Multimedia-Learning in a Life Science Workflow Environment

Carsten Ullrich[1], Ruimin Shen[1], and Su-Shing Chen[2]

[1] Shanghai Jiaotong University
[2] PICB, Shanghai
ullrich_c@sjtu.edu.cn, rmshen@sjtu.edu.cn,
suchen@cise.ufl.edu

The Taverna workbench allows constructing highly complex analyses over life sciences data and computational resources. It provides access over 1000 of bioinformatic services, e.g., analysis algorithms for comparing genome sequences, and facilitates the construction of bioinformatic workflows. These workflows make tacit procedural bioinformatics explicit and as such lend themselves for being used in bioinformatics education. However, until now, no Taverna e-learning service exists. In this paper, we describe how Taverna can be used for learning and the services that need to be integrated in Taverna for that purpose. This includes a digital library of multimedia resources since multimedia, especially visualization, plays an important role in bioinformatics. Equally important is an intelligent educational service that automatically assembles learning activities and resources into a pedagogically coherent whole.

1 Introduction

The Taverna workbench [7] facilitates the composition and execution of workflows for the life sciences community and allows a biologist or bioinformatician with limited computing background to construct highly complex analyses over public and private data and computational resources. Currently, Taverna provides access over 1000 of bioinformatic services, e.g., analysis algorithms for comparing genome sequences.

Bioinformatics or, more general, biomedicine is a very rich field of multimedia information. Resources include texts, images and videos obtained by a variety of methods such as X-Ray, ultrasound and magnetic resonance devices. However, despite the fact that bioinformatics is highly multi-medial in nature, Taverna has no explicit support for multi-media resources and services.

Additionally, the workflow-based approach realized in Taverna lends itself for being used for learning. Workflows make tacit, i.e., implicit, knowledge explicit and represent skills and activities that students in bioinformatics need to learn. But again, there is no Taverna service that realizes learning support.

In this paper, we will describe what it takes to use Taverna for multimedia learning. We start with a general overview on the Taverna workbench (Section 2),

followed by a brief introduction on state-of-the-art adaptive learning technologies (Section 3), namely course generation. Section 4 then sketches Taverna multimedia services that we are currently developing: VIACIPA, a digital library service of multimedia objects, and a course generator learning service based on VIACIPA.

2   A Workflow Environment in Life Sciences

The Taverna workflow workbench environment [7] allows a user (typically a biologist or bioinformatician) to specify and execute scientific workflows. The workflows model "in silicio" experiments: they involve the combination of data and analyses made available by the research teams. In bioinformatics, this type of experiments complements lab-based experiments and allows to generate new information from the publicly available data and to form hypothesis which can be assessed in lab studies.
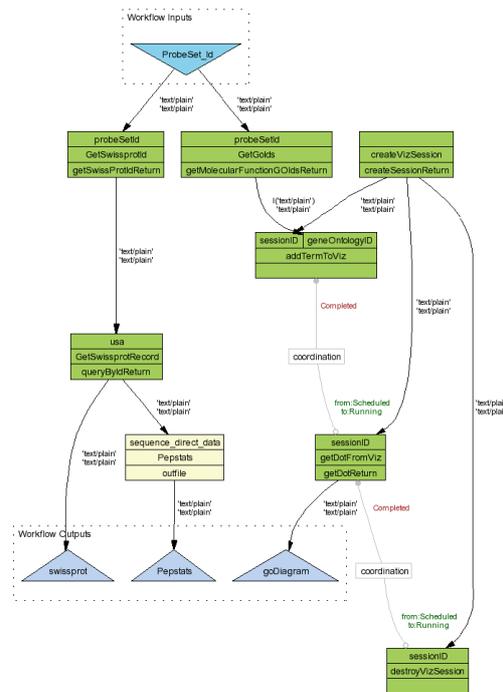

Fig. 1. An example of a Taverna workflow

Taverna integrates more than 1000 bioinformatical services, ranging from applications such as analysis algorithms for comparing sequences, over databases arising from species-specific genome projects or holding cross species data sets for proteins or nucleotides, to visualization tools for protein structures, simulations of heart excitation models. Extensibility was one major design goal in Taverna. New

services (called processors) can be easily plugged-in based on Web-service interfaces and similar means.

Workflows as realized in Taverna enable a scientist to model and execute their experiments in a repeatable and verifiable way (see Fig. 1 for an example of a workflow). Just like experiments can be repeated by other research groups, Taverna workflows can be exchanged and are resources in their own right. Thus, workflows capture the essential aspects of "in silicio" experiments and, what makes them so valuable for learning, they make tacit procedural knowledge explicit.

