Databases and ontologies

The Taverna Interaction Service: enabling manual interaction in workflows

Anders Lanzén1,* and Tom Oinn2

1Computational Biology Unit, Bergen Center for Computational Science, University of Bergen, 5008 Bergen, Norway and 2EMBL European Bioinformatics Institute, Hinxton, Cambridge, CB10 1SD, UK

Received on June 12, 2007; revised on February 12, 2008; accepted on March 1, 2008

Advance Access publication March 12, 2008
Associate Editor: John Quackenbush

ABSTRACT

Summary: Taverna is an application that eases the integration of tools and databases for life science research by the construction of workflows. The Taverna Interaction Service extends the functionality of Taverna by defining human interaction within a workflow and acting as a mediation layer between the automated workflow engine and one or more users.

Availability: Taverna, the Interaction Service plug-in and web application are available as open source and can be downloaded from http://taverna.sourceforge.net/

Contact: taverna-users@lists.sourceforge.net

1 INTRODUCTION

Recent years have seen a virtual explosion in the number of analytical applications, databases and other resources available for life science research. Furthermore, these tools and databases (hereafter called services) are often heterogeneous in terms of access, data formats and definitions used, as well as user interfaces (Stein, 2002). Manual cut-and-paste work, as well as competence in bioinformatics and programming is often needed in order to combine results from several services in a meaningful way. Web service technology provides a solution to some of these issues, by specifying a standardised programmatic interface for computational resources. This technology has gained popularity within bioinformatics recently with an increasing number of services now providing Web services access (Neerincx et al., 2005). The application Taverna (Oinn et al., 2007) offers an environment to access Web services through a graphical user interface, without technical knowledge of Web services or programming. Most importantly, it allows for the creation, execution and reusability of workflows, by combining several services in a coordinated and well-defined manner. While many other workflow editing environments exist, Taverna is one of the most popular in life sciences with an estimated user base of around 1500 installations in February 2006 (Hull et al., 2006). It has also been actively used in genomics research (Stevens et al., 2004).

In addition to computer resources, there is often a need to include user intervention in workflows. More often than not, automatically generated prediction results require some form of manual quality control. In the standard version of the Taverna Workbench, a user cannot control the behaviour of a workflow once it is running. In simple workflows with relatively short execution time, this can be dealt with by manually inspecting intermediate or final results and restarting the workflow with modified parameters, as needed. In a typical genomics project, however, there is a need for workflows that include large volumes of data and computationally demanding services. Total running time of such workflows can be as long as several hours or even days. Further, there is often a need to include other people than the primary Taverna user in the review process. This could also include external collaboration partners that may not have direct access to the same file server as the primary investigator.

A simplified example of a workflow that includes manual interaction is illustrated in Figure 1. This conceptual workflow predicts the boundaries of the protein-coding portion of genes in a bacterial genome, and subsequently predicts the function of the encoded protein. This illustrates the requirement to include user interaction as an integrated part of a workflow, which raises the issue of how to define human interaction to the workflow designer and ultimately to the workflow engine executing the workflow. To this end, we have developed the Taverna Interaction Service, an extensible mediation layer in between the automated workflow system and the user. As far as the workflow design is concerned, there is no obvious reason to separate this kind of human inspection from computational analysis, hence the slogan ‘because users are services too’ was chosen for the application.

2 IMPLEMENTATION AND FEATURES

The Interaction Service is implemented as a Java web application, providing a programmatic interface for communication with the Taverna workflow engine. Thus, it can be deployed to a Java Servlet container independently of the Taverna installation of its user. Besides its programmatic interface towards Taverna, it presents a status screen to the user when accessed through a web browser, showing its status and available interaction patterns. A working demonstration workflow can also be downloaded from this web page.

In order for a Taverna Workbench installation to communicate with the Interaction Service, a plug-in must be installed.

*To whom correspondence should be addressed.
The second default pattern provided handles genome annotation data. The input for this process consists of one or more genome ‘flat files’ in EMBL, Genbank or GFF format, each with a title. A textual comment may also be included. When invoked, this pattern will send an email to the targeted reviewer containing the comment submitted and a hyperlink for opening and reviewing the results. Following this link will launch a modified version of the Artemis sequence and annotation editor (Rutherford et al., 2000). This step utilises Java Web Start technology. Upon opening, the genome and annotation data is automatically downloaded from the Interaction Service and presented to the reviewer, who then reviews and edits the data in Artemis as usual. A notepad is also provided for writing down comments about the data and modifications made. Having reviewed the data, the user can choose to either accept the results with or without changes, or to reject them. The edited data is sent back to the Interaction Service along with review notes and the decision made, and appears as the output of the interaction processor in the workflow.

There are many situations where an interaction pattern may be useful to allow manual interaction in an analysis workflow. Thanks to the modular design of the Interaction Service that such interaction patterns are relatively easy to define, create and add at runtime, providing basic programming skills and familiarity with Java. The developer of a new pattern must first download the full java source tree from the Taverna website and then implement the ServerInteractionPattern interface, preferably by inheriting the partial implementation AbstractInteractionPattern. The new ServerInteractionPattern can then be compiled and added to a .jar file. This file may be uploaded to an Interaction Service web server, which can discover and add it to its repository at runtime. More information about how to do this can be found at http://bioinfo.no/software/interaction-service.

3 DISCUSSION AND FUTURE PERSPECTIVES

User interaction in workflows is far from a new topic, particularly in the field of business process management (BPM). Limiting the scope to Web service technology, the de facto standard for modelling executable workflows today is Business Process Execution Language (BPEL) (http://docs.oasis-open.org/wsbpel/2.0). Taverna instead uses a workflow language called SCUFL (Simple Conceptual Unified Flow Language), containing a number of features that separates its functionality quite fundamentally from BPEL. However, what the two languages have in common is that neither of them define an interface for human interaction in workflows. To include user interaction, many developers of BPEL workflow software have overcome this limitation by implementing special web applications, presenting a BPEL compliant WSDL (Web Service Description Language) interface to the workflow engine, but handling user interaction by mechanisms hidden from the workflow engine, i.e. external to BPEL. However, no adopted standard for describing the interaction interface itself exists, so a workflow designer often needs knowledge of the inner workings of the web application responsible. The Taverna Interaction Service uses a similar approach, but does not expose
Interaction processes as WSDL, which in our opinion is poorly suited for this purpose. Instead, it uses a custom format exposing some of the interaction related metadata to the workflow designer. The communication between the external Interaction Service and the user is initiated by email. It should be noted that a consortium of major BPM software developers recently proposed an extension of BPEL called BPEL4People and WS-HumanTask specifying how user interaction can be included in workflows, but whether this will contribute to increased standardisation of the interaction interface and portability of BPEL workflows with user interaction steps, is still unclear.

In connection to ongoing genomics projects at CBU, we are planning to implement a number of new interaction patterns. These will be added to the Interaction Service distribution. One such pattern is selecting relevant sequence alignments from a list. Another is curating a list of automatically generated gene names.

ACKNOWLEDGEMENTS

We thank the Taverna development team, Pål Puntervoll and three anonymous reviewers for helpful comments on this manuscript and Jan-Christian Bryne for fruitful discussions and debates about WS technology. The development of the Taverna Interaction Service has been supported by the UK e-Science program through the myGrid project and by the National Programme for Research in Functional Genomics (FUGE) of the Research Council of Norway.

Conflict of Interest: none declared.

REFERENCES