Erhard Rahm (Ed.)

LNBI 2994

Data Integration in the Life Sciences

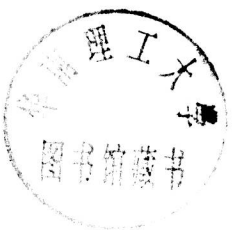
First International Workshop, DILS 2004
Leipzig, Germany, March 2004
Proceedings

Springer

Erhard Rahm (Ed.)

Data Integration in the Life Sciences

First International Workshop, DILS 2004
Leipzig, Germany, March 25-26, 2004
Proceedings



Springer

Volume Editor

Erhard Rahm
Universität Leipzig, Institut für Informatik
Augustusplatz 10-11, 04109 Leipzig, Germany
E-mail: rahm@informatik.uni-leipzig.de

Library of Congress Control Number: 2004102414

CR Subject Classification (1998): H.2, H.3, H.4, J.3

ISSN 0302-9743
ISBN 3-540-21300-7 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Boller Mediendesign
Printed on acid-free paper SPIN: 10994115 06/3142 5 4 3 2 1 0

Lecture Notes in Bioinformatics 2994

Edited by S. Istrail, P. Pevzner, and M. Waterman

Subseries of Lecture Notes in Computer Science

Springer
Berlin
Heidelberg
New York
Hong Kong
London
Milan
Paris
Tokyo

Preface

DILS 2004 (Data Integration in the Life Sciences) is a new bioinformatics workshop focusing on topics related to data management and integration. It was motivated by the observation that new advances in life sciences, e.g., molecular biology, biodiversity, drug discovery and medical research, increasingly depend on bioinformatics methods to manage and analyze vast amounts of highly diverse data. Relevant data is typically distributed across many data sources on the Web and is often structured only to a limited extent. Despite new interoperability technologies such as XML and web services, integration of data is a highly difficult and still largely manual task, especially due to the high degree of semantic heterogeneity and varying data quality as well as specific application requirements.

The call for papers attracted many submissions on the workshop topics. After a careful reviewing process the international program committee accepted 13 long and 2 short papers which are included in this volume. They cover a wide spectrum of theoretical and practical issues including scientific/clinical workflows, ontologies, tools/systems, and integration techniques. DILS 2004 also featured two keynote presentations, by Dr. Thure Etzold (architect of the leading integration platform SRS, and president of Lion Bioscience, Cambridge, UK) and Prof. Dr. Svante Pääbo (Director, Max Planck Institute for Evolutionary Anthropology, Leipzig).

The workshop took place during March 25–26, 2004, in Leipzig, Germany, and was organized by the Interdisciplinary Bioinformatics Center (IZBI) of the University of Leipzig. IZBI was founded in 2002, and became one of five German bioinformatics centers funded by the German Research Association (DFG) after a highly competitive selection process. The University of Leipzig, founded in 1409, has chosen biotechnology as one of its high priority fields. Its new Center for Biotechnology and Biomedicine was established in 2003 and closely cooperates with IZBI and other research institutes as well as with industrial partners.

As the workshop chair and editor of this volume, I would like to thank all authors who submitted papers, as well as the program committee members and additional referees for their excellent work in evaluating the submissions. Special thanks go the Max Planck Institute for Evolutionary Anthropology for providing us with their new facilities to run the workshop, and the IZBI organization team, in particular Hai Hong Do, Toralf Kirsten and Hans Binder. Finally, I would like to thank Alfred Hofmann and his team at Springer-Verlag for their cooperation and help in putting this volume together.

Leipzig, February 2004

Erhard Rahm
Workshop Chair

International Workshop on Data Integration in the Life Sciences (DILS 2004)

