Searching Toxics: A Web Service for Chemoinformatics Workflows

Jungkee Kim

Department of Glocal IT, Sungkonghoe University,
1-1 Hang-Dong, Kuro-Ku, Seoul, 152-716, Korea
jake@skhu.ac.kr

Abstract. Service-oriented e-science workflow has emerged as a paradigm for integrating heterogeneous distributed science computations. Life sciences also utilizes workflow management systems based on chemical information for accelerating scientific progress. We have developed an infrastructure of chemoinformatics Web services that make those approaches efficient. In this paper, we describe a Web service wrapper for estimating toxic hazard within chemoinformatics workflows. The wrapper makes a secluded application functioned as a Web service process and participated in a series of workflow processes.

Keywords: Workflow, Web service, Chemoinformatics.

1 Introduction

Service-oriented computing based on Web service technology has recently emerged as a paradigm to address the traditional challenge of heterogeneous systems integration. A set of atomic Web services can compose workflows with control and data flow among the services. The Business Process Execution Language (BPEL) and the Business Process Modeling Language (BPML) are proposed for industry standards of the business service composition. E-science workflows have similarity to the business workflow for their requirements. Utilizing workflow management systems to automate computational activities provides benefits to various fields such as astronomy, biology, chemistry, environmental science, engineering, geometry, life science, physics, and social sciences.

In life science, drug discovery scientists demand rapid information process to deal with large amount of data produced by technologies like high throughput screening (HTS), micro-array assays and combinatorial chemistry. The Murray-Rust group at Cambridge has accomplished those chemical information studies in distributed environments since the early stage of the Web [1]. A chemical information research group at Indiana University has developed an infrastructure of chemoinformatics Web services [2]. And the services through the infrastructure can easily utilize the information.

A survey for Grids workflow projects can be found in [3]. And the survey introduced seven major projects: Triana [4], Kepler [5], Taverna [6], Pegasus [7],

ASKALON [8], QoWL [9], and GPEL [10]. Triana is started from a single platform but supports distributed services with Grid awareness. Kepler is also started from a single platform and it fully supports Grid environment. It is widely used in e-Science workflows. Taverna is part of MyGid[11] project and focuses on applications of life science. It recognizes the importance of provenance and semantics. Pegasus concentrates large scaled applications. ASKALON is another Grid framework to support distributed workflows. QoWL supports Quality of Service in the workflow based on BPEL. GPEL is another BPEL based system for Grid and scientific workflow focusing on dynamic and adaptive large scaled application.

We utilize Taverna for a workflow tool. The Taverna workflow framework requires a local process or a Web service process to serialize several operations into a scientific workflow. A number of chemical structures are probed their toxic hazard and proceeded to other analytical processes in the workflow. We employ a Java application named ToxTree[12], which estimates toxicity of chemical components. But the application has no communication to other remote processes on the Web. To avoid local installation in the workflow workbench, we design a wrapper for the secluded application to communicate on the Web.

The rest of this paper is organized as follows. In the next section we describe our workflow framework, Taverna Work bench. Section 3 describes ToxTree, the toxic hazard estimation application. We illustrate the Web service wrapper for science applications in Section 4. In Section 5, we summarize and conclude.

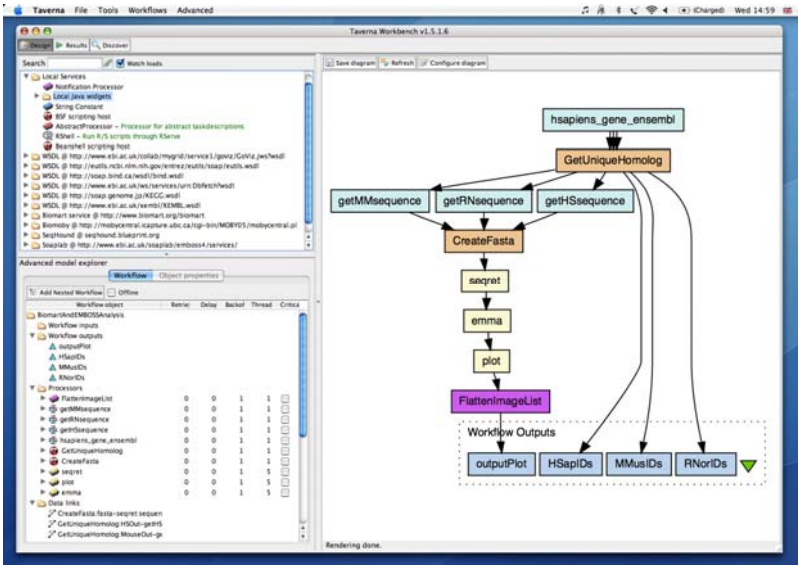

Fig.1. Graphical user interface of Taverna workbench [13]

