A Scientific Workflow Approach to Distributed Geospatial Data Processing using Web Services

Efrat Jäger[†], Ilkay Altintas[†], Jianting Zhang[‡], Bertram Ludäscher[†,§], Deana Pennington[‡], William Michener[‡]
[†]San Diego Supercomputer Center, [‡]University of New Mexico, [§]University of California Davis
{*efrat,altintas*}*@sdsc.edu*, {*jzhang,dpennington,wmichener*}*@lternet.edu, ludaesch@ucdavis.edu*

Abstract

Geospatial analyses of distributed data from surveys and sensors are often stored and managed in diverse regional, national and global repositories. The nature of scientific processes requires composition of these resources in a meaningful order to solve a specific geoscience problem. These tasks can be viewed as scientific workflows. Web based interfaces allow access to remote data and tools, and enable running computational experiments using different online resources. However, it requires manual processing to combine multiple resources in pipelines and scientists still need IT experts to automate their large-scale scientific workflows. The challenging problem is how to enable the scientists to harvest online data and models for designing and executing experiments in a seamless manner. A solution becomes feasible by the introduction of Web Services in a variety of scientific domains. These services can be discovered and composed through generic visual interfaces and scientific workflow tools. For this purpose, we present a complete framework for registering, discovering, composing and executing Web Services to support online science.

1. Introduction

Geospatial analytical functionality is essential to environmental modeling. As large scale distributed geospatial data is available, software reuse and sharing becomes more and more important in integrated environmental modeling, experimenting and analysis. The Service Oriented Architecture (SOA) allows cooperation of data and process components among different organizational units and supports reusability and interoperability of components on the Web, thus increasing the efficiency of assembly and decreasing the cost of development.

In recent years, the need for adaptable interfaces and tools for accessing scientific data and executing complex analyses on the retrieved data has risen in a variety of disciplines (e.g., geology, biology, ecology). Such analyses can be modeled as scientific workflows in which the flow of data from one analytical step to another is described in a formal workflow language. While traditional business workflows are oriented towards document processing, task management and control-flow, scientific workflows typically are data- and/or compute-intensive, dataflow-oriented, and often involve data transformations, analysis, and simulations. Kepler [3, 14] is a system for design, discovery, execution and deployment of scientific workflows from different scientific domains. In this paper, we propose to use the Kepler scientific workflow system to compose geospatial services for environmental modeling. To the best of our knowledge, there exists no previous work on utilizing scientific workflows for geospatial analysis and environmental modeling by composing and executing Web Services in a systematic manner.

We present an approach to provide uniform access to the vast amount of highly heterogeneous services. These services and the related metadata are archived using extensive storage capabilities through registries, and the services are discovered using a metadata description of their operation. We use the Kepler workflow system to demonstrate the registration and discovery process of services within repositories. We further describe how within our framework, services can be composed into scientific workflows and executed to perform scientific tasks. Workflows can also be stored in repositories and shared between scientists. The workflow execution can be monitored to detect and recover from failures, to capture intermediate and end results of the process for data provenance, and to log process information and save execution logs. A challenging issue is to provide a web based access for viewing, executing and sharing scientific workflows and deploying scientific workflows as a new service that can be applicable to other applications.

The rest of the paper is organized as follows. In section 2, we provide a brief overview of distributed geospatial data processing and propose to use Web Services as the building blocks of distributed geospatial data processing within a scientific workflow system. Section 3 introduces the Kepler scientific workflow system's Web Services framework

and the overall architecture. Section 4 presents a running example to illustrate using Kepler to support environmental modeling. Finally, in Section 5, we provide a summary and future work directions. Although the motivating example of this paper is on geospatial data processing, the solutions are applicable to other domains as well.

2. Distributed Geospatial Data Processing

In the early days of computer-aided environmental modeling, geospatial data and process sharing between different machines was available only through manually copying it using mediums, such as floppy disks or CDs. With the development of computer networks, much of this work can be automated using tools and scripts. However, this approach is still inherently labor intensive and requires considerable human interactions which is both inefficient and error prone.