Workshop Chair

Erhard Rahm	University of Leipzig, Germany

Program Committee

Howard Bilofsky	GlaxoSmithKline, USA
Terence Critchlow	Lawrence Livermore National Laboratory, USA
Peter Gray	University of Aberdeen, UK
Barbara Heller	University of Leipzig, Germany
Ralf Hofestaedt	University of Bielefeld, Germany
Jessie Kennedy	Napier University, Edinburgh, UK
Ulf Leser	Humboldt University, Berlin, Germany
Bertram Ludäscher	San Diego Supercomputer Center, USA
Sergey Melnik	Microsoft Research, USA
Peter Mork	University of Washington, Seattle, USA
Felix Naumann	Humboldt University, Berlin, Germany
Frank Olken	Lawrence Berkeley National Laboratory, USA
Norman Paton	University of Manchester, UK
Erhard Rahm	University of Leipzig, Germany
Louiqa Raschid	University of Maryland, USA
Kai-Uwe Sattler	Technical University Ilmenau, Germany
Steffen Schulze-Kremer	Freie Universität Berlin, Germany
Robert Stevens	University of Manchester, UK
Sharon Wang	IBM Life Sciences, USA
Limsoon Wong	Institute for Infocomm Research, Singapore

Additional Reviewers

Jens Bleiholder	Humboldt University, Berlin, Germany
Shawn Bowers	San Diego Supercomputer Center, USA
Hong-Hai Do	University of Leipzig, Germany
Ulrike Greiner	University of Leipzig, Germany
Efrat Jaeger	San Diego Supercomputer Center, USA
Kai Lin	San Diego Supercomputer Center, USA
Toralf Kirsten	University of Leipzig, Germany
Angela Stevens	Max Planck Institute for Mathematics in the Sciences, Leipzig, Germany
Melanie Weis	Humboldt University, Berlin, Germany

Sponsoring Institutions

Max Planck Institute for Evolutionary Anthropology, Leipzig, Germany
http://www.eva.mpg.de/

Interdisciplinary Center for Bioinformatics, University of Leipzig, Germany
http://www.izbi.de/

Organization Committee

Erhard Rahm	University of Leipzig, Germany
Hong-Hai Do	University of Leipzig, Germany
Toralf Kirsten	University of Leipzig, Germany
Hans Binder	University of Leipzig, Germany

Website

For more information on the workshop please visit the workshop website under the URL http://www.izbi.de/dils04/

Lecture Notes in Bioinformatics

Vol. 2994: E. Rahm (Ed.), Data Integration in the Life Sciences. X, 221 pages. 2004.

Vol. 2983: S. Istrail, M.S. Waterman, A. Clark (Eds.), Computational Methods for SNPs and Haplotype Inference. IX, 153 pages. 2004.

Vol. 2812: G. Benson, R.D.M. Page (Eds.), Algorithms in Bioinformatics. X, 528 pages. 2003.

Vol. 2666: C. Guerra, S. Istrail (Eds.), Mathematical Methods for Protein Structure Analysis and Design. XI, 157 pages. 2003.

Table of Contents

Scientific and Clinical Workflows

Ontologies and Taxonomies

Indexing and Clustering

Integration Tools and Systems

Integration Techniques

An Ontology-Driven Framework for Data Transformation in Scientific Workflows*

Shawn Bowers and Bertram Ludäscher

San Diego Supercomputer Center
University of California, San Diego
La Jolla, CA, 92093-0505, USA,
{bowers, ludaesch}@sdsc.edu

Abstract. Ecologists spend considerable effort integrating heterogeneous data for statistical analyses and simulations, for example, to run and test predictive models. Our research is focused on reducing this effort by providing data integration and transformation tools, allowing researchers to focus on "real science," that is, discovering new knowledge through analysis and modeling. This paper defines a generic framework for transforming heterogeneous data within scientific workflows. Our approach relies on a formalized ontology, which serves as a simple, unstructured global schema. In the framework, inputs and outputs of services within scientific workflows can have structural types and separate semantic types (expressions of the target ontology). In addition, a *registration mapping* can be defined to relate input and output structural types to their corresponding semantic types. Using registration mappings, appropriate data transformations can then be generated for each desired service composition. Here, we describe our proposed framework and an initial implementation for services that consume and produce XML data.

1 Introduction

For most ecological and biodiversity forecasting, scientists repeatedly perform the same process: They select existing datasets relevant to their current study, and then use these datasets as input to a series of analytical steps (that is, a *scientific workflow*). However, ecological and biodiversity data is typically heterogeneous. Researchers must spend considerable effort integrating and synthesizing data so that it can be used in a scientific workflow. Reusing analytical steps within workflows involves a similar integration effort. Each analytic step (or *service* since they can be implemented as web services [CCMW01]) in a workflow consumes and produces data with a particular structural representation, much like a dataset. To compose existing services, the structural and semantic differences between the services must be resolved, and this resolution is typically performed by the scientist either manually or by writing a special-purpose program or script.