2 Taverna Workflow Workbench

Taverna is a workflow framework designed for life science applications. But it is also applicable to general workflow compositions. Taverna is developed as part of MyGrid project, which builds a middleware for data-intensive processing. The workflow construction based on the service composition is a powerful approach and there are over thousand applicable component services in workflows.

Figure 1 shows a graphical user interface workbench of Taverna and it formulates workflows. Through the user friendly interfaces, the workbench can compose complex workflows. The workbench provides decorative and explorer illustration of workflows. It enables users to compose their own workflows or to load existing workflows acquired from a common repository with expert providers. The workbench also lists the available resources where the workflows can run. After resources and enactor engine types are selected, the workflow can be started and its progression is monitored. The workflow can be interrupted for cancelling a step or stopping the whole process.

The Taverna workbench depends on the Simplified conceptual workflow language (Scufl). The sculf language is designed for a data-centric flow. A network of processors and links composes Scufl. The input and output nodes as well as basic entities can be added to Scufl. The language is useful for the users who mastered Web forms and script languages. It also provides practical Taverna extensions.

3 ToxTree Application

The European Chemicals Bureau (ECB) within the European Commission's Joint Research Centre (JRC) has developed a Java application, which estimates toxic hazard in chemical structures. The application enables structure-based predictions like a decision tree approach. Currently it applies Cramer rules, Verhaar scheme, Skin irritation, Eye irritation, and Benigni/Bossa rulebase. Datasets in SMILES chemistry format from various file types can be accepted as input. Though it is not applicable to the Web service, the original software provides a graphical user interface to organize user-defined molecular structures either in SMILES [14] format or in a graphical format using built-in two dimensional structure diagram editor.

The application is suitable for a standalone PC, and this means it is not sharable to other remote processes on the Web. In our Web service, only the text molecular format is allowed. The figure 2 shows an ordinary graphical user interface of a ToxTree application.

4 Web Service Wrapper for Science Applications

Many science applications are platform dependant and often written in traditional program languages like C or FORTRAN. They are usually difficult to integrate with applications from other disciplines.

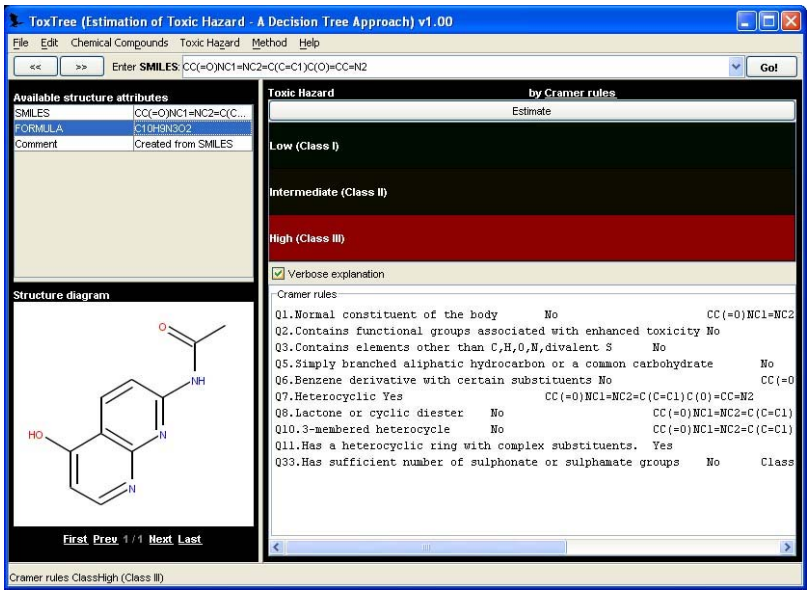

Fig.2. Graphical user interface of ToxTree

Access from remote programs to these applications is normally difficult. There is no standard way to illustrate their input and output or to monitor their progressing. By converting these platform dependant shell applications into application services, the application can connect to other application and enlarge its usage.

