Lifemapper, VisTrails and EML

Documented, Re-executable Species Distribution Models

CJ Grady[1], Jim Beach[1], Jeff Cavner[1], Aimee Stewart[1]
[1] University of Kansas
cjgrady@ku.edu, beach@ku.edu, jcavner@ku.edu, astewart@ku.edu

***Abstract*— Lifemapper is an archive of species distribution models as well as web services used to access and create them. We have decided on the Ecological Metadata Language standard for providing metadata for each of our service objects and process metadata for each experiment documenting not only how the process was completed but also how it can be re-executed in the future. Combining this with the clients we have created, we have provided software that can be used to regenerate and re-execute any experiment we have created strictly from the metadata used to describe the inputs, the process, and the outputs. This is especially useful when combined with the VisTrails environment as it gives non-programmers access to powerful tools for scientific experiment generation through a user-friendly graphical interface. Additionally, providing metadata for our service items allows us to track data provenance over time. When this information is added to the documentation of an experiment, a reviewer can see exactly what was done to get from the inputs to the outputs, promoting transparency and reproducible scientific experiments**

Keywords—metadata; software; documentation; reproducability; web services

I. Introduction

The Lifemapper Project (http://www.lifemapper.org) is an NSF-funded effort to compile species distributions and computed, predictive range and diversity models. The project is comprised of two primary components. The first is an archive of species distribution models and the second is a collection of web services that access, create, and store data for species distribution modeling and biogeographical experiments. The experiments in the archive are compiled from occurrence data at both the genus and species level acquired from a local cache of species data aggregated by the Global Biodiversity Information Facility (GBIF, http://www.gbif.org) and this data is modeled and projected using scenarios of climate data from WorldClim (http://www.worldclim.org) and the Intergovernmental Panel on Climate Change (IPCC, http://www.ipcc.ch). These inputs are fed to one of the modeling algorithms such as GARP Best Subsets [1] and Bioclimatic Envelope [2] that are available to the openModeller library (http://openmodeller.sourceforge.net). Model outputs are a rule set of habitat suitability parameters and maps indicating the predicted habitat suitability for the organism in question based on the inputs to the algorithm.

The second major component of Lifemapper is the web services. All data and metadata in the Lifemapper system is available from these web services and these services are both RESTful and Open Geospatial Consortium (OGC, http://www.opengeospatial.org) compliant. User uploaded and created content is done using OGC's Web Processing Service (WPS) standard and raster data is retrieved using OGC's Web Coverage Service (WCS) standard for actual data or the Web Mapping Service (WMS) for scaled map images. Metadata about each service item is returned by tacking the desired interface parameter on the end of the REST URL. This metadata includes information of the data that is returned by an OGC service. This can include the inputs to the experiments, keywords, modification time, geospatial and temporal coverage of the data in question, cell size and resolution, or anything else related to the service items.

As an adjunct to our web services, we also provide software clients for users to efficiently access the services through applications. These clients use the published Lifemapper services API to post and request data and experiments. Our software client integration with VisTrails is especially useful (http://www.vistrails.org). VisTrails (VT) is a scientific workflow management system that allows a user to assemble and document exploratory computational tasks. Vistrails provides a graphical user interface for authoring workflows, parameterizing modules, and for pipelining data through computational steps and output visualizations. A distinguishing feature of VisTrails is its ability to generate comprehensive provenance information or metadata about complete workflows. The result of our LM/VT is a powerful tool that can be used generate complex experiments while maintaining an easy-to-use user interface.

One of our primary goals is to promote transparency and repeatability in species distribution modeling. For that, we require metadata about the inputs to an experiment. That includes where the original data can be obtained, any transformations that have been done to them, etc. Once the input data is thoroughly documented, produce metadata about the processes that transform these inputs into the final outputs. We need to track data provenance to really ensure that we capture all of the manipulations of a data set from start to finish and so that we can expose them for evaluation and validation for someone that is looking to repeat an experiment [3].

For assembling and archiving Lifemapper workflows, we use Ecological Metadata Language (EML, http://knb.ecoinformatics.org/software/eml/). EML is a metadata specification implemented as a series of XML document types [4]. Our rationale for using EML was that nearly all of the metadata we had already been providing fit into the schema and since it is XML, it could be extended to include anything else we needed to capture. Additionally, capturing process details is essential to our work and the EML specification includes process metadata in two forms, protocol and method. Allowing processes to be documented descriptively to explain what was done and prescriptively to describe how to do it [5]. This allows us to provide information to replicate the experiment by hand as well as provide instructions that can be used by our clients to automate the experiment replication, a concept we are calling "Executable EML".

II. Progress to Date

Our initial step was to generate EML for all of the data we provide in the Lifemapper archive. Each service provides metadata for each item. Climate layers and species distribution projects are provided as Spatial Rasters and point data is provided as data tables. Additionally, experiments contain the methods used to generate them as well as the protocols to use if the experiment is to be generated again.

