

Generic Portlets

for Bioinformatics in a Grid Infrastructure

Sandra Gesing and Oliver Kohlbacher

Simulation of Biological Systems, Center for Bioinformatics, Eberhard Karls University Tübingen, Germany gesing@informatik.uni-tuebingen.de



Bioinformatics tools often require advanced computing skills and specific resources. To provide users without the necessary computational background with an intuitive user interface for different appli-

cations, we are developing a workflow-enabled portal for bioinformatics in a grid infrastructure.

One of the first steps to achieve our goal is the design of generic portlets. We have started with a ge-

neric mask portlet and a BPEL generator portlet. Our portlets minimize user interaction on technical adjustments and the users are able to concentrate on their specific research.

EBERHARD KARLS

Universität

TÜBINGEN

Motivation

An enormous number of complex and sophisticated algorithms and tools has been developed to aid the research in the area of bioinformatics. These tools often require specific computational resources and rather advanced computing skills for installation, administration, and use. However, scientists of disciplines like biology, biochemistry, and biomedicine want to focus on their specific research, but do not want to deal with unpleasant details of software installation, usability, and hardware configuration.

To hide complexity from the user we are developing a workflow-enabled portal for bioinformatics in a grid infrastructure. The portal is based on the open source portal software GridSphere[1] and workflows are built with WS-BPEL[2].

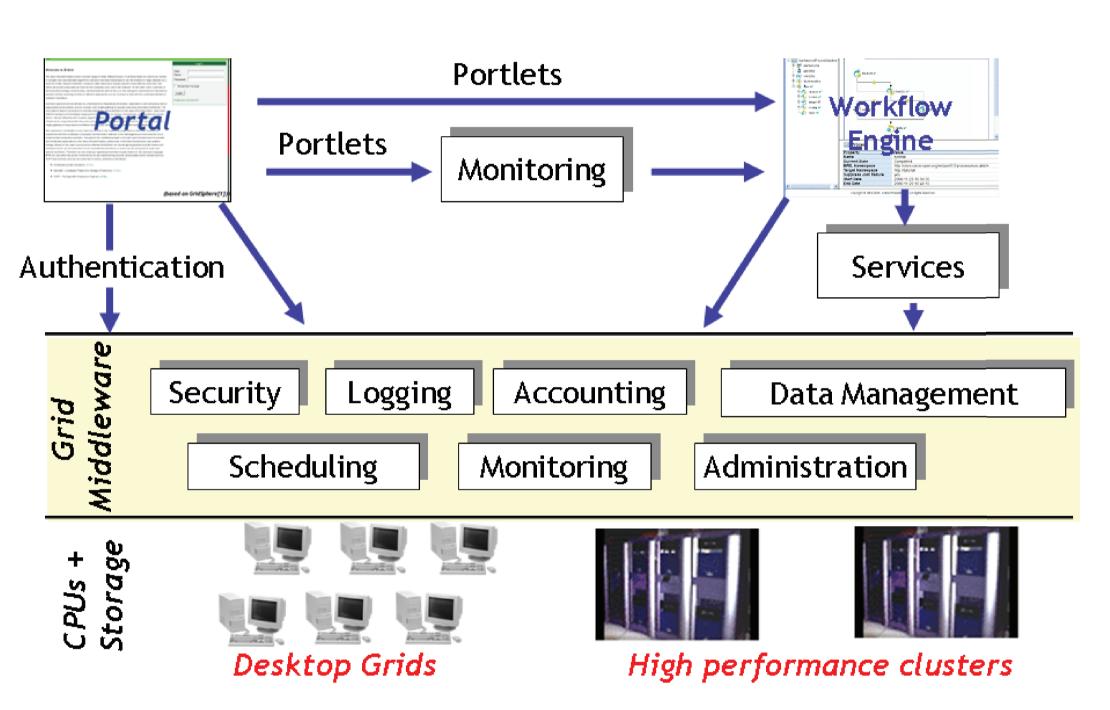


Fig.1 Architecture Overview

Basics

Portlet

A portlet is a Java program that extends the features of a portalserver, comparable with a servlet and a server.

