Using Web Services and Scientific Workflow for Species Distribution Prediction Modeling[1]

Jianting Zhang, Deana D. Pennington, and William K. Michener

LTER Network Office, the University of New Mexico,
MSC 03 2020, 1 University of New Mexico,
Albuquerque, NM, 87131, USA
jzhang@lternet.edu

Abstract. Species distribution prediction modeling plays a key role in biodiversity research. We propose to publish both species distribution data and modeling components as Web services and composite them into modeling systems using the scientific workflow approach. We build a prototype system using Kepler scientific workflow system and demonstrate the feasibility of the proposed approach. This study is the first step towards building a virtual e-science laboratory for ecologists to perform distributed and cooperative research on species distribution predictions.

1 Introduction

The Convention on Biological Diversity (CBD) agreement signed at the United Nations Conference on Environment and Development (UNCED) held in Rio De Janeiro 1992 is a milestone towards conservation and sustainable use of biological diversity (biodiversity). Identifying species distribution and understanding global patterns of global biodiversity have become key issues in developing conservation strategies and policies [1]. During the past years, there is a massive development of biodiversity related information systems on the Internet [2] and the available species distribution data has been increased dramatically [3]. Meanwhile, considerable amount of species distribution models have been developed [4]. There are increasing needs for species distribution modeling ranging from basic ecological and biogeography research to routine conservation practices. On the other hand, most current software implementations of species distribution prediction models are developed by domain scientists which only accept ad-hoc data formats and run locally on a single machine.

With the emerging GRID technologies [5], particularly the maturing and widely adopting Web Services (WS) architecture [6], we envision a new approach to species distribution prediction: publishing both observed species presence/absence data and prediction models as Web services and then chain these Web services as scientific workflows. In this study, we aim at enabling distributed species distribution predictions with greater interoperability, flexibility and usability. While Web Services and workflow applications have been reported in other e-sciences (such as genomic

[1] This work is supported in part by DARPA grant # N00017-03-1-0090 and NSF grant ITR #0225665 SEEK.

W. Fan, Z. Wu, and J. Yang (Eds.): WAIM 2005, LNCS 3739, pp. 610 – 617, 2005.

researches [7]), we are not aware of existing modeling systems on species distribution predictions using Web Services.

This study is the first step towards building a virtual e-science laboratory for ecologists to perform distributed and cooperative research [8] on species distribution predictions. As part of a bigger project, the Science Environment for Ecological Knowledge (SEEK, [9]) – a NSF funded five year large Information and Technology Research (ITR) project, the prototype implemented for this study is built on top of several other components of SEEK project [10], particularly the EcoGrid (a collection of distributed data and analytic resources) and the Analysis and Modeling System (AMS) called Kepler (a scientific workflow system) [11].

The rest of this paper is arranged as follows. Section 2 introduces some background knowledge on species distribution prediction modeling using the Genetic Algorithm for Rule Set Production (GARP). Section 3 presents the architecture of proposed approach and discusses some implementation details. Section 4 demonstrates the feasibility of the proposed approach through a running example. Finally Section 5 is the summary and future work directions.

2 Species Distribution Prediction Modeling Using GARP

In this study, we use a specific species distribution prediction model called GARP (Genetic Algorithm for Rule Set Production, [12][13]) to demonstrate the feasibility for the proposed approach. GARP has proven especially successful in predicting species' potential distributions under a wide variety of situations [14] and have been integrated into several projects, such as Lifemapper [15] and SEEK [9]. GARP modeling requires a set of geospatial data of a study area that may be related to the species' distribution. These data sets are called environmental layers and are typically in raster format, such as elevation and precipitation. GARP also requires the observation data that recording the locations of presence or absence of species. There are three major components in GARP which are Pre-sampling, Rule generation (or Training) and Predicting. Pre-sampling produces training and testing sets by random sampling of observation data and associates the locations with the values of environmental layers at the locations. Rule generation is the process of training the genetic algorithm by the samples and generating rules that associate the values of environmental layers and the presence/absence of species for prediction. Predicting is to apply the generated rules to the values of environmental layers of all the locations of the study area and predict whether a species will be present or absent in the locations. From data mining perspective, GARP modeling can be thought as a classification or an association problem.

Traditionally, GARP systems can be run locally as a desktop application [15] or remotely through a Web browser [13]. However, all the data sets of environmental layers and all the three components of a GARP system need to be resident on a same machine. The communications between the components are often ad-hoc as well. At the same time, the volumes of environmental layer data from Geographical Information Systems (GIS) and Remote Sensing (RS) are exploding. This makes installing and running a GARP system locally inefficient since all the data need to be downloaded to the user's local machine. On the other hand, while hosting a GARP

system in a Web server avoids heavy data communication problems, it can suffer from single point failure. If any of the components in a GARP system fails working properly, it will not response to user requests any further.