Once our services started producing EML, we wrote libraries that could read this metadata. This is our "Executable EML" concept. For this first iteration, only very specific EML can be read and handled correctly, however, from this specifically formatted EML, an entire experiment can be regenerated. The experiment EML includes the methods actually used to generate it, including the software used, as well as the protocol for recreating the experiment. These protocol entries are links to web services that can be called to post data and submit a new experiment using the parameters contained in the document.

The most visible products we have produced related to EML are our clients, including a Python library and a package of VisTrails modules. The Python library has the capability to read Lifemapper produced EML and produced objects from it that can be used to resubmit the experiment. The Lifemapper VisTrails modules can read Lifemapper produced EML and recreate the workflow used to create the experiment. This workflow will retrieve data for the experiment that is available from a URL. When the workflow is executed, this data will be posted through the Lifemapper REST services and the experiment will be run. The EML may be loaded either in a text box that allows copy and paste or direct entry, or the VisTrails module will go out and retrieve the EML content from a provided URL.

III. Current Development

We are currently working on extending the EML we produce, and we're continuing to explore the standard to ensure that we are providing the most complete metadata possible. This expansion also includes new services that we are creating. Our goal is to provide EML for every Lifemapper service, and to add mechanisms to ensure that new EML passes validation testing.

Our present efforts also include reading and processing more generic EML. This allows us to handle external data both in our clients, and in our web services. The aim of this is to increase our interoperability with other projects and data sources. This can also allow us to expand our user upload services while at the same time providing a simpler user interface from our clients.

By allowing user uploads via EML, we can get all of the metadata for a climate layer or a collection of occurrence points. This may also be particularly useful if the user's desired data is a large file and available online. The Lifemapper system can just download the data directly, rather than the user downloading the data and then uploading it to our server.

We are currently considering options for EML cataloging and querying. We are initially looking at setting up a Metacat installation (http://knb.ecoinformatics.org/index.jsp) for EML storage, searching, and retrieval. We are also exploring the possibility of setting up a portal as part of a collaboration with the University of Oklahoma, Kansas State University, and Oklahoma State University in the EPSCoR program. This portal would store event based EML and provide a shared catalog among the institutions. EML will be a primary component allowing for the interoperability of scientific data and processes from multiple science fields. If successful, this may open the door for new scientific discovery from a hybrid field and new experiments.

IV. Future Work

In the future, we would like to introspect the EML specification XSD file to create Python objects representing each element. This approach provides multiple benefits for creating and parsing EML. New EML will be easy to validate if type checking is added to these objects. Uploaded EML documents will be validated and quickly transformed into a tree structure that will provide simple access to requested data.

We would like to expand our clients to publish EML to any server requested. This will include any EML catalog associated with Lifemapper, any DataONE node (http://www.dataone.org), or any other server that takes EML from a HTTP POST request. We will also expand the clients to generate EML for anything produced in the client. This improved interface will work similar to the way Morpho (http://knb.ecoinformatics.org/morphoportal.jsp) works and will allow users to document their newly created experiments.

Lifemapper's use of EML is helping us accomplish our goals of queryability, self-standing metadata, interoperability, and repeatable science. By using EML with our own and external portals, users can search for our data that is related to their interests. Providing EML documents with all information needed to recreate an experiment allows users to recreate and verify experiment results without using the Lifemapper system if they so choose. They are able to acquire all of the data used and know how to process it from the metadata as well as the actual procedure used to create the experiment. They can also use our clients and "Executable EML" to read all of the metadata about an experiment, get all necessary data, and then re-execute the experiment automatically. Overall, we have become a more viable option for collaboration with other projects, expanded our user-base, and are promoting transparent and repeatable science.

Acknowledgment

The Lifemapper project is funded by NSF grants EPS 0919443, DRL 0918590, BIO/EF 0851290, and OCI 0753336.

References

[1] Anderson, R. P., D. Lew, and A. T. Peterson. "Evaluating Predictive Models of Species' Distributions: Criteria for Selecting Optimal Models," Ecological Modelling, vol. 162, pp. 211-232, 2003.

[2] Nix, H. A. "A Biogeographic Analysis of Australian Elapid Snakes," Atlas of Elapid Snakes of Austrailia, pp. 4-15, 1986.

[3] Freire, J., D. Koop, and L. Moreau (Eds.). "The Open Provenance Model: An Overview," IPAW, LNCS 5272, pp. 323-326, 2008.

[4] Ecological Metadata Language (EML) http://knb.ecoinformatics.org/software/eml/

[5] Ellison, A. M., L. J. Osterweil, L. Clarke, J. L. Hadley, A. Wise, E. Boose, D. R. Foster, A. Hanson, D. Jensen, P. Kuzeja, E. Riseman, and H. Schultz. "Analytic Webs Support the Synthesis of Ecological Data Sets," Ecology, 87 (6), pp. 1345-1358, 2006.