WSDL [3]

The Web Services Description Language defines an XML standard for the description of webservices for operations and exchange of messages.

SOAP

The Simple Object Access Protocol "is a light-weight protocol intended for exchanging structured information in a decentralized, distributed environment"[4]. A SOAP message is also XML-based and can be generated automatically from a WSDL file.

WS-BPEL

The Web Services Business Process Execution Language is a commonly used XML standard for workflow design.

Generic Mask Portlet

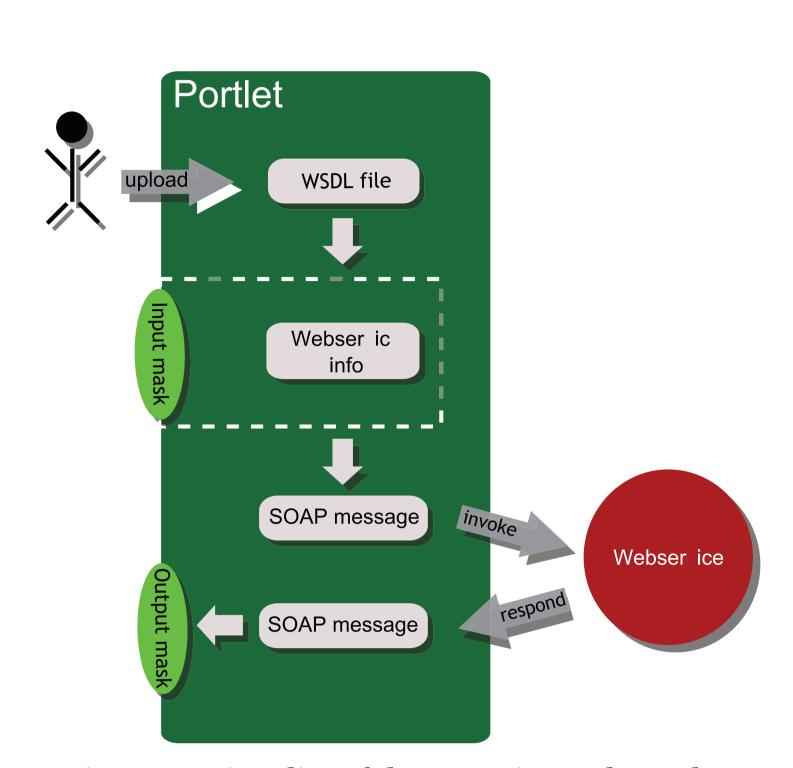


Fig.2 Functionality of the Generic Mask Portlet

Various tools have been developed for invoking webservices via HTTP using the corresponding WSDL file as basis to generate SOAP, e.g. SOAPUI [5]. By our state of knowledge the existing open-source programs support the user with an XML file for the request and deliver accordingly an XML file with the response. The user has to know how to interpret these files. Our portlet will also use the WSDL files but will generate an intuitive mask for the input of a webservice and an output mask containing the response of a webservice.

