

## A Prototype Publishing Registry for the Virtual Observatory

Ramon Williamson, Raymond Plante

*National Center for Supercomputing Applications, University of Illinois  
Urbana-Champaign, Urbana, IL 61801*

**Abstract.** In the Virtual Observatory (VO), a registry helps users locate resources, such as data and services, in a distributed environment. A general framework for VO registries is now under development within the International Virtual Observatory Alliance (IVOA) Registry Working Group. We present a prototype of one component of this framework: the publishing registry. The publishing registry allows data providers to expose metadata descriptions of their resources to the VO environment. Searchable registries can harvest the metadata from many publishing registries and make them searchable by users. We have developed a prototype publishing registry that data providers can install at their sites to publish their resources. The descriptions are exposed using the Open Archive Initiative (OAI) Protocol for Metadata Harvesting. Automating the input of metadata into registries is critical when a provider wishes to describe many resources. We illustrate various strategies for such automation, both currently in use and planned for the future. We also describe how future versions of the registry can adapt automatically to evolving metadata schemas for describing resources.

### 1. Introduction

In the Virtual Observatory (VO), a registry helps users locate resources, such as data and services, in a distributed environment. A general framework for VO registries is now under development within the International Virtual Observatory Alliance (IVOA) working group. We present a prototype of one component of this framework: the publishing registry.

A publishing registry provides a mechanism for data providers to publish descriptions of their data and resources that they want to be made available. This publishing registry can then be harvested by fully searchable registries that take the descriptions and make them available to the general user for interrogation and searching (see articles in this publication concerning searchable registries by Greene et al. 2004 and McGlynn et al. 2004). Local searchable registries can also be created that may contain specially harvested datasets for specialized searches. For more information about the Registry Framework, see Plante et al. 2004.

The publishing registry is composed of two parts: an entry form and a harvesting interface. The entry form is used to enter the data and publish it into the registry, and the harvesting interface which exposes the data for discovery.

## **2. VORegistry-in-a-Box: Astronomical Data Registration Made Easy**

Setting up, creating, and maintaining a publishing registry can be a tedious job, especially for those new to the concept. We have tried to simplify the process by making available our VORegistry-in-a-Box. VORegistry-in-a-Box contains all of the scripts required to create a publishing registry, including an entry form and an OAI-Compliant harvesting interface. All that is required to start your own publishing registry is Perl and a Web Server.

### **2.1. The Entry Form**

The entry form now in use at the NCSA NVO Registration Portal is a Perl-CGI form. Throughout the design process, an emphasis was placed on simplifying the ingestion of multiple resources, as well as easy accommodation of evolving metadata schemas. The CGI form features:

- Password protection  
Multiple users can enter data into the registry without harm to each other's data
- Resource List for easy perusal of the resources entered so far
- Separate publishing step so that previously published registries can more easily be synchronized.
- When adding new resources, the entry form automatically inherits values from chosen resources, minimizing the amount of typing when entering multiple similar resources
- Minimal cost to buy-in to create your own registry (A Web Server machine, Apache, and Perl)

The resource descriptions are stored in XML files on disk using the emerging IVOA standard schema for describing resources called VOResource (IVOA Registry Working Group 2003); this is the primary export format delivered through the harvesting interface.

### **2.2. The Harvesting Interface**

The harvesting interface provided with the VORegistry-in-a-Box implements the Protocol for Metadata Harvesting, a standard for disseminating resource metadata developed by the Open Archives Initiative (OAI; Legoze et al. 2002). This standard was chosen because it is an existing, well-tested standard, there exists a number of supporting software tools, and its wide use in the digital library world makes our metadata available to the broader library community.

The OAI harvesting interface enables agents to collect metadata from multiple registries in a uniform way. The most common reason for collecting metadata would be to centralize it and make it searchable by users. Thus, the OAI interface intentionally does not support complex queries, only the simplest filters based on topic and date of last update (that is, a complex query interface is what defines the "searchable registry"; see Plante et al. 2004). The OAI standard can

support any community-specific, XML-based format for metadata; however, it mandates that an implementation must at least support the OAI-Dublin Core format to allow cross-community interoperability.

The harvesting interface included in the VORegistry-in-a-Box package is the OAI-XMLFile package created by Hussein Suleman of Virginia Tech (Suleman 2002). We modified the package slightly to support the protocol's feature for marking deleted records. We use the interface primarily to export the metadata in the IVOA-specific format, VOResource; however, the required OAI-Dublin Core format is also supported automatically via an XSL stylesheet.

