e-BioFlow: Improving Practical Use of Workflow Systems in Bioinformatics

Ingo Wassink\textsuperscript{1,6}, Matthijs Ooms\textsuperscript{1,6}, Pieter Neerincx\textsuperscript{2,6}, Gerrit van der Veer\textsuperscript{1,3,6}, Han Rauwerda\textsuperscript{4,6}, Jack A.M. Leunissen\textsuperscript{5,6}, Timo M. Breit\textsuperscript{4,6}, Anton Nijholt\textsuperscript{1,6}, and Paul van der Vet\textsuperscript{1,6}

\textsuperscript{1} Human Media Interaction Group, University of Twente, The Netherlands
\textsuperscript{2} Biomolecular Mass Spectrometry and Proteomics Group, Utrecht University, The Netherlands
\textsuperscript{3} Open University, Heerlen, The Netherlands
\textsuperscript{4} Microarray Department and Integrative Bioinformatics Unit, University of Amsterdam, The Netherlands
\textsuperscript{5} Laboratory of Bioinformatics, Wageningen University and Research, The Netherlands
\textsuperscript{6} Netherlands Bioinformatics Centre, Nijmegen, The Netherlands

Abstract. Workflow management systems (WfMSs) are useful tools for bioinformaticians. As experiences with using WfMSs accumulate, shortcomings of current systems become apparent. In this paper, we focus on practical issues that hinder WfMS users and that arise in the design and execution of workflows, and in access of web services. We present e-BioFlow, a workflow engine that demonstrates in which way a number of these problems can be solved. e-BioFlow offers an improved user interface, can deal with large data volumes, stores all provenance, and has a powerful provenance browser. e-BioFlow also offers the possibility to design and run workflows step by step, allowing its users an explorative research style.

1 Introduction

Today, workflow management systems (WfMSs) are recognised as useful tools for chaining computational tasks\cite{12} and in particular for orchestrating web services\cite{34}. Open-source WfMSs for scientific computation (e.g., Kepler\cite{5} and Triana\cite{6}) and specifically for bioinformatics (e.g., Taverna\cite{7}) enjoy worldwide use. Several companies also sell proprietary WfMSs, but the bioinformatics community tends to standardise on open-source systems.

In the course of developing several bioinformatics workflows, including a workflow for demonstrating the use of R in Taverna\cite{8} and OligoRAP (to be discussed below), we have run against a number of issues that stand in the way of smooth use of WfMSs in daily practice. Our experiences complement those of\cite{2} and\cite{4}. To demonstrate that many of the problems we found can be solved, we designed and built our own workflow system called \textit{e-BioFlow}\cite{9}. e-BioFlow pays particular attention to user interaction and provenance. e-BioFlow moreover enables...
the user to build and run workflows step by step. Bioinformaticians can experiment with their workflows until they are satisfied and only then store them. The combination of the ability to handle large data volumes, improved user interaction, and the feature of designing a workflow in an ad-hoc fashion, inspecting all data at all times, is, we believe, unique. e-BioFlow is based on an open-source workflow engine that has proven its worth in practice, YAWL [10]. e-BioFlow is itself an open-source experimentation platform.

Summarising our experiences, we found user interaction provided by existing tools cumbersome, in design, in execution, and afterwards when provenance has to be accessed. We also found it necessary to take extra measures to handle gigabytes of data flowing through the system. Running times can be long for bioinformatics workflows; most of the workflows we ran take hours to complete. This makes working with most current workflow tools laborious. One has to re-run the entire workflow when one wants to change parts of the workflow or when an unexpected termination occurs, for example the system crashes or an essential external resource proves to be unavailable. We realised that we can solve the latter problems by building an elaborate mechanism for storing and retrieving provenance, which in itself is a necessary component of any workflow system for bioinformatics. We avoided many problems associated with web services by using only BioMoby [11] services on databases that were mirrored in-house. Nevertheless, we know that web services pose problems of their own and we therefore address such problems in this paper as well. Finally, we wanted to profit from the wealth of experience with workflows in the business administration community.

A typical example of the kind of bioinformatics workflows we address in this paper is OligoRAP, used for automatic re-annotation of transcriptomics probe libraries [12][13]. A high-quality oligonucleotide probe library is an essential component of genome-wide microarray-based gene-expression experiments. In order to maintain the quality of the information associated with the probes, this information has to be updated when new sequence or annotation data is released. An OligoRAP client orchestrates BioMoby web services to automatically check the target specificity of the oligonucleotide probes and update their annotations. The client calls nucleotide alignment search tools such as BLAT [14] and BLAST [15], fetches annotations provided by the Ensembl [16] project and performs a quality assessment. An OligoRAP run is not wholly linear; for example, the BLAT service is iteratively polled for output and the BLAST service is only called under certain conditions. The result of an OligoRAP run consists of XML files that provide detailed information per probe and a quality assessment of the whole library.

## 2 Definitions

We will first lay down our terminology. A workflow is a representation of a coherent collection of several tasks to execute a procedure. At the lowest level, tasks (Taverna says “processors”) are executed in a predefined order and with a preset data exchange. This may not be obvious at a higher level, as when, for