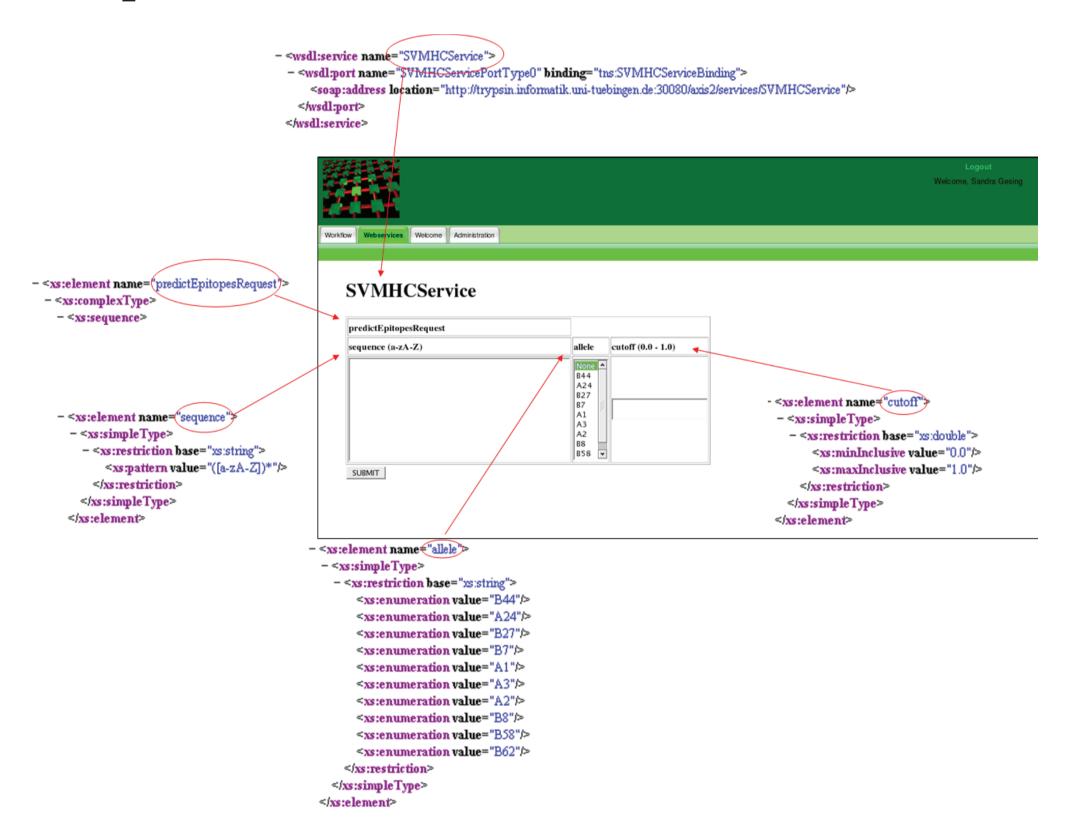


Fig.3 Input mask for the example SVMHCService[6] with the corresponding parts of the WSDL file

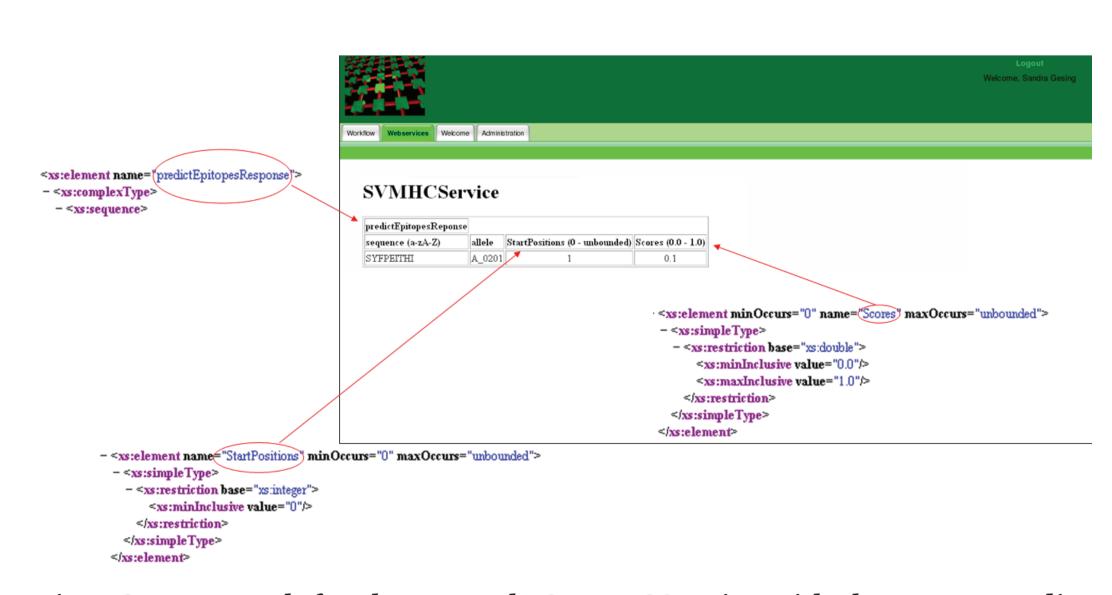


Fig.4 Output mask for the example SVMHCService with the corresponding parts of the WSDL file for the sequence "SYFPEITHI" as input

BPEL Generator Portlet

To minimize user interaction on deriving a BPEL file from a WSDL file, we will implement a BPEL generator as portlet.