In this study, we propose a scientific workflow approach to species distribution prediction modeling in distributed and heterogeneous computation environments. A scientific workflow can be seen as a scientific data analysis pipeline that connects multiple analytical steps. Compared to business workflows, scientific workflows can be data-intensive, compute-intensive, analysis-intensive, visualization-intensive, etc. There are two issues in using a scientific workflow approach to species distribution prediction modeling: The first one is that we need a workflow composition and execution environment. The second is how to represent a single analytical step in a pipeline. For the first issue, we use Kepler scientific workflow system [11][16]. For the second issue, we propose to use Web Services technology. In the proposed approach, each analytical step is implemented as a Web service and Web services are chained together to form a modeling task. In the core of Web Service technology is the Web Services Description Language (WSDL, [17]). WSDL provides a framework for defining interfaces (operations and inputs/outputs), access specification (typically Simple Object Access Protocol –SOAP [18] is used) and the endpoint (the location of the service). WSDL is written in XML and can be understood by both machine and human and can achieve greater interoperability. More specifically, we decompose the three components of GARP into Web services and multiple copies of these services can be deployed in a distributed and heterogeneous computing environment (we currently aim at supporting Windows and Linux). By using Kepler scientific workflow system, scientists can choose appropriate Web services and chain them together to compose species distribution prediction workflows and execute them either in a batch mode or interactive model as described next.

3 Architecture and Implementations

Kepler builds upon the mature, dataflow-oriented Ptolemy II system [19] which is used for modeling, simulation and design of concurrent, real-time, embedded systems. Ptolemy controls the execution of a workflow via so-called directors that represent models of concurrent computation. Individual workflow steps are implemented as reusable actors that can represent data sources, sinks, data transformers, analytical steps, or arbitrary computational steps. An actor can have multiple input and output ports, through which streams of data tokens flow. Additionally, actors may have parameters to define specific behavior. Ptolemy can perform both design-time (static) and runtime (dynamic) type checking on the workflow and data. Kepler inherits and extends these advanced features from Ptolemy and adds several new features for scientific workflows, such as prototyping workflows, distributed execution of Web and Grid services, database access and querying and supporting foreign language interfaces. We refer readers to [11][16] for more detailed information regarding to Kepler scientific workflow system.

The Web Services actor developed in Kepler serves as a proxy between the workflow system and the Web Service endpoints. There are two steps to use a Web Service actor in Kepler. The first step is to specify the URL of the WSDL of a Web service. The Web

Service actor will parse the WSDL document and retrieve available methods and their input/output types declared in the WSDL document. In the second step, users can select a method from a dropdown list. After the selection, the actor will add the corresponding ports to itself and is ready to connect its ports to ports in the other actors. We refer reader to [20] for more information of the Web Services actor in Kepler.

When a workflow is executed, the actors in the workflow will be scheduled according to the computation model of the workflow. While Ptolemy/Kepler supports a variety of computation models, two of them are frequently adopted in workflows using Web services actors, namely the Synchronous Data Flow (SDF) and the Process Network (PN). Actors need to be executed sequentially in SDF while they can be executed in parallel in PN.

The architecture of the prototype implementing the proposed approach is shown in Fig. 1. We decompose DesktopGARP [15] functions into three Web services that represent the three components in a GARP system. A database Web Services actor connecting to EcoGrid [10] which allows retrieving species occurrence data from multiple sources (museums, research institutes, etc.) is also developed. All data tokens (such as samples and rules) are encoded in XML. String data type is used for the encoded XML tokens when they pass through Kepler actors and the Web service endpoints to archive maximum system compatibility and user interpretability. The prototype was targeted at running under Apache Tomcat and Apache Axis Java. Apache Tomcat, Axis Java, Ptolemy II and Kepler are all open source based on Java. Since DesktopGARP was written in C++, we use JNI technology to wrap their native APIs into Java classes before deploying them in Apache Axis Java.