During the past few years, major Geographical Information System (GIS) software vendors (e.g., ESRI [1] and Oracle [6]) expanded their software functionality to provide distributed geospatial data management using mainstream Database Management Systems (DBMS) to support computation remotely in a computer network environment. However, there are several limitations to using a pure vendor-specific DBMS approach to distributed geospatial data processing for environmental modeling, such as cost of ownership, technology complexity and interoperability. More importantly, though database systems are naturally suited for querying/filtering using spatial indexing, they provide poor support for spatial transformations. Spatial transformations are required for transforming between data types or geospatial data values.

Nowadays, some major commercial database systems are beginning to offer some spatial transformations capabilities. However, the underlying ORDBMS model still makes it difficult to apply them to environmental data, since the data is usually unstructured or semi-structured. Therefore, a pure distributed spatial DBMS approach based on SQL-like queries for environmental modeling is undesirable if not infeasible. On the other hand, although Web GIS [12] has put major efforts towards distributed geospatial data processing, adopting the client/server architecture, they are mostly restricted to visualization capabilities, offering little support for complex queryies and analysis.

Furthermore, rendering geospatial data at client side in the form of images or Java/COM object makes the integration and reuse of geospatial data very inefficient. Although using XML as the communication protocol has been proposed for geospatial data integration purposes [15], distributed geospatial data processing that involves data transformation has hardly been investigated.

SOA provides a publishing interface to data and tools using the platform independent Web Service Definition Language (WSDL) [10]. SOA can be used for exposing geospatial data processing methods to the Web.

Within the efforts for standardization of geospatial data formats (i.e. [11, 13]), Geographical Markup Language (GML) [13] is expected to bridge between various data formats. Thus, we propose to use the GML data format and the SOA in distributed geospatial data processing.

We envision that an open architecture is vital for newly emerging integrated and distributed environmental modeling. The architecture should support (1) both flat data (such as operation system files), semi-structured and structured data (such as databases), (2) legacy models written in traditional languages or scripts, and (3) interactive and automatic executions of environmental models in the form of scientific workflows. We believe that using Web Services as the building blocks for geospatial data processing within a scientific workflow system fulfills the above requirements.

In this paper, we propose publishing geospatial data and processes as Web Services and composing them using a scientific workflow approach. In order to efficiently access distributed data and process services, we use web service registries that are accessible through the Kepler system. In the next section we present the Kepler scientific workflow system as our Web Service composition and execution framework to achieve distributed environmental modeling.

3. Kepler Web Service Framework

Kepler [3] is a system for the design and execution of scientific workflows. It is built on top of the PtolemyII system, a modeling and design tool for assembling concurrent components by means of various models of computation [7]. Kepler is an extensible open source scientific workflow system that provides scientists with a graphical user interface to register and discover resources, and to interactively design and execute scientific workflows using emerging Web and Grid-based technologies to distributed computations.

Kepler is unique in that it seamlessly combines high-level workflow design with execution and runtime interaction, access to local and remote data, and local and remote service invocation along with a built-in concurrency control and job scheduling mechanism. Other unique features are inherited from the underlying PtolemyII system, e.g., the ability to combine different models of computations in a single scientific workflow.

Computational units in Kepler are called *actors*, which are reusable components that communicate with each other via input and output ports. Actors are linked to each other to compose a *scientific workflow*. The workflow execution is orchestrated by a *director* that provides the model of computation, that is, scheduling components interaction. In this paper we explain how the Kepler environment can be utilized to discover, compose and execute geospatial data pro-

cessing workflows for environmental modeling, most essentially using a generic Web Service invocation component.

Several generic Web services actors have been implemented in Kepler that serve as clients for accessing distributed resources within Kepler workflows. Specifically, the WebService actor provides a simple plug-in mechanism to execute any WSDL-defined Web Service. An instantiation of the actor acts as a proxy for the Web Service being executed and links to the other actors through its ports. Using this component, any application that can be deployed as a remote service, can be used as a Kepler component.

