Delivering Web Service Coordination Capability to Users

Tom Oinn
EMBL European Bioinformatics Institute

Matthew Addis, Justin Ferris, Darren Marvin
IT Innovation Centre

Mark Greenwood, Carole Goble
University of Manchester

Anil Wipat, Peter Li
University of Newcastle upon Tyne

Tim Carver
MRC Rosalind Franklin Centre for Genomics Research

ABSTRACT

As web service technology matures there is growing interest in exploiting workflow techniques to coordinate web services. Bioinformaticians are a user community who combine web resources to perform *in silico* experiments. These users are scientists and not information technology experts; they require workflow solutions that have a low cost of entry for service users and providers. Problems satisfying these requirements with current techniques led to the development of the Simple conceptual unified flow language (Scufl). Scufl is supported by the Freefluo enactment engine [1], and the Taverna editing workbench [3]. The extensibility of Scufl, supported by these tools, means that workflows and the use of web services can be matched to how users view their problems. The Taverna workbench exploits the web to keep Scufl simple by retrieving detail from URIs when required, and by scavenging the web for services. Scufl and its tools are not bioinformatics specific. They can be exploited by other communities who require user-driven composition and execution of workflows coordinating web resources.

Categories and Subject Descriptors

D.2.6 [**Programming Environments**]: Integrated environments, Programmer workbench

General Terms

Design, Experimentation, Languages

Keywords

Web services, scientific workflows, web service coordination, web programming, e-science, bioinformatics

1. INTRODUCTION

Web services use existing Internet communications and data exchange standards to provide applications with a web-based application-programming interface. As web service technology matures there is considerable interest in exploiting workflow techniques to coordinate web services. Workflow techniques form an important part of *in silico* experimentation within the bioinformatics domain and potentially allow the e-scientists to describe and enact their experimental processes in a structured, repeatable and verifiable way. These users are scientists and not information technology experts; they require workflow solutions that have a low cost of entry for service users and providers, including ease of use and open source tool support**.** As a result, the EPSRC funded myGrid [2] project has, in collaboration with the European Bioinformatics Institute and the Human Genome Mapping Project, developed a graphical toolset and workflow enactment engine which uses its own high level representation of a process flow, Simple conceptual unified flow language (Scufl).

2. SCUFL LANGUAGE AND TOOLS

In Scufl a workflow is a network of processors and links. It is supported by the Freefluo [1] workflow enactment engine, and the Taverna workbench [3] for editing and testing Scufl workflows. The Taverna workbench exploits the web in two ways: by keeping Scufl simple and dynamically retrieving detail from URIs, (e.g. URIs for WSDL interfaces descriptions) when required, and by scavenging the web to populate the list of potential processors that its user can incorporate into a workflow.

The Scufl language is primarily aimed at users who currently use web forms or scripting languages to interact with web resources. They want to develop workflows in an exploratory fashion, and to easily share such workflows with their colleagues. The Scufl language aims to be practical. It recognizes that there is, and will continue to be, significant development in computer accessible web resources in general and web services in particular. As a result, the language is designed with extensibility mechanisms so that it can be extended to incorporate new developments, or specialized to the de-facto standards of a user community.

The principles behind the Scufl language are captured in the origin of the acronym: simple, conceptual, unified, flow language.

Simple. The language aims to be as simple as possible. The target user communities have domain expertise and domain problems to investigate; their interest in programming is secondary.

Conceptual. A Scufl workflow should match the users' conceptual model of their problem. Implementation detail should be kept to a minimum, by exploiting techniques such as introspection over service descriptions.

Unified. Coordinating web services was the initial motivation. However users want to treat these and other services, which are essentially the same from their perspective (e.g. grid services, local applications), in a unified manner.

Flow Language. The basic conceptual model is of a network of processing activities linked by data and control flows.

The core of a Scufl workflow is a network of processors (nodes) connected by data links. In addition to these two entity types, there can also be input and output nodes to represent to overall workflow interface, and coordination constraints for relationships between processors that are not captured by data links. A Scufl model consists of a set of instances of these types. Each instance has a unique name within the workflow. Figure 1 is a simple example, from the Taverna workbench, of a Scufl workflow

Copyright is held by the author/owner(s).
WWW 2004, May 17-22, 2004, New York, NY USA.
ACM xxx.

showing one input, three processors and two outputs, linked by six data links.