The components of a Taverna workflow consists of the following: a set of data inputs; a set of outputs (the exit points); a set of processors each of which represents an individual step within a workflow: a processor receives data on its input ports, processes the data internally and produces data on its output ports. The final components of a workflow are links between the data sources and/or the processors, which, for instance, specify that the output of one data source serves as the input of another one.

3 Technology-Supported Learning

Technology-enhanced learning uses computers to support the learning process. Examples range from very basic Learning Management Systems that primarily administer learning to Intelligent Learning Environments (ILE). An ILE helps the learner during the complete learning process, by selecting and annotating learning content (resources), by suggesting what resources to read etc. The ActiveMath system is an example of a state-of-the-art ILE.[1] It is a Web-based, multi-lingual, user-adaptive learning system for mathematics that operates on semantically encoded learning objects annotated with metadata for providing advanced adaptive features. One of its main components is a course generator that automatically assembles individual courses according to a learner's learning goal, learning scenario, competencies, learning context and preferences. The following section takes a closer look at the course generator.

3.1 Assembling Sequences of Learning Activities

Course generation uses pedagogical knowledge to generate structured sequences of educational activities that are adapted to the learners' competencies, individual variables, and learning goals [1]. The educational activities include viewing of learning objects but also using tools and services that support learning or that the student needs to master (e.g., computer algebra systems and function plotters for mathematic students, and bioinformatic services as available in Taverna for students of biomedicine). The generation process uses pedagogical knowledge, information about the resources and tools and, if available, information about the learner to select and sequence the resources that are to be studied and the tools to be used, and their order.

Paigos [10] is a course generator that was developed for ActiveMath. Its distinguishing features are a complex model of pedagogical knowledge and a

---

[1] www.activemath.org

service-oriented architecture. Its pedagogical knowledge allows *Paigos* to generate courses that support a learner in achieving a number of learning goals, such as discovering new content ("discover" in short), rehearsing, and training specific competencies. For these learning goals, *Paigos* generates complete courses which contain all the learning material required by a learner to achieve the goals. *Paigos* is also used to retrieve single elements that specifically fulfill a given purpose, such as presenting an example or exercise adequate for the current the learner. This functionality is important for remedial, e.g., if a learner fails to solve a task, then the presentation of an example might help the learner to overcome that difficulty.

Paigos contains about 300 rules that determine the selection, ordering and structuring of resources. This "expensive" functionality lends itself to being "outsourced": if the course generator is available as a service then other learning environments can access the functionality without having to re-implement it. Thus, *Paigos* makes its functionality available as a Web-Service (CGWS). Clients send a learning goal to the CGWS and receive a structured sequence of resource identifiers (URI) as a result. The sequence is represented using a standard representation called IMS Manifest. The CGWS implements a mediator architecture that enables a client to easily register its own repository and thus make it available to *Paigos*. At registration time, a client has to provide the name and the location of its repository, together with an ontology that describes the metadata structure used in the repository and a mapping of the metadata used in the CG onto the repository ontology. The metadata in *Paigos* consists of an ontology of instructional objects that describes LO from a pedagogical point of view sufficiently precise in order to allow for intelligent automatic pedagogical services.

After receiving a learning goal, *Paigos* automatically assembles the sequences of resources, without any human intervention. However, this requires that the resources are annotated in a way that allows for pedagogical reasoning. This means that the resource repository needs to able to answer questions about the existence of resources, e.g., whether there exists an example of a domain concept. The following section will further elaborate on the required metadata.

3.2   Requirements on Metadata

Educational services need to have access to information about the resources that are to be included in the learning process. This information is called metadata. Metadata describes characteristics of resources relevant to the application domain the resources are used in.

The most basic and most relevant aspect of a resource is the domain concept it describes (or explains, exercises, defines). This information can be given as simple keywords or as pointer to concept defined in a separate concept space, which contains the domain concepts additionally annotated with relationships among them, such as "prerequisite-of", "part-of", etc.

In the case of educational services, a widely employed standard is LOM [2], which includes metadata such as difficulty, typical learning time, etc. However, LOM was primarily designed for human actors, e.g., authors and teachers. It does not describe resources precise enough for pure software services, i.e., services that automatically perform operations on it, such as course generation. For instance, in LOM it is not

possible to express that a resource is of the type "example". An alternative metadata vocabulary is defined by the OIO (Ontology of Instructional Objects [9][10]. The OIO defines a set of about 20 classes and several relationships that allow representing the "instructional semantics" of resources. The classes include "definition", "law", "process", "interactivity", "exercise", "evidence", "example", etc.