Other features of the Kepler framework to support Web Service execution are shown in Figure 1 . The figure depicts the Kepler overall architecture for facilitating web service based scientific experiments. The sequence of events involved in performing and analyzing a scientific experiment are as follows. A service in our framework can be a *process service* to perform an analysis operation, or a *data service* to query over a dataset. The geospatial analytical functions that are wrapped as Web Services are process services, whereas services that query different formats of geospatial data are data services. The user or provider publishes scientific datasets and processes. The purpose of registering services is to facilitate their discovery and provide methods for their execution. In the Kepler system, services are registered using domain ontologies and can be discovered by querying over concepts in the related domain ontologies. The user can then access distributed scientific resources by searching and harvesting them. A search can be either syntactic, that is, a text based search by the services names, or semantic by issuing a query against semantic information stored in the registries. Harvesting is facilitated by a *Web Service harvester* to conveniently plug in a whole set of (possibly related) services. Discovered components can be composed to a scientific workflow and may also be registered within the system either as a local component or within domain repositories. The system provides several features for monitoring workflows execution, such as, failure recovery, data and process provenance and post execution processing. Another functionality that is currently under research and development is the deployment of a scientific workflow as a new remote service.

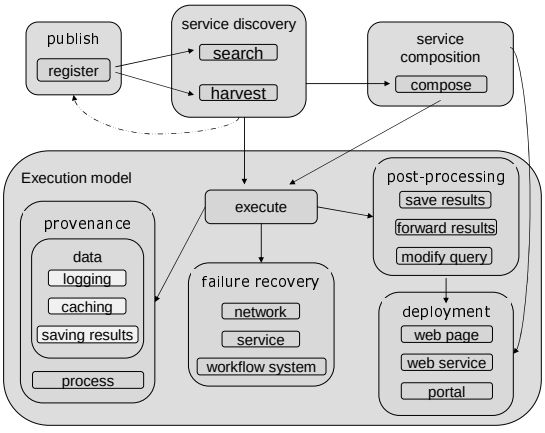

Figure 1. Life-cycle of Kepler Web Services.

4. Geospatial Data Processing Example

The following example in species occurrences analysis is used to illustrate geospatial data processing within the Kepler system. The goal is to find all the occurrences of species A that are within the intersection of the convex hulls of the occurrences of species B and species C. We assume that the occurrences datasets (point data) are stored in three different formats: species A data is stored in a flat file, species B data is stored and managed by SQL Server 2000 and species C data is stored and managed by Oracle 10g with spatial capabilities. We further assume that these three datasets are accessible through Web Services which return a GML representation of the data. Three geospatial functions are used to process the query: GRASS' Convex Hull, Oracle Spatial's Polygons Intersection and a Java Point in Polygon algorithm. These functions are wrapped as Web Services to provide a uniform, domain independent access. Finally, a visualization component is used to display GML documents. This component wraps GeoTools' [2] GML displayer as a Kepler actor. All of these components are registered within a geological repository and are discovered, composed and executed within the Kepler scientific workflow system.

Figure 2 provides a snapshot of the geospatial data processing workflow. During the workflow composition, the user first searches for the desired data and process services. Discovered services appear in top left panel of the Kepler graphical user interface and can be dragged and dropped onto the workflow canvas to perform within a scientific workflow. The services are linked to one another from their visual ports using the GUI. The semantic data transformations in this example are transparent to the user. The datasets are automatically transformed into a GML format while being accessed, using the discovered data services, therefore, no additional intermediate components are required. As for the process services, those were initially designed to consume and produce GML document strings, and thus require no further format integration processing.

As shown in Figure 2, accessing the datasets and the two Convex Hull operations can be done concurrently while the Point in Polygon and Polygon Intersection execute sequentially. Such an execution is feasible in Kepler through a *Process Network director*, (PN) [9], which schedules the workflow execution to a parallel mode (when possible) by creating a separate execution thread for each actor.

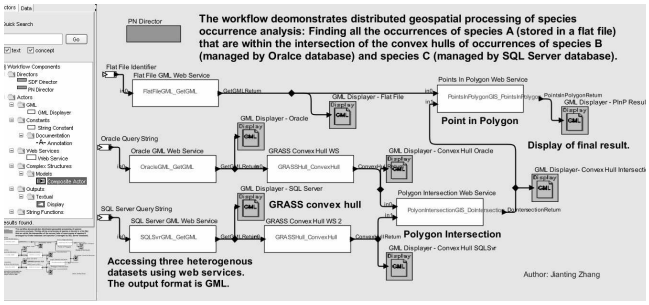

Figure 2. Modeling species occurrences.