Generic Service Toolkit (GFac) [15] can convert any command-line application into an application service. With a set of input parameters, it launches the application, monitors the application, and returns the results. The toolkit requires no modification of program code in the application. The toolkit also has Web user interface producing tools. When a user accesses an application service, the user is presented with a graphical user interface (GUI) to that service. The interface encloses a list of processes that the user enables invoking on that service. After selecting an operation, the user is accessible with an interface for the operation, which enables the user to specify all the input parameters to the operation. And the user can invoke the operation on the service and obtain the output results.

OPAL [16] is a Web service toolkit, which provides the features of scheduling, security, and persistent state management. The toolkit utilizes the Globus GRAM for job submission, and requires a configuration file for the location of the application, and command line options can be provided on demand. The Web services based on Opal enables the development of generic Web services in an application framework.

Our solution prior to the above toolkits leverages Apache Ant to wrap the legacy application. The wrapper makes the application services on the Web. The wrapper can call executables and set environment variables. There are a lot of useful built-in scripts in the Apache Ant and they are also extensible. We developed Ant build scripts to run the ToxTree application. In our system a Java program can call other Java programs through the Ant.

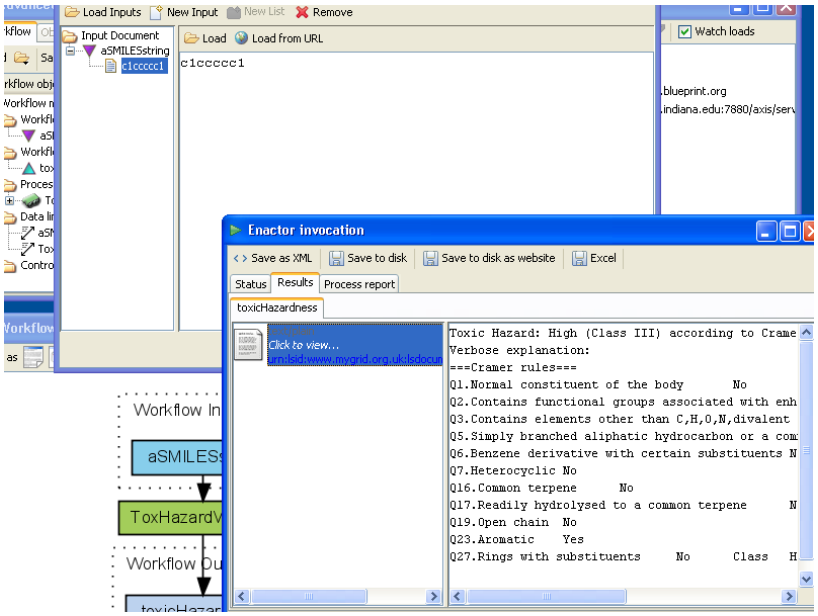

Fig.3. Taverna workbench with ToxTree application service

Figure 3 illustrates our implementation of the legacy application wrapping Web service process in the Taverna workflow workbench. In the picture the result screen shows the verbose text return values against the benzene SMILES input. There is a simple mode to print out the toxicity only.

5 Conclusions

Our approach demonstrates the feasibility of using a Web service wrapper to estimate toxic hazard within chemoinformatics workflows. The wrapper enables a pure Java application functioned as a Web service process and participated in a series of workflow processes. This wrapped Web service roles a distributed process attached to any workflow, which establishes a level of exposure for all chemicals. And various works using our Web service can reduce the threat to human health.

Acknowledgments. This work is implemented when the author worked at Community Grids Lab, and supported by the National Institutes of Health under NIH Roadmap Molecular Libraries Initiative for Exploratory Centers for Cheminformatics Research (NIH Grant No. 1 P20 HG003894-01). Sungkonghoe University also supports part of this paper. The Author would like thank Dr. Geoffrey Fox and Dr. Marlon Pierce at Community Grids Lab for their support. We would like to acknowledge Peter Murray-Rust group at Cambridge and the Chemoinformatics group at Indiana University for their exchange of ideas.

