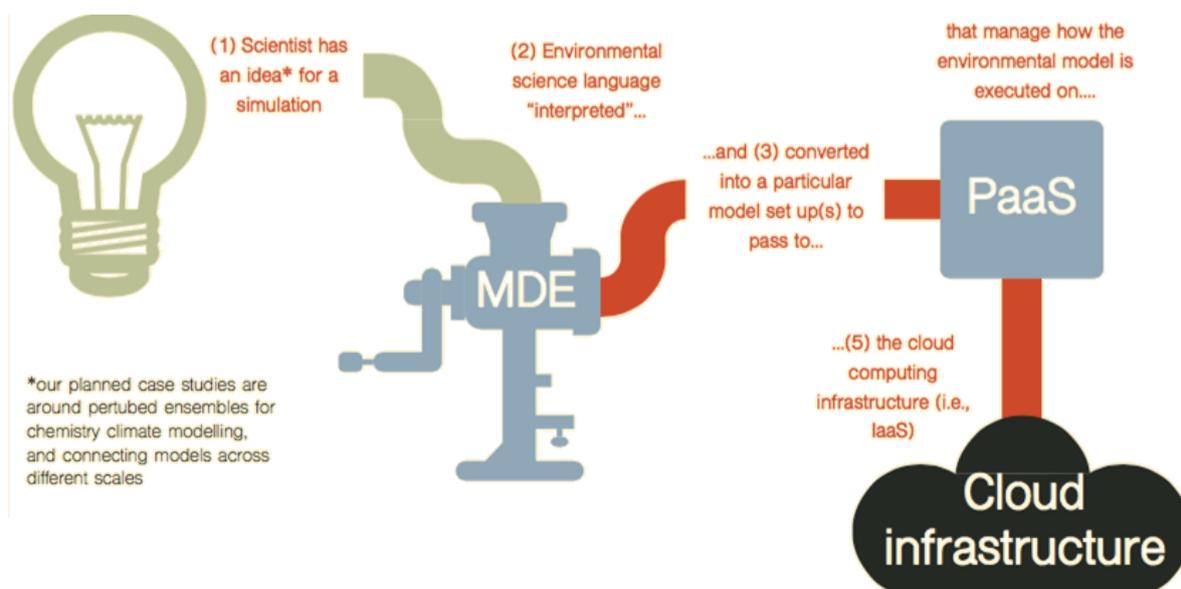


Figure 1: Models in the Cloud

We aim to address the above challenges through the "Models in the Cloud" project by using a Model Driven Engineering (MDE) approach to create abstract representations of the knowledge and activities related to a particular domain. The MDE approach promotes communication between systems by maximising compatibility, resulting in enhancing the automation of program development and re-usability of standard models/methods.

More precisely, we aim to develop software frameworks and domain-specific languages (DSLs) to support the execution of a range of environmental models and ensembles in the cloud.

The key to software frameworks is to offer software packages that support common functionalities but allowing them to be specialised, hence supporting separation of the scientific tasks from the computation. A DSL is a less comprehensive and more expressive specialised language designed for a particular application domain to help explaining complex and lengthy design process in an easy and simplified way. This can allow scientists to describe experiments in their language, and the frameworks to manage the deployment of environmental models to better architectures. Moreover, it can facilitate access to underlying, potentially complex data stores and other suits of tools, including data analytics. The process is represented by Figure 1 where (1) the scientist describes an experiment in a DSL close to the science; (2) this is interpreted and the model configured (3); (4) the platform tools control the deployment on appropriate computing architectures (5).

We intend to design and develop two DSLs with a user group of environmental scientists fully engaged in an agile, iterative development process.

1. **DSL for deployment of models:** to define the configuration for deployment infrastructure which can target different deployment strategies such as stand alone, cluster, HPC or cloud architecture. This will be generalisable to many types of environmental model.
2. **DSL for experiment description:** to define a scientific simulation in the form of a flow based paradigm to define the overall experimental process. The process can be composed of multiple pieces of code or refer to some existing simulation code, and will simplify the configuration of experiments and managing dependencies. This DSL will likely be specific to the model concerned.

The first case study we conducted was for the automated deployment of the Weather Research and Forecasting model (WRF) [2], a complex atmospheric numerical weather prediction system, on a cloud architecture. The intention was to abstract the process of installing, configuring and running the model, and use this learning to inform the development of the two DSLs. The first building block was an automated install and deployment framework cutting out hours of work. The result was taken to a workshop conducted with WRF expert users to get their feedback related to abstraction, automation and usage of cloud services. We found the feedback on this work promising. Firstly, the time to install a model has been reduced to minutes when previously it could take days for an experienced user or months for a beginner. Secondly, our end-user community were excited to see new opportunities for running their scientific simulations within cloud architectures.

We aim to expand our work by looking at further areas where software engineering may enhance environmental science. One avenue is to explore container-based virtualisation to support a single model deployment to a variety of environments. Moreover, to provide a framework to run more complex model experiments, for example ensemble runs. In addition, we aim to enrich this work with more case studies from environmental science, including for example the TOPMODEL [1] hydrological model.

3 Conclusion

We believe software frameworks and DSLs can facilitate seamless deployment of environmental models, allowing scientists to take advantage of on-demand cloud computing, supported by emergent software architectures. Cloud computing provides an ideal opportunity to not only extend computational resources and provide levels of abstraction, but integrate with other useful services such as data stores, analytics and machine learning.

4 Acknowledgements

This work is supported by two EPSRC grants involving the Data Science Institute at Lancaster University and CEH: Models in the Cloud: Generative Software Frameworks to Support the Execution of Environmental Models in the Cloud, EPSRC: EP/N027736/1; and a DT/LWEC Senior Fellowship (awarded to Blair) in the Role of Digital Technology in Understanding, Mitigating and Adapting to Environmental Change, EPSRC: EP/P002285/1. We thank all our partners involved in these initiatives for their insights and contributions to this work.

References

- [1] K.J. Beven and M.J. Kirkby. A physically based, variable contributing area model of basin hydrology. *Hydrological Sciences Bulletin*, **24:1**, 43-69, (1979).
- [2] W.C. Skamarock, J.B. Klemp, J. Dudhia, D.O. Gill, M. Barker, K.G. Duda, X.Y. Huang, W. Wang, J.G. Powers. A description of the Advanced Research WRF Version 3. Technical Report. National Center for Atmospheric Research. 1-113, (2008).