Scufl2 – because a workflow is more than its definition

Stian Soiland-Reyes, Alan R Williams, Stuart Owen, David Withers and Carole Goble
School of Computer Science, University of Manchester, UK
{stian.soiland-reyes, alan.r.williams, stuart.owen, david.withers, carole.a.goble}@manchester.ac.uk

Project site: http://www.taverna.org.uk/
Source code: https://github.com/mygrid/scufl2 http://taverna.googlecode.com/
License: GNU Lesser General Public License (LGPL) 2.1

Taverna is a scientific workflow management system that has gained popularity amongst scientists in bioinformatics, chemistry, astronomy and other domains. Since its inception in 2003, Taverna's workflow language has evolved from *Scufl* which was designed for the *FreeFluo* workflow engine. Scufl defines a directed acyclic graph of the *data flow* between *processors* (services) which receive and produce data at input/output *ports*. Scufl workflows are stored in a lightweight XML format.

t2flow adapted these concepts for the Taverna 2 workflow engine, including new capabilities such as extensible execution control. The t2flow XML format was created as a serialisation of internal Java beans, which unfortunately makes it difficult to consume or produce by non-Taverna clients compared to Scufl. There have also been growing demands for bundling a workflow with related resources such as example data and semantic annotations. Addressing these issues culminated in forming a new workflow language *Scufl2,* which is presented here.

By adapting *linked data* technology and preservation methodologies for research objects (http://bit.ly/dKiGz7), Scufl2 is not only a platform-independent workflow language which can be inspected, modified, created and executed by third-party tools and systems; it is extensible to allow the capture of workflow execution inputs and outputs, reference data sets, provenance, annotations, documentation, publications, alternative formats and representations, and even embedded binary service implementations.

Scufl2 comes with a Java API independent of Taverna that can be used for programmatic access to read and write Scufl2 *workflow bundles.* A workflow bundle is a structured ZIP-file based on *Adobe UCF*, *ePub OCF* and the *Open Document Format* (ODF), with the workflow definitions included as XML Schema-conformant documents which are also valid RDF/XML. The constraints from the schema allow clients to read and write Scufl2 workflow definitions as regular structured XML. Richer RDF-enabled clients may link workflow definitions with external resources and additional annotations stored in separate RDF files in the bundle, perhaps using vocabularies such as Dublin Core.

The workflow structure is defined using an OWL ontology, and annotated with URIs so that third parties can form semantic statements about any component of any Scufl2 workflow, for instance to say that a particular service produces outputs of a certain type, or that a given data link was added by a different researcher to the builder of the rest of the workflow. An unzipped workflow bundle can be made public using any standard web server and become part of the Linked Open Data cloud.

Semantic annotations and a manifest for the bundle declare the purpose of, and links between the different components forming a workflow. This allows third parties to extract and append annotations about data and services used by the workflow. The evolution of the workflow definition itself can also be included in these annotations, with references to previous versions and authors.

In recent years, Taverna has become an interesting target for researchers investigating published workflows, such as those found on myExperiment. Exploring these workflow definitions using the Scufl2 API can reveal compatibilities and implicit data types of web services, providing new annotations for service registries like BioCatalogue. Sites and tools can add additional descriptions of the workflow and the experiment directly to the Scufl2 workflow bundle without affecting execution behaviour or requiring any deeper understanding of the format. They may for instance automatically insert links to the original source and author in a downloaded bundle, which can be persisted even if the workflow is evolved further and distributed elsewhere.

Using these capabilities to their full extent, a workflow bundle is a research object, including everything required for a full re-enactment of the virtual experiment (possibly a full virtual machine in OVF format), full provenance and data sets of a published workflow execution, and semantic descriptions of what are the purposes and origins of the produced values. By also including a PDF representation of the paper arising from the experiment, Scufl2 captures a fully repeatable and reproducible e-science publication.

As a general workflow language, scufl and t2flow have been the target of translations and generation for example by BioMoby's Seahawk, and executed on alternative engines, like MOTEUR. By defining the workflow language independently from the execution engine, Scufl2 encourages such extensions and wider use. This presentation demonstrates how the Scufl2 format can be used from Ruby and Clojure programs to generate and execute a Scufl2 workflow that can also be executed by Taverna.

Funded by: EPSRC EP/G026238/1; European Commission's 7th FWP FP7-ICT-2007-6 270192; FP7-ICT-2007-6 270137