* This work supported in part by the National Science Foundation (NSF) grants ITR 0225676 (SEEK) and ITR 0225673 (GEON), and by DOE grant DE-FC02-01ER25486 (SciDAC-SDM).

E. Rahm (Ed.): DILS 2004, LNBI 2994, pp. 1–16, 2004.

The Science Environment for Ecological Knowledge (SEEK)[1] [Mic03] is a multidisciplinary effort aimed at helping scientists discover, access, integrate, and analyze distributed ecological information. We envision the use of web-enabled repositories to store and access datasets, including raw data and derived results, software components, and scientific workflows. Additionally, we wish to exploit formalized, ecological ontologies to help scientists discover and integrate datasets and services, as workflows are designed and executed.

This paper proposes a framework that exploits ontological information to support structural data transformation for scientific workflow composition. We believe data transformation is an integral part of *semantic mediation*, which aims at providing automated integration services within SEEK. The framework is designed to allow researchers to easily construct scientific workflows from existing services, without having to focus on detailed, structural differences. In particular, when a service is stored in SEEK, each input and output is annotated with a structural and (optionally) a semantic type. In our framework, a structural type—similar to a conventional programming-language data type—defines the allowable data values for an input or output, whereas a semantic type describes the high-level, conceptual information of an input or output, and is expressed in terms of the concepts and properties of an ontology. Thus, although structurally different, two services may still be semantically compatible based on their semantic types.

The goal of the framework is to exploit semantic types to (semi-) automatically generate mappings between services with heterogeneous structural types. By defining input and output *registration mappings*, which link a structural type to a corresponding semantic type, ontological information is used to inform data transformation. When a scientist wishes to compose two services, the input and output registration mappings are combined to create a correspondence between the two structural types. The correspondence is then used, when possible, to generate the desired data transformation, thus making the services structurally compatible.

The rest of this paper is organized as follows. Section 2 briefly describes scientific workflows and introduces an example workflow used throughout the paper. Section 3 defines our proposed framework for data transformation. Section 4 describes an initial implementation of the framework for services that exchange XML data. In particular, we define a language for specifying registration mappings, based on structural types expressed using XML Schema. Section 5 discusses related work. Section 6 concludes with a discussion of future work.

2 Scientific Workflows

SEEK extends the Ptolemy system [BCD+02] to support scientific workflows. Ptolemy is an open-source, Java-based application that provides interfaces for designing and executing data-flow process networks [LP95]. In particular, we

[1] See http://seek.ecoinformatics.org/

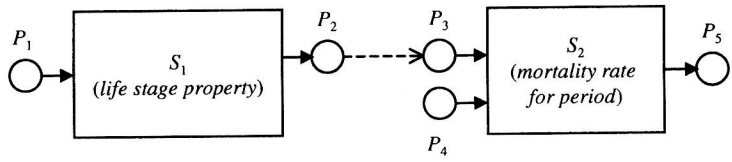


Fig. 1. Two connected services to calculate the k-value [BHT96] during a particular life-stage period.

are extending Ptolemy to support web-services, web-enabled repositories and workflow execution, scientific datasets, and semantic-type annotations through ontologies. This section briefly describes scientific workflows and introduces our running example. We note that our definition of a scientific workflow is inspired by, and compatible with, the dataflow models of Ptolemy.

We define a *scientific workflow* as a set of connected services. By *service*, we mean any software component that takes input and produces output, including but not limited to web services. A service has zero or more uniquely named *ports*. A port serves as either an input or an output to a service. Services exchange information using *connections*, which link an output port to one ore more input ports. When a pipeline is executed, data is transferred via connections from output to input ports according to an execution model. An execution model is an algorithm that determines how services should be scheduled and how data should flow through services.