Ideally, resources are described using the OIO and LOM. The more precise the metadata, the higher the quality of service that tools like a course generator can offer.

Despite the fact that *Paigos* was developed for the ActiveMath system, which is primarily used for mathematics, the pedagogical knowledge formalized within *Paigos* is independent of the mathematical domain but can be applied to other domains as well. Previous work investigated the usage of *Paigos* within a workflow-oriented proactive delivery of educational resources in order to support learning at an office workplace [8]. In the following, we will explore what it takes to use *Paigos* within a truly multimedia environment for bioinformatics.

4   Multimedia Services in Taverna

The current version of Taverna is focused on bioinformatical services. Bioinformatics or, more general, biomedicine is a very rich field of multimedia information (see Figure 2 for examples). Resources include texts, images and videos obtained by a variety of methods such as X-Ray, ultrasound and magnetic resonance devices. However, despite the fact that bioinformatics is highly multi-medial in nature, Taverna has no explicit support for multi-media resources and services.

In the following, we will first describe a digital library for fusion of multimedia information resources that can be used as an important service for mediating multimedia information in Taverna. We then describe how the resources can be used for learning purposes.

4.1   Digital Library Server for Fusion of Multimedia Information

In [2,3], a digital library server for fusion of multimedia information has been developed that addresses the problem that biomedical information has disparate information sources. In [2], Chen used an ontology-based approach for fusion of different multimedia sources, such as the Gene Ontology (GO), Clinical Bioinformatics Ontology (CBO) and the Foundational Model of Anatomy (FMA). An even higher level integration was developed in [3] which handled various system level issues. Ontology deals with concept level issues of the information content. For examples, anatomy deals with the human body, while cells are substructures within the organ structures of the human body. Ontology relates these concepts in a tree like structures. However multimedia information are different representations of possibly the same concept. For examples, a single organ can be imaged by both ultrasound and X-ray media. Thus we need to address the integration issue of these two kinds of information. In [2], Chen has addressed this issue, which has been a key topic of the multimedia research community.
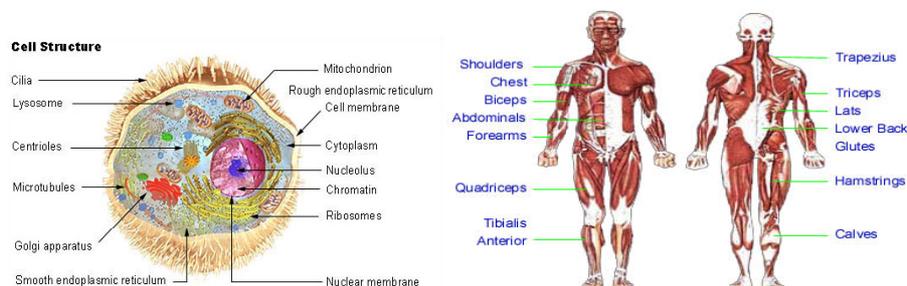
Fig. 2. Example of multimedia information for bioinformatics

4.2   Learning in a Workflow Environment

In this section, we make two suggestions of how these multimedia resources can be used for learning within a workflow environment as exemplified by Taverna. The first approach corresponds to the "traditional" usage of course generation, while the latter is an innovative approach directly based on workflows and the embedded services.

4.2.1   Traditional Course Generation

In this approach, the course generator takes a learning goal (given by the learner or set by a teacher) and assembles a sequence of resources and tool usages that supports the learner in achieving the goal.

In an open environment as Taverna, which is not specifically geared towards learning, the quality of the educational resource metadata will differ significantly. This reflects on the educational services: they have to be able to cope with such limited metadata, but at the same time use good metadata, if available. We developed a specific type of course for these situations, called "guided tour". The guided tour is based on guidelines from instructional design [6] and multimedia learning [5] and is structured as follows. For each concept given in the goal task, and for each unknown prerequisite concept, the following sections are created:

- **Introduction.** This section arises a learner's interest by presenting educational resources of the type introduction.
- **Problem.** This section inserts a real world problem for the concept.
- **Concept.** This section presents the concept.
- **Explanation.** This section contains educational resources that provide explaining and deepening information about the concept.
- **Illustration.** This section provides opportunities for the learner to examine demonstrations of applications of the concepts.
- **Practice.** This section enables a student to actively apply what he has learned about the concept.
- **Conclusion.** This section presents educational resources that contain concluding information about the concept.
- **Reflection.** This section provides the learner with an opportunity to reflect and discuss his new knowledge.