GML display actors are used to provide a visualized display of the resulting datasets of each processing step. This actor may also be replaced with a string display actor to display the resulting GML content, or a file writer to store the computation result. Kepler also provides support (as mentioned above) for execution monitoring, such as, a time-out mechanism for connecting to a Web Service, exceptions handling, and enabling to continue halted executions from a certain checkpoint by caching components' computations. The interactive workflow system essentially provides users with a visual programming environment that can effectively harvest, compose, execute and monitor distributed geospatial processes for environmental modeling.

5. Discussion and Future Work

In this paper, we describe a framework for service-based workflows and illustrated the application of it to distributed geospatial data processing for environmental modeling. The Kepler scientific workflow system provides a convenient interface for discovering and harvesting services, and interactively composing and executing them with support for execution monitoring and data provenance.

We demonstrate the feasibility of the proposed approach for geospatial data processing with a simple running example within the Kepler system. The example emphasizes Kepler's strength to facilitate environmental modeling. By publishing geospatial data and processes as Web Services, a unified framework is achieved for accessing various formats and systems. To the best of our knowledge, there exists no previous work on experimenting with geospatial data through scientific workflows, using Web Services as building blocks. Although the focus of this example is towards environmental modeling, the proposed solutions have already been utilized within other scientific domains that uses the Kepler system's web service framework.

As for future work, from a geospatial perspective, we would like to develop a client application to facilitate publishing of datasets as GMLs, including solving efficiency issues when processing very large datasets.

We are also interested in extending Kepler's web service framework in a direction that allows efficient third-party data transfers, has specialized directors (computation engines) for optimized distributed execution for service-based scientific workflows, and provides a role-based Kepler access framework that allows users to create ad-hoc usage groups in order to enable them share their data and processing services through a unified infrastructure.

Acknowledgments

Kepler includes contributors from SEEK [4], SDM Center-SPA [8], Ptolemy II [7] and Geon [5]. Work supported by NSF ITRs 022567 (SEEK), 0225673 (GEON), DOE DE-FC02-01ER25486 (SciDAC/SDM), DARPA F33615-00-C-1703 (Ptolemy), and DARPA N00017-03-1-0090. Special thanks to Kai Lin and the rest of the Kepler developers for their support.

References

[1] ESRI, http://www.esri.com.

[2] Geotools Lite, http://sourceforge.net/projects/geotools/.

[3] Kepler: An Extensible System for Scientific Workflows, http://kepler.ecoinformatics.org.

[4] NSF/ITR: Enabling the Science Environment for Ecological Knowledge (SEEK), www.seek.ecoinformatics.org.

[5] NSF/ITR: GEON: A Research Project to Creat Cyberinfrastructure for the Geosciences, www.geongrid.org.

[6] ORACLE Spatial, http://www.oracle.com/technology/products/spatial/index.html.

[7] PtolemyII, http://ptolemy.eecs.berkeley.edu/ptolemyII/.

[8] Scientific Data Managment Center (SDM), http://sdm.lbl.gov.

[9] The Process Networks (PN) Domain, http://ptolemy.eecs.berkeley.edu/publications/papers/99/HMAD/html/pn.html.

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

[11] D. Green and T. Bossomaier. *Online GIS and Spatial Metadata*. Taylor & Francis, 2002.

[12] A. R. Hardie. The Development and Present State of Web-GIS. *Cartography*, 27(2):11–26, 1998.

[13] W. Kresse and K. Fadaie. *ISO Standards for Geographic Information*. Springer, 2004.

[14] B. Ludaescher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger-Frank, M. Jones, E. Lee, J. Tao, and Y. Zhao. Scientific Workflow Management and the Kepler System. *Concurrency and Computation: Practice & Experience, Special Issue on Scientific Workflows*, to appear 2005.

[15] J. Zhang, M. Javed, A. Shaheen, and L. Gruenwald. Prototype for wrapping and visualizing geo-referenced data in a distributed environment using XML technology. *ACM-GIS*, pages 27–32, 2000.