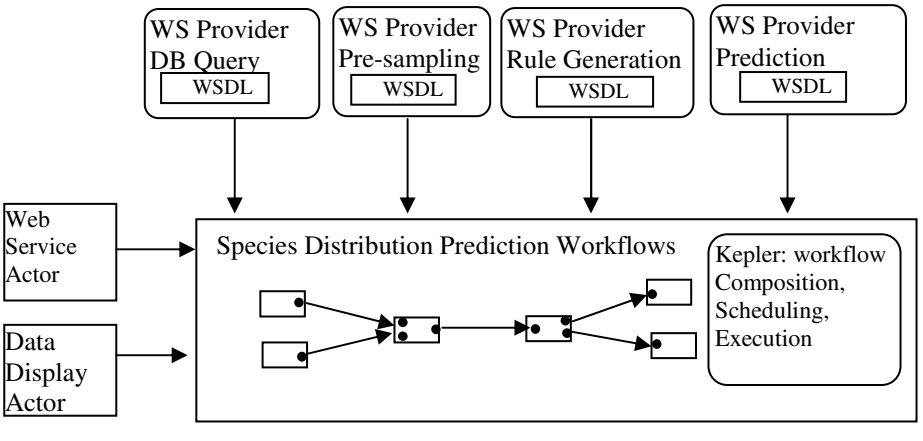

Fig. 1. Architecture

The benefits of using Web Services and scientific workflow technologies for species distribution prediction modeling can be briefly summarized as follows:

1. Efficiency. Unlike running desktop GARP systems locally, the environmental layer data management is shifted to the server side which allows using powerful database systems (such as Oracle and its Spatial option) for geospatial spatial indexing and query optimization. The communications between clients and

servers are just the pre-sampling results and the resulting rules which are generally only a small fraction of the environmental layer data sets.

2. Interoperability. Compared with GARP systems that adopt the browser/server architecture which targets at end users, the proposed Web Services approach is more component-oriented which allows integrating the Web services into a variety of end systems for different purposes. The Web Services approach can be treated as an extension of traditional Web approach with higher interoperability through a set of self-explained and machine-understandable WSDL documents.
3. Robustness. The Web Services approach allows deploying multiple copies of the GARP functional components in a distributed computation environments. If one component of a composed workflow fails working properly, it can be replaced by other similar components either manually or automatically and the workflow can still work properly.
4. More user controls. Kepler allows user to drag-and-drop workflow components and compose scientific workflows interactively. Users can add various display actors to view intermediate results and apply filtering and transformation actors to change the inputs to the next actor in a workflow. It also allows users to change the parameters of actors interactively and watch the changes of results. Finally, users can run workflows in a step-by-step mode to understand the executions of workflows better. Kepler scientific workflow system essentially provides a visual programming environment for species distribution prediction modeling without requiring any programming.

4 Demonstration

We use species Mephtitis to demonstrate the proposed approach. 60 locations of observed occurrences of the species are retrieved from EcoGrid. Three instances of the Web Service actor are materialized with Pre-sampling and Prediction Web Service instances locate on one machine and Training Web Service instance locates on another machine. Once the workflow is constructed, users can execute it in Kepler scientific workflow environment by hitting the red triangle icon located on the top of Kepler window. Kepler allows users to execute a workflow in batch mode or interactive step-by-step mode. At any time during the execution of a workflow, users can hit stop and resume the execution and watch the intermediate results. Users can view the predicted distribution map using any Internet browsers. A screen snapshot is shown in Fig. 2.

A unique feather of Kepler scientific workflow is that it allows animating the execution process of a workflow which provides a user a vivid impression of how a workflow is executed (as shown in Fig. 3 where the Web service that is being invoked is highlighted). We believe this feature is important to scientists to better understand the workflows composed by their remote colleagues. Kepler is also able to output the execution schedule when users choose to open “Listen to Director” window. The execution log and any debug information will be output to the window. Each actor in a workflow has two statuses: will be iterated and was iterated. Scientists are thus able to monitor the progress of the workflow execution process. The execution schedule and execution log for the demonstration is shown in Fig. 4.