Figure 1 depicts a simple workflow that contains two services S_1 and S_2, where the output port P_2 of S_1 is connected to the input port P_3 of S_2. The purpose of this simple workflow is to compute mortality rates for periods within the lifecycle of an organism [BHT96]. For example, consider the two datasets shown in Figure 2. The table on the left, Figure 2(a), gives population samples for specific development phases of the common field grasshopper (taken from Begon, *et al* [BHT96]). The dataset on the right, Figure 2(b), gives periods of development defined in terms of phases. Note that only one such period is given. Here, S_2 applies the "killing power," or k-value statistic to determine the rate of mortality for a set of observations (given in P_3) and a set of phases (given in P_4). For the datasets in Figure 2, S_2 would output a single pair (Nymphyl, 0.44) on port P_5. We note that, in SEEK, S_1 and S_2 represent stand-alone services that would be selected by an ecologist from a repository and connected, possibly as part of a larger workflow.

We require each port to have a *structural type*, which is similar to a conventional data type found in programming languages. In general, a structural type is expressed according to a *type system*, which consists of a set of base types, a description of the allowable types (of the type system), and rules defining when one type is a subtype of another. A type definition is a restriction on the structure of values being produced or consumed by a port. Thus, any value

(a)

Phase	*Observed*
Eggs	44,000
Instar I	3,513
Instar II	2,529
Instar III	1,922
Instar IV	1,461
Adults	1,300

(b)

Period	*Phases*
Nymphal	{Instar I, Instar II, Instar III, Instar IV}

Fig. 2. Example datasets for computing k-values during lifecycle periods.

that conforms to the structural type (or a subtype of the structural type) is an allowed value for the port.

We assume the function $structType(P)$ is well defined for each port P and returns the structural type of P. Given two ports P_a and P_b, we write $P_a \preceq P_b$ to denote that the structural type of P_a is a subtype of the structural type of P_b. If $P_a \preceq P_b$, we say P_a is *structurally compatible* with P_b.

One of the goals of SEEK is to allow scientists to reuse existing services when building new ecological models. In general, we assume services are not "designed to fit." That is, two distinct services may produce and consume compatible information, but have incompatible structural types.

3 Ontologies for Data Transformation

In this section, we define our proposed framework for ontology-driven data transformation. We first describe the use of semantic-type annotations in SEEK for enriching workflow services. We then describe how semantic types are exploited in our framework via registration mappings, which are used to generate data transformations between structurally incompatible services.

3.1 Ontologies and Services

As part of SEEK, we are developing technology that lets ecologists define, manage, and exploit ontologies. In general, we permit scientists to construct domain or project-specific ontologies and define mappings between them, when appropriate. In addition, we are developing, with the help of ecologists, an upper-level ecological ontology to serve as a framework for incorporating the various domain-specific ontologies. The upper-level ontology includes concepts and relationships for ecological properties, methods, measurements, and taxonomies. In the rest of this paper, we assume a single global ontology, however, in our envisioned environment there will more likely be many ontologies that when combined, for example, through mappings, form a single, global ontology.

A *semantic type* is defined using the concepts and properties of the ontology. In SEEK, we use semantic types to annotate services and datasets, where a semantic-type annotation defines the conceptual information represented by the

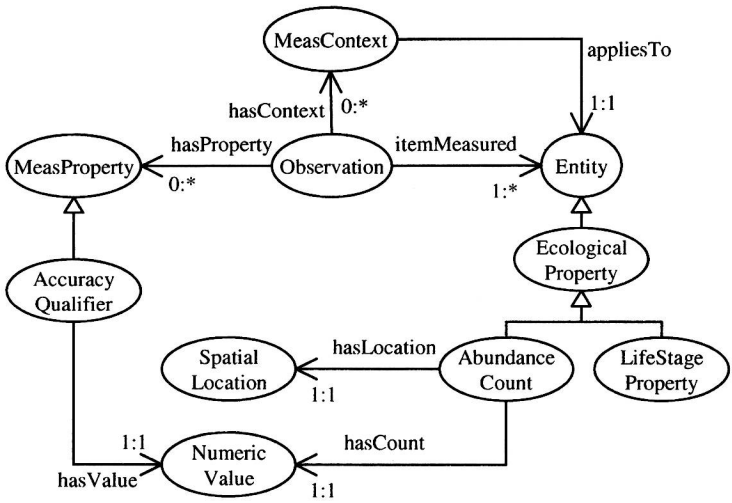


Fig. 3. Portion of a SEEK ontology for ecological measurements.