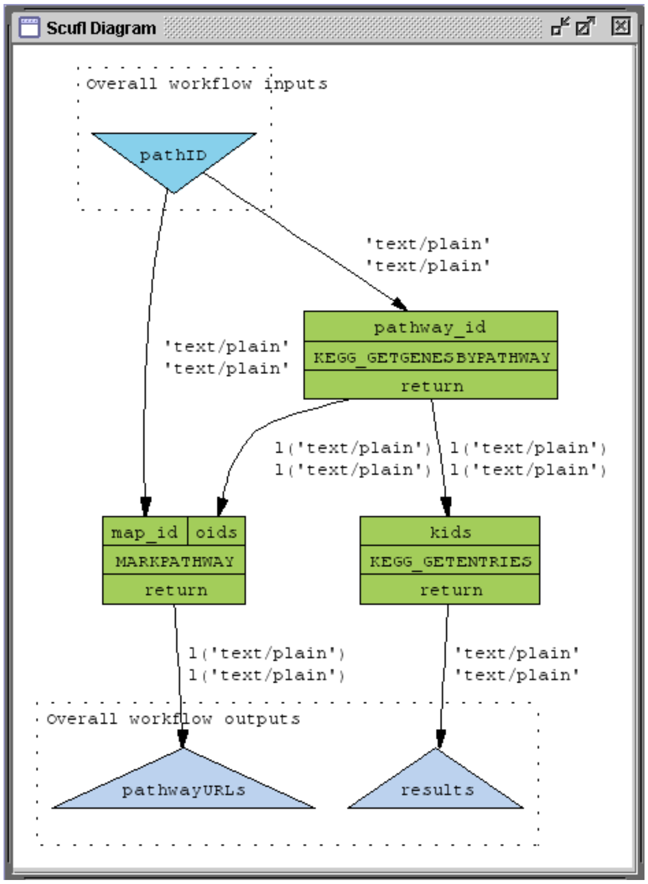

Figure 1 Diagrammatic view of simple Scufl workflow

The combination of Scufl, Taverna and Freefluo is already being used and tested by an expanding community. Bioinformaticians have created workflows that include services developed independently and found on the web. For example the DDBJ-XML services (http://xml.ddbj.nig.ac.jp/wsdl/index.jsp) and KEGG services (http://www.genome.ad.jp/kegg/soap) have proved to be valuable resources. These tools have also been used, as part of the myGrid project, in the design and development of a set of sophisticated workflows for researchers investigating Graves' disease, an immune disorder causing hyperthyroidism, at the Newcastle Centre for Life.

The Taverna workbench provides a number of different views of a Scufl workflow: an explorer view of the model entities, a diagrammatic view, an XML view (e.g. Figure 1), and a view for providing inputs and starting workflow instances. The Taverna workbench also includes an "available services" view. This can be seen as a view of the web. It is a simple browser of those services, known to Taverna, whose operations can be added to a model as a new processor. This "available services" view provides users with a palette of services when changing a model. It is populated either from models loaded into Taverna, by introspection on a service description, or by simple web search and introspection (scavenging). There is ongoing work into enhancing this simple "registry" with semantic annotations, richer search facilities and hence better guidance during (perhaps semi-automatic) workflow construction, based on users' high-level specifications of their *in silico* experiments.

3. CONCLUSIONS and USER FEEDBACK

The Taverna workbench, with its embedded Freefluo enactment engine, gives e-scientists an environment for using the web from a perspective of services and compositions of services (workflows). They can browse service resources available on the web, construct workflows combining services, and test them out. In this way, Scufl has proved excellent for the exploratory, information gathering workflows that are its prime concern. As users have got past the novelty of testing out Taverna and Freefluo, they have started to use them in their day-to-day work. An early request, now incorporated, was for much better status information and recovery support; vital if your workflow takes a few hours to run and the web resources used are not 100% reliable.

The ability of Scufl to match its users' conceptual model for expressing their problem is based on the extensibility gained through the declaration of new Sculf processor types. The utility of this approach has been shown by the development of processor types for specific types of bioinformatics resources, and ongoing extensions implemented in terms of further new processor types.

The use of workflow techniques to coordinate web services for scientific users is an active research area [4]. Scufl's niche in this community is as a user-driven language that is simple, flexible, based on web services, and supported by open source tools.

4. ACKNOWLEDGMENTS

Thanks are due to the Taverna and Freefluo user communities.

The myGrid project is supported by the UK e-Science programme (EPSRC grant GR/R67743). The authors would like to acknowledge the myGrid team (past and present).

5. ADDRESSES

Tom Oinn, EMBL European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Essex, CB10 1SD, UK, Tel. +44 (0)1223 494 677, tmo@ebi.ac.uk. **Matthew Addis, Justin Ferris and Darren Marvin**, IT Innovation Centre, 2 Venture Road, Chilworth Science Park, Southampton, UK, Tel. +44 (0)23 80 760834, {mja, jf, djm}@it-innovation.soton.ac.uk. **Mark Greenwood and Carole Goble**, Department of Computer Science, University of Manchester, Oxford Road, Manchester, UK, Tel. +44 (0)161 275 6186, {markg,carole}@cs.man.ac.uk. **Anil Wipat and Peter Li**, School of Computing Science, University of Newcastle upon Tyne, NE1 7RU, UK, Tel. +44 (0)191 222 7856, {Peter.Li,Anil.Wipat}@ncl.ac.uk. **Tim Carver**, MRC Rosalind Franklin Centre for Genomics Research, Hinxton, Cambridge CB10 1SB, UK, Tel. +44 (0)1223 494 500, tcarver@hgmp.mrc.ac.uk.

6. REFERENCES

[1] Freefluo workflow enactment engine. http://freefluo.sourceforge.net

[2] myGrid - directly supporting the e-scientist http://www.mygrid.org.uk

[3] Taverna project. http://taverna.sourceforge.net

[4] UK National e-Science Centre (NeSC) workshop on "e-Science Workflow Services", Edinburgh, Dec 2003, http://www.nesc.ac.uk/action/esi/contribution.cfm?Title=303