Publishing Active Workflows to Problem-Focused Web Spaces

Luigi Marini, Rob Kooper, Peter Bajcsy, James D. Myers
National Center for Supercomputing Applications of the University of Illinois
{lmarini, kooper, pbajcsy, jimmyers}@ncsa.uiuc.edu

Abstract

We present a new paradigm for publishing scientific workflows to the web. Publishing in this system involves not only creating a web space for running the workflow on demand, but also the ability for the publisher to convey expertise about its use by providing ways to focus attention on problem-relevant parameters, inputs and outputs.

As part of the publishing step we include the ability to configure visualizations for specific outputs of the workflow. The resulting system is a web based environment focused on the ability to publish existing workflows so that other users in the community can execute those workflows and visualize the results from the web. Workflow executions and visualizations can be shared with the community along with workflow definitions.

1. Introduction

Scientific workflow management systems support scientist by providing frameworks for the reuse and sharing of codes and algorithms. Most scientific workflow environments are desktop based as opposed to web based to provide more flexible graph based editors for workflow creation and provide local resources for the orchestration and execution of workflow steps.

We argue that this methodology supports the scientific user who is creating the workflow, but that it does not help in reducing complexity for the end user who is interested in executing a particular workflow to create new results and investigate a particular scientific problem. To address the needs of end users who might not be workflow creators, we present a system to publish workflows as web environments that support workflow execution and allow the publisher to convey their expertise about its use by focusing attention on problem-relevant inputs and outputs. The environment provides input forms that (1) present the salient aspects of the workflow that should be altered by the end users (2) provide a web based visualization framework to visualize output data in predefined ways, and (3) keep track of provenance of the current and past executions. The overall system stores the underlying semantic web of information linking visualizations, forms, workflows, models, data, and executions in distributed repository in the Resource Description Framework (RDF) format.

Sites such as MyExperiment [1] provide social networking environments for sharing workflow definitions, but do not provide simple ways to execute existing workflows and to store results. On the other hand, sites such as Many Eyes [2] provide ways to visualize specific datasets uploaded as part of the visualization creation process, but do not support dynamic creation of data or of visualizations that result from the same process run with different parameters. In some sense, our work bridges these two concepts by providing a combined way to share processes and results in an interactive, collaborative framework. While useful as a stand-alone capability, we also see such a system as a critical enabler of rich virtual observatories and digital watersheds that provide a holistic view of observational and modeled data [3]. We present a prototype called the Digital System Explorer (DSE) built on top of open standards, and open source libraries and tools.

2. Web Publishing

There are several motivations for providing simple ways to publishing workflows in custom problem-focused environments. Workflow representations can be too complex for the end user who wants to execute the code without having to understand implementation details. Web based execution environments provide a second abstraction layer on top of the workflow layer that can focus attention on science and even on a particular scientific problem. Publishing workflows on the web allows scientists to quickly share their untested workflows and data with other users. At the same time using the web publishing paradigm allows colleagues, students and the general public to experiment with these workflows with minimal hardware and software requirements. Most scientists nowadays are highly
versatile in the use of web technologies, and simple web interfaces should put very low requirements on the end user.

By coupling the visualizations with the execution environment, users can publish interactive visualizations that are driven by models and analysis tools executed on the fly. Conversely, one can view the benefit as gaining a way to scaffold use and provide meaningful visualizations of workflow outputs instead of simply providing a raw template or execution service. Publishers can create interactive visualizations driven by different workflows and a single workflow can be published in multiple ways depending to support different use scenarios.

In the current implementation users select the workflow they want to publish from the ones available in a shared content repository and provides metadata to configure the online environment (for example a title and a description). Then select which parameters and inputs the user will enter via the web—these parameters are the “knobs” the scientists want the end users to be able to turn—and provide alternative names, descriptions and groupings for those parameters and inputs. Finally the select which data sets produced by the workflow will be published as part of the result page of a particular execution and configures interactive visualization widgets for each of the published outputs.

The resulting web execution environment is published and made available to the community. When a user submits a new execution, a new instance of the workflow is launched with the parameters and inputs they entered. The results are returned in the predefined visualizations and are stored to be shared with other scientists as part of a library of executions and visualization of derived data sets.

3. Architecture

The system consists of five main components, Tupelo, DSE, Cyberintegrator, and a scheduling service.

The system is built on top of Tupelo providing a single view to different metadata stores. The separate parts of the system communicate using a shared content repository which stores information in the RDF format.

The DSE allows for the creation of workflow execution forms and visualizations. Forms and executions can be shared to foster collaboration and to rapidly share information in a group of scientists. Every time a new execution is submitted a new set of derived data sets and visualizations becomes available. Users can download those data sets, the underlying workflow or they can share a particular execution with colleagues by sending a URL.

The Cyberintegrator [3] is the scientific workflow management system used to create and execute workflows on the desktop. Cyberintegrator stores all information needed to run a workflow in the shared repositories. When creating new tools inside the Cyberintegrator, using convenient wizards, the tool definitions as well as any additional resources needed are stored in shared content repositories.

The Cyberintegrator server can be used to execute workflows or steps in a workflow on a remote machine. All information needed by the server is available in the shared content repositories, and results are store back in the shared content repositories. This server can be used to schedule jobs on more powerful HPC's.

A scheduling service allows for running workflows as specific intervals in time. For example this enables a scientist to schedule a workflow to run to collect data from a remote sensor every day. It will process the data and have the data and visualizations ready for inspection every morning when the scientists walks into the office.

5. Acknowledgements

The authors would like to acknowledge the Office of Naval Research, which supports this work as part of the Technology Research, Education, and Commercialization Center (TRECC) (Research Grant N00014-04-1-0437) managed by NCSA.

7. References