References

1. Murray-Rust, P., Rzepa, H. S., Chemical Markup, XML, and the Worldwide Web. 1. Basic Principles. Journal of Chemical Information and Computer Sciences 1999, 39(6): 928-942.
2. Dong, X., Gilbert, K., Guha, R., Heiland, R., Kim, J., Pierce, M., Fox G., and Wild, D.: Web Service Infrastructure for Chemoinformatics. Journal of Chemical Information and Modeling. Vol. 47, pp. 1303-1307, July, 2007.
3. Fox, G. and Gannon, G., A Survey of the Role and Use of Web Services and Service Oriented Architectures in Scientific/Technical Grids", Technical Report, August 2006.
4. Taylor, I., Shields, M., Wang, I., and Harrison, A., The Triana Workflow Environment: Architecture and Applications, Workflows for eScience: Scientific Workflow for Grids, I. Taylar, E. Deelman, D. Gannon, M. Shields, eds., Springer-Verlag, 2007
5. Pennington, D., Higgins, D., Peterson, A., Jones, M., Ludascher, B., Bowers, S., Ecological Niche Modeling Using the Kepler Workflow System, Workflows for eScience: Scientific Workflow for Grids, I. Taylar, E. Deelman, D. Gannon, M. Shields, eds., Springer-Verlag, 2007.
6. Oinn, T., Li, P., Kell, D., Goble, C., Goderis, A., Greenwood, M., Hull, D., Stevens, R., Turi, D., and Zhao, J., Taverna / myGrid: aligning a workflow system with the life sciences community, Workflows for eScience: Scientific Workflow for Grids, I. Taylar, E. Deelman, D. Gannon, M. Shields, eds., Springer- Verlag, 2007.
7. Deelman, E., Mehta, G., Singh, G., Su, M., Vahi, K., Pegasus: Mapping Large-Scale Workflows to Distributed Resources, Workflows for eScience: Scientific Workflow for Grids, I. Taylar, E. Deelman, D. Gannon, M. Shields, eds., Springer-Verlag, 2007.
8. Fahringer, T., Prodan, R., Hofer, J., Nadeem, F., Nerieri, F., Podlipnig, S., Qin, J., Rubing, D., ,Siddiqui, M., Truong, H., Villazon, A., Wieczorek, M., ASKALON: A Development and Grid Computing Environment for Scientific Workflows, Workflows for eScience: Scientific Workflow for Grids, I. Taylar, E. Deelman, D. Gannon, M. Shields, eds., Springer-Verlag, 2007.
9. Brandic, I., Pllana, S. and Benkner, S., High-level Composition of QoSaware Grid Workflows: An Approach that Considers Location Affinity, Proceedings of WORKS06, Paris, 2006.
10. Slominski, A., Adapting BPEL to Scientific Workflows, Workflows for eScience: Scientific Workflow for Grids, I. Taylar, E. Deelman, D. Gannon, and M. Shields, eds., Springer-Verlag, 2007.
11. MyGrid, http:// www.mygrid.org.uk/.
12. ToxTree, http:// sourceforge.net/projects/toxtree.
13. Taverna, http://taverna.sourceforge.net.
14. Weininger, D. (1988), SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules, J. Chem. Inf. Comput. Sci. 28, 31-36.
15. W. L., Wilfred, Krishnan, S., Mueller, K., Ichikawa, K., Date, S., Dallakyan, S., Sanner, M., Misleh, C., Ding, Z., Wei, X., Tatebe, O., and Arzberger, P. W., Building Cyberinfrastructure for Bioinformatics Using Service Oriented Architecture, CCGrid, p.39, Sixth IEEE International Symposium on Cluster Computing and the Grid Workshops (CCGRIDW'06), 2006.
16. Kandaswamy, G., Fang, L., Huang, Y., Shirasuna, S., Marru, S. and Gannon., D., Building Web Services for Scientific Grid Applications. IBM Journal of Research and Development, 50(2/3):249-260, 2006.