This portlet will re-use the validation of WSDL files from the Generic Mask Portal. Afterwards it will generate a BPEL file exclusively with the informations given in the validated WSDL file. The resulting BPEL file describes a structural very simple workflow. The workflow consists of invoking this one webservice. We decided to embed the open source ActiveBPEL engine[7] as workflow engine in our portal.

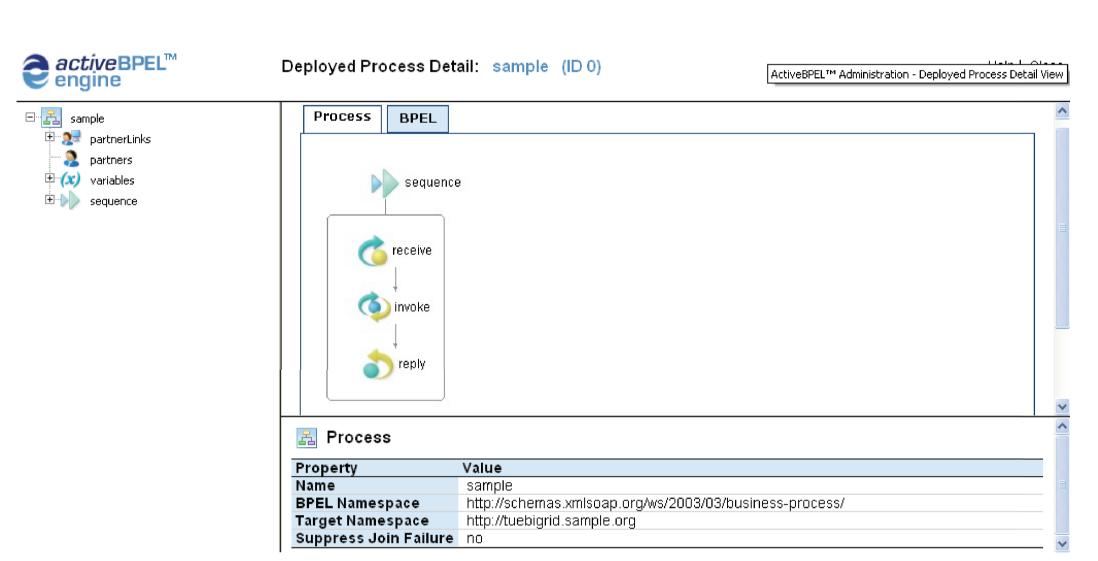


Fig.6 Example Workflow for invoking SVMHCService (view in the admin console of the ActiveBPEL engine)

A workflow in ActiveBPEL is characterized by being itself in turn a webservice. For generating this, we need to extend the given WSDL file with information about partner links and the workflow port. The partner links are bound to the corresponding ports in the WSDL.

After this a BPEL file can be created which needs the following information from the WSDL file:

- partner link name
- variable with message type
- operation with variable and port



Fig.5 BPEL file for invoking SVMHCService

Before the BPEL and the WSDL file can be deployed into the workflow engine for enactment, ActiveBPEL expects to additionally get a PDD file (Process Deployment Descriptor) and a wsdlCatalog file. Both can be created from the information in the WSDL file and the generated information in the BPEL file.

References

- [1] GridSphere portal framework http://www.gridsphere.org
- [2] WS-BPEL specification http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.o.pdf
- [3] WSDL specification http://www.w3.org/TR/wsdl
- [4] SOAP specification http://www.w3.org/TR/2007/REC-soap12-parto-20070427/
- [5] SOAPUI http://www.soapui.org/
- [6] P. Dönnes and O. Kohlbacher. SVMHC: a server for prediction of MHC-binding peptides. Nucleic Acids Res, 34: W194--W197, 2006.
- [7] ActiveBPEL engine http://www.active-endpoints.com/active-bpel-engine-overview.htm