### 3. Future Work

Currently under development is a Java package to automatically create the entry GUI on-the-fly from an XML Schema which defines the data structure. The XML schema is read using a Java SAX parser. Widget components are created based on the data type and numbers of allowed values as parsed from the data schema. This allows the publishing tool to adapt to new and changing data models. The widgets verify that the values being entered into them are valid for their datatype warn of illegal values.

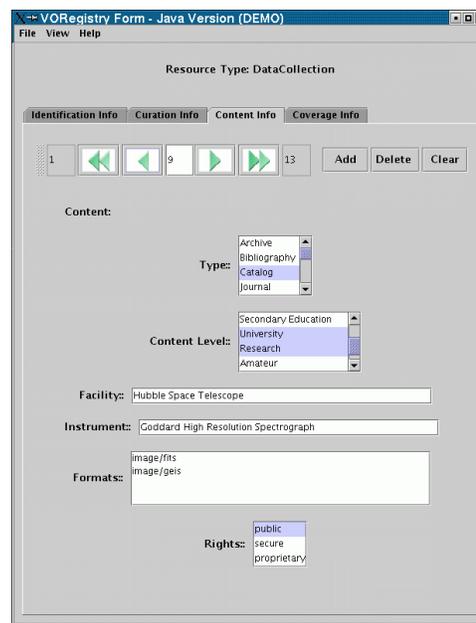


Figure 1. Java GUI

Figure 1 shows a prototype of the Java interface. With this GUI, a user provides values for a complex element called “Content” that contains six simpler child elements. This complex element, hypothetically, can appear multiple times in the schema; the VCR-style buttons allows the user to flip through the thirteen sets of metadata and make changes to each of them independently, as well as add or delete sets of values.

Java was chosen for this prototype due to:

- The ability to handle complex, customized inputs
- The ability to generate rich GUI's on the fly from an XML Schema
- Automated handling of help and user tips from XML documentation.

The scope of the Schema to GUI translation is limited to VO schemas, and includes handling of primitive types such as Strings, Integers, Floats, Dates and Booleans, and more complex types composed of combinations of the primitive types.

#### 4. Accomplishments

We have created a useful tool for creating and maintaining publishing registries. It is targeted to data providers wishing to expose a moderate number of resources to the VO environment. The user requires no knowledge of the OAI Standard or the internal formats of the data, yet they get an OAI 2.0 Compliant publishing registry that can immediately be used. The package is easily installed and set up with little outlay of time or resources. The Java version of VORegistry-in-a-Box, now in development, will generate entry forms automatically from the XML Schema with no additional programming necessary.

#### 5. Availability

- To try out the interface, you can access our registry directly at:  
<http://nvo.ncsa.uiuc.edu/nvoregistration.html>

You can either login as user "sample" to view a sample set of resources, or you can try adding your own resources, either as a test or to be published.

- To create your own publishing registry using VORegistry-in-a-Box, download the gzipped tar of the package at:  
[http://nvo.ncsa.uiuc.edu/VO/software/VORegistry\\_in\\_a\\_Box.tar.gz](http://nvo.ncsa.uiuc.edu/VO/software/VORegistry_in_a_Box.tar.gz)

#### References

- Greene, G., O'Mullane, W., Hanisch, R., & Gaffney, N. 2004, this volume, 285  
IVOA Registry Working Group, 2002, IVOA Resource Registry,  
<http://www.ivoa.net/twiki/bin/view/IVOA/IvoaResReg>
- Legoze, C., Von de Sompel, H., Nelson, M., Warner, S. 2002, The Open Archives Initiative Protocol for Metadata Harvesting,  
<http://www.openarchives.org/OAI/openarchivesprotocol.html>
- McGlynn T., Lee, J., Hanisch, R., O'Mullane, W., & Greene, G. 2004, this volume, 319
- Plante, R., Green, G., Hanisch, B., McGlynn, T., O'Mullane, W., Williams, R., Williamson, R. 2004, this volume, 585
- Suleman, H. 2002, OAI-PMH2 XMLFile File-based Data Provider  
<http://www.dlib.vt.edu/projects/OAI/software/xmlfile/xmlfile.html>