item. A semantic type defines the conceptual information that is either consumed or produced by a port. Thus, semantic types for ports are similar to semantic pre- and post-conditions in DAML-S [ABH+02]. We note that in SEEK, however, semantic types are independent of the structural details of a port. We believe decoupling structural and semantic types has the following advantages. First, although we require structural types, semantic types are optional. Thus, a service can still be used even if it is not fully specified. Second, a port's semantic type can be defined or updated after the service is deployed, without requiring changes to the structural type. Finally, we believe semantic types are easier to specify if they do not mix structural information. For example, a port's semantic type may be as simple as a single concept label, even though it has a complex structural type.

We have adopted the Web Ontology Language (OWL) [MvH03] to express ontologies in SEEK. Figure 3 shows a fragment of the SEEK measurement ontology. The ontology is represented graphically, using RDF-Schema conventions [BG03]. Only a subset of the OWL constructs are used in Figure 3. In particular, we only consider class and property definitions, subclass relationships, and cardinality constraints on properties. (The notation 0:* and 1:* represent value and existential restrictions, respectively, and 1:1 represents an exactly-one number restriction [BN03].) Similarly, Figure 4 gives the semantic types for ports P_2 and P_3 of Figure 1. As shown, the semantic type of port P_3 accepts observations, each of which measures an abundance count within the context of a life-stage property. The semantic type of P_2 outputs similar observations, but with accuracy qualifiers. In general, the semantic type of a port is defined as an OWL concept.

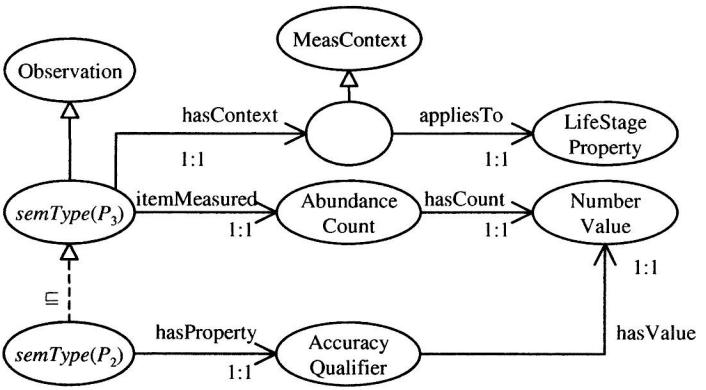


Fig. 4. Example semantic types for output and input ports P_2 and P_3.

We assume the function $semType(P)$ returns the semantic type of a port P. As stated above, a semantic type denotes a concept, which either exists as, or is a restricted subclass of, a concept within an ontology. As shown in Figure 4, $semType(P_2)$ is a subtype of $semType(P_3)$, which we denote using the standard $\sqsubseteq$ relation used in description logics. Intuitively, $P_2 \sqsubseteq P_3$ holds if every instance of the concept $semType(P_2)$ is also an instance of the concept $semType(P_3)$.

3.2 Connecting Semantically Compatible Services

To correctly compose two services, a valid connection must be defined between an output port of the source service and an input port of the target service. Intuitively, a desired connection (see Figure 5) is valid if it is both semantically and structurally valid. A connection from port P_s to port P_t is *semantically valid* if $P_s \sqsubseteq P_t$ and *structurally valid* if $P_s \preceq P_t$. The connection is *structurally feasible* if there exists a structural transformation δ such that $\delta(P_s) \preceq P_t$.

Our focus here is on the situation shown in Figure 5. Namely, that we have semantically valid connections that are not structurally valid, but feasible. Thus, the goal is to find a δ that implements the desired structural transformation.

3.3 Structural Transformation Using Registration Mappings

Figure 6 shows our proposed framework for data transformation in scientific workflows. The framework uses *registration mappings*, which consists of a set of rules that define associations between a port's structural and semantic types. In this paper, we use registration mappings to derive data transformations. A rule $q \leftrightarrow p$ in a registration mapping associates data objects identified with a query q to concepts identified with a concept expression p. For example, q may expressed as an XQuery [BCF+03] for XML sources or as an SQL query for relational sources, selecting a set of objects belonging to the same concept