If no resource was found for a specific section, then the section is skipped. As an example, we will assume *A*, *B* and *C* are concepts and that concept *A* is a prerequisite of concept *B*. If the learning goal is *(guidedTour (B,C))*, then the resulting course will consists of three sections, one for each concept *A*, *B*, and *C,* and each of these sections will consist of up to 8 subsections corresponding to the list above. The course generator will automatically retrieve the prerequisites, the concept *A* in the above example.

From the viewpoint of the course generator, the services that are integrated in Taverna are modeled using resources stored in the library. The specific type of such a service resource depends on how they are used. If they are used for illustrative purposes, e.g., showing how to access a service and what results to expect, then they would be of type example. If the user should use them interactively to come up with a specific result, then they would be classified as exploration. For each of these service usages, a resource needs to be created.

The result of the course generation is a structured sequence of links to resources. They can be presented in Taverna's enactor invocation window.

4.2.2 Workflow Instantiation

As elaborated in Section 2, workflows form an integral part of the knowledge of an experienced biologist and bioinformatician. Workflows make tacit procedural knowledge about experiments and data analyses explicit and thus offer a way of communicating this knowledge to students. Taverna treats workflows as resources in their own rights and as such they can serve to guide learning processes. Recall that a workflow essentially consists of input and output nodes and of processor nodes that perform operations on data, such as multiple sequence alignment. For learning, a processor node can be seen as an example of the application of the service it represents. Thus, when studying a workflow, a learner should be able to select a processor node and request educational resources that elaborate on this service and its specific operation. This requires a corresponding set of resources that are annotated with the service they illustrate. Resources can thus play a double role. They describe a specific phenomenon and at the same time they are examples of the usage of the service they were created with.

Since the resources are annotated with the service name, each usage of this specific service can be linked to the corresponding resource, regardless of the workflow. Take as an example a workflow that takes in a protein sequence and then runs the InterProScan service to find family domians.[2] If the resources are annotated with the service name (*InterProScan_proteinraw*), then the learner can request information about this service in each workflow that uses it. In this way, a workflow is annotated with additional information that may help to understand it, and in addition, a workflow is an example of how to use services.

This type of learning support will benefit significantly from multimedia resources: since here the goal is to give examples of tool usage, presenting video sequences allows illustrating the complete process of the usage, including the results.

---

[2] Workflow example by Paul Fisher (http://www.cs.man.ac.uk/~fisherp/).

5   Conclusion

In this paper, we described how to employ multimedia services to a workflow environment used in bioinformatics. We sketched two services that add a new layer of functionality to the Taverna workflow workbench: VIACIPA, a web-based digital library service of multimedia objects, and *Paigos*, a course generator that assembles sequences of resource with the goal of supporting the learning process. *Paigos* can be used in Taverna in two ways: in the traditional, course-based way, but also within a workflow to illustrate the different processes employed in the workflow.

References

[1] Brusilovsky, P., Vassileva, J.: Course sequencing techniques for large-scale webbased education. International Journal of Continuing Engineering Education and Lifelong Learning 13(1/2), 75–94 (2003)

[2] Chen, S.: Fusion of Multimedia Information in Biomedicine, this Conference

[3] Kim, H., Choo, C., Chen, S.: Generating a Meta-DL by Federating Search on OAI and Non-OAI Servers. Journal of Intelligent Systems

[4] IEEE Learning Technology Standards Committee. 1484.12.1-2002 IEEE standard for Learning Object Metadata (2002)

[5] Mayer, R.: Multimedia Learning. Cambridge University Press, New York (2001)

[6] Merrill, M.D.: First principles of instruction. Educational Technology Research & Development 50(3), 43–59 (2002)

[7] Oinn, T., Greenwood, M., Addis, M., Alpdemir, M.N., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., Li, P., Lord, P., Pocock, M.R., Senger, M., Stevens, R., Wipat, A., Wroe, C.: Taverna: Lessons in creating a workflow environment for the life sciences in Concurrency and Computation: Practice and Experience. Grid Workflow Special Issue 18(10), 1067–1100 (2005)

[8] Rostanin, O., Ullrich, C., Holz, H., Song, S.: Project TEAL: Add adaptive e-learning to your workflows. In: Tochtermann, K., Maurer, H. (eds.) Proceedings: I-KNOW 2006. 6th International Conference on Knowledge Management, Graz, Austria, pp. 395–402 (September 2006)

[9] Ullrich, C.: The learning-resource-type is dead, long live the learning- resource-type! Learning Objects and Learning Designs 1(1), 7–15 (2005)

[10] Ullrich, C.: Course Generation as a Hierarchical Task Network Planning Problem. Unpublished Doctoral Thesis, Department of Computer Science, Saarland University (2007)