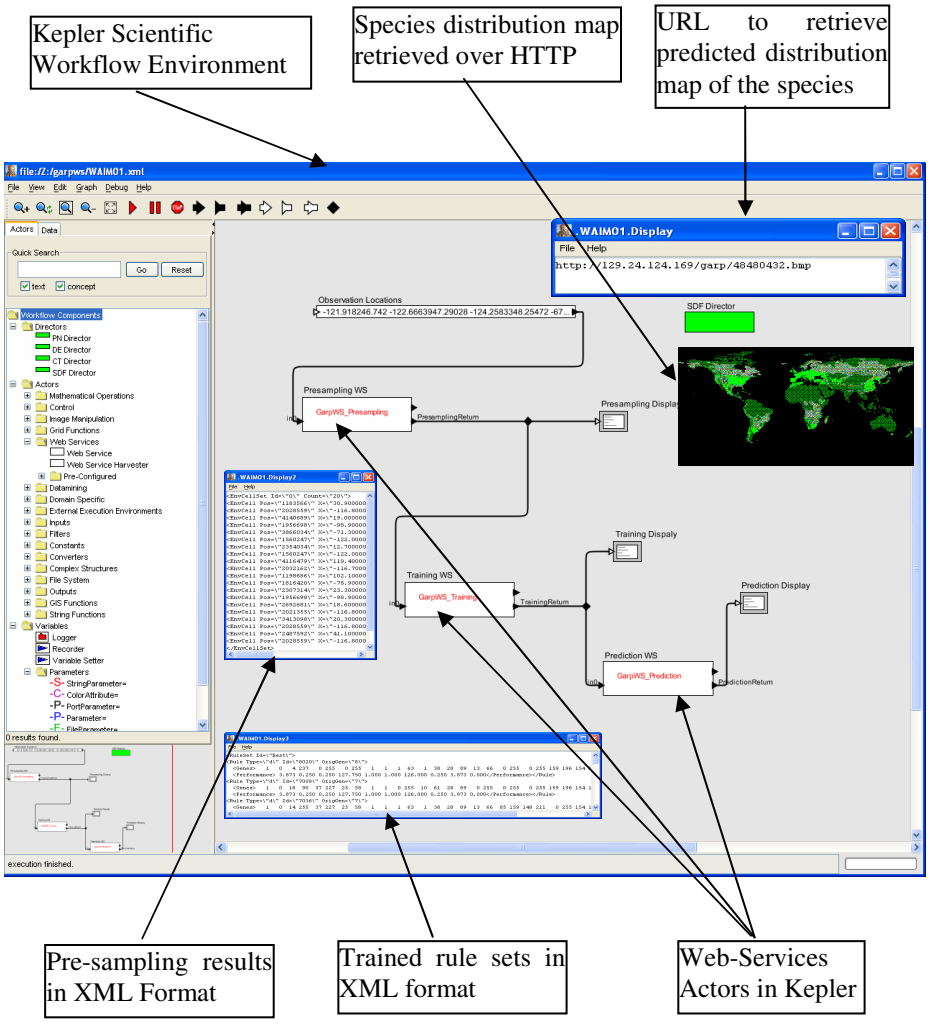

Fig. 2. A Demonstrative Species Prediction Workflow Using Web Services

5 Summary and Future Work Directions

In this study, we propose to use Web Services and scientific workflow for species distribution prediction. The proposed approach is compared with existing ones in terms of efficiency, interoperability, robustness and user controls. A running example is provided to demonstrate the feasibility of proposed approach.

For future work, we plan to implement more species distribution prediction models as Web Services in our prototype system to enable scientist users to compose more diverse scientific workflows. We also plan to incorporate another SEEK component, the Semantic Mediation System (SMS, [9]), into the prototype which allows matching both data and Web Services at semantic level to achieve greater interoperability and

enable semi-automated compositions of scientific workflows. Finally, we believe visualization tools are important for scientist users to calibrate model parameters, interpret and evaluate prediction results and we thus plan to provide such tools in our prototype in the form of Kepler actors.

