Geophysical Research Abstracts
Vol. 12, EGU2010-3680, 2010
EGU General Assembly 2010

Using the Workflow Engine TRIDENT as a Hydrologic Modeling Platform

Michael Piasecki and Bo Lu
Drexel University, Dept of Civil, Architec. & Env Engrg, Philadelphia, USA (mp29@drexel.edu)

This is an invited contribution.

Hydrologic communities around the world have developed a plethora of codes in a multitude of programming languages over the past few decades in order to assess environmental processes and to predict changes in the hydrologic realm. While these codes represent a vast amount of knowledge, expertise and resources spent and also have been extremely useful for the purposes they were designed, increasingly the realization emerges that a better coordination and coupling of both models among each other and also to data sources that drive the models is essential. The emergence of the idea of community models or integrated modeling systems (such as CSDMS, ESMF, or WRF) as well as coupling frameworks (such as openMI) have opened up new ways of thinking about how to link up legacy codes or integrate them into frameworks that allow for a more holistic modeling approach than before.

This paper discusses the effort that has been undertaken to utilize a workflow engine (MicroSoft's TRIDENT system) for the purpose of designing a modeling environment that permits the seamless integration of data flows from source, to preparation (preprocessing), to ingestion, to model execution, to harvesting (postprocessing) and analysis of the generated result data through the design of workflow sequences. While this approach has the great benefit of documenting the modeling effort from A to Z so it is repeatable (this is very important for provenance) it also permits the creation and collection of actors (or activities) that can be reused by others either in the same or other workflow environments. For example, this allows for creating a number of alternative execution modules that can be linked together in a workflow sequence and then automatically compared via sets of post processing activities thus providing great time savings.

We will be demonstrating how data sources can be directly tapped into via the use of web services, how they can be interpreted and prepared for model preparation (for example, to convert digital elevation model (DEM) data to a TIN and then a modeling grid for the delineated watershed), how legacy code can be integrated into the workflow, and how modular process actors van be liked to form a complex hydrologic modeling system, and how these numerical kernels can be executed. The resulting data volumes can then be stored in common data stores (such a netCDF) or be analyzed via statistical actors to automatically extract information about the generated data before it is been stored. We will address issues of scalability of these systems, and also offer an opinion on the ease of use and feasibility of a system like this for community modeling purposes.