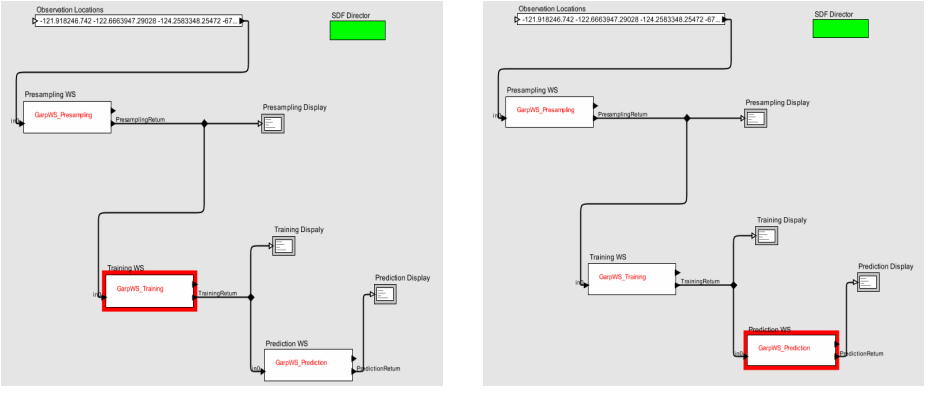

Fig. 3. Animation of Workflow Execution in Kepler for the Demonstration

```
Execute Schedule{
Fire Actor org.sdm.spa.StringConst {.WAIM01.Observation Locations}
Fire Actor org.sdm.spa.WebService {.WAIM01.Presampling WS}
Fire Actor ptolemy.actor.lib.gui.Display {.WAIM01.Presampling Display}
Fire Actor org.sdm.spa.WebService {.WAIM01.Training WS}
Fire Actor ptolemy.actor.lib.gui.Display {.WAIM01.Training Dispaly}
Fire Actor org.sdm.spa.WebService {.WAIM01.Prediction WS}
Fire Actor ptolemy.actor.lib.gui.Display {.WAIM01.Prediction Display}
}

The actor .WAIM01.Observation Locations will be iterated.
The actor .WAIM01.Observation Locations was iterated.
The actor .WAIM01.Presampling WS will be iterated.
The actor .WAIM01.Presampling WS was iterated.
The actor .WAIM01.Presampling Display will be iterated.
The actor .WAIM01.Presampling Display was iterated.
The actor .WAIM01.Training WS will be iterated.
The actor .WAIM01.Training WS was iterated.
The actor .WAIM01.Training Display will be iterated.
The actor .WAIM01.Training Display was iterated.
The actor .WAIM01.Prediction WS will be iterated.
The actor .WAIM01.Prediction WS was iterated.
```

Fig. 4. Workflow Execution Schedule and Execution Log for the Demonstration

References

[1] Secretariat of the Convention on Biological Diversity, Handbook of the Convention on Biological Diversity, Earthscan, 2001.

[2] F.A.Bisby, The quiet revolution: biodiversity informatics and the Internet, Science, 289(5488):2309-12, 2000.

[3] A. T.Peterson, , D. A.Vieglais, A.G.Navarro-Sigüenza, M. Silva, A global distributed biodiversity information network: Building the world museum. Bulletin of the British Ornithologists' Club, 123A:186-196, 2003.

[4] A.Guisan, N.E.Zimmermann, Predictive habitat distribution models in ecology. Ecological Modelling. 135:147-186, 2000

[5] Global Grid Forum, http://www.gridforum.org/

[6] Web Service (WS), http://www.w3.org/2002/ws/

[7] M.Claudia Cavalcanti, etc., Managing structural genomic workflows using Web services, Data & Knowledge Engineering, 53:45-74, 2005

[8] W.E Johnston, Semantic services for grid-based, large-scale science, IEEE Intelligent Systems, 19(1):34 – 39, 2004

[9] The Science Environment for Ecological Knowledge (SEEK), http://seek.ecoinformatics.org/

[10] S.Romanello, etc., Creating and Providing Data Management Services for the Biological and Ecological Sciences: Science Environment for Ecological Knowledge, to appear in the 17th International Scientific and Statistic Database Management (SSDBM) Conference, Santa Barbara, California, USA, June 27-29, 2005

[11] Kepler Scientific Workflow System, http://www.kepler-project.org/

[12] D.R.B.Stockwell, I.R.Noble, Induction of sets of rules from animal distribution data - a robust and informative method of data-analysis, Mathematics and Computers in Simulation 33 (5-6): 385-390, 1992

[13] D.R.B.Stockwell, D. Peters, The GARP Modeling System: problems and solutions to automated spatial prediction. International Journal of Geographical Information Science 13(2):143-158, 1999

[14] R. P.Anderson, D. Lew, A. T. Peterson, Evaluating predictive models of species' distributions: criteria for selecting optimal models. Ecological Modelling, 162:211-232, 2003

[15] Lifemapper, http://www.lifemapper.org/

[16] I.Altintas, C.Berkley, E.Jaeger, M.Jones, B.Ludäscher, S.Mock, Kepler: An Extensible System for Design and Execution of Scientific Workflows, the 16th International Scientific and Statistic Database Management (SSDBM) Conference, 423-424, 2004

[17] Web Services Description Language (WSDL), http://www.w3.org/TR/wsdl

[18] Simple Object Access Protocol (SOAP), http://www.w3.org/TR/soap/

[19] Ptolemy II, http://ptolemy.eecs.berkeley.edu/ptolemyII/

[20] Ilkay Altintas, Efrat Jaeger, Kai Lin, Bertram Ludäscher, Ashraf Memon: A Web Service Composition and Deployment Framework for Scientific Workflows, the Second IEEE International Conference on Web Services (ICWS), 814-815, 2004