Using iRODS to Preserve and Publish a Dataverse Archive

Mason Chua*, Antoine de Torcy**, Jewel H. Ward***, Jonathan Crabtree*
*H.W. Odum Institute for Research in Social Science
** Data Intensive Cyber Environments Center
*** School of Information and Library Science
The University of North Carolina at Chapel Hill

Abstract

We developed a method for transferring the contents of an archive running Dataverse, a publishing program for scientific data, into iRODS [5]. This method respects the encapsulation of the Dataverse archive by exporting its contents through documented methods using the OAI-PMH and HTTP protocols. Since the metadata exported from Dataverse conforms to documented standards (including the OAI-PMH and at least one other metadata specification), we were able to use an XSL transformation to reformat it into a document whose contents can be deserialized into the iRODS metadata catalog. As a result, iRODS users can use iRODS metadata to do keyword searches on the serialized copy of the Dataverse archive. Furthermore, this method lets administrators use iRODS to apply data preservation policies, including storage resource redundancy, to the contents of a Dataverse archive.

Index Keyword Terms**—**OAI-PMH, DDI, data archive, digital library, descriptive metadata, preservation, web publishing, migration, interoperability, XML, XSL transformation, serialization, HTTP, search.

1. Introduction

The archivists at the H. W. Odum Institute for Research in Social Science use an open-source web publishing platform called Dataverse to publish their extensive collection of files related to social science research studies. As part of the Odum Institute's effort to test interoperability between archive platforms and data grid technologies, we developed a method to automatically copy the contents of a Dataverse archive into iRODS. The result is an accurate copy of a Dataverse archive inside iRODS, which data grid administrators can preserve over the long term by, for example, replicating the information to many geographically distributed storage resources. The transfer process also automatically populates the iRODS metadata catalog with descriptions of the data. This descriptive metadata lets iRODS users search the archive much as Dataverse users can do keyword searches using the web.

This paper assumes no knowledge of Dataverse, whose relevant features are explained in the following section. We do, however, assume a basic understanding of iRODS, as explained in chapter 2 of the iRODS Primer [3]. Section 2.2 explains the relevant differences between Dataverse and iRODS, and Section 3 describes how we overcome these differences in order to automatically transfer a Dataverse archive into iRODS. Section 4 explains the significance of this work.

title World urbanization, 1950-1970
handle hdl:1902.29/D-488
distributor Odum Institute Dataverse Network
citation Davis, Kingsley, 1970, ‘‘World Urbanization, 1950-1970’’, http://hdl.handle.net/1902.29/D-488, Odum Institute [Distributor]
holdings URI http://arc/study?globalId=hdl:1902.29/D-488

Figure 1: On the Dataverse website, metadata is displayed as human-readable attribute-value pairs, hiding the underlying hierarchical structure (as seen in figure 2).

2. Background

2.1. The source: Dataverse

Dataverse is a publishing program for data archives. From the user's point of view, it is a web library of files associated with metadata.

```
<record>
<metadata>
  <docDscr>
    <citation>
      <titlStmt>
          <titl>World urbanization, 1950-1970</titl>
          <IDNo agency="handle">hdl:1902.29/D-488</IDNo>
     </titlStmt>
     <distStmt>
          <distrbtr>Odum Institute Dataverse Network</distrbtr>
          <distDate date="2007-11-30">2007-11-30</distDate>
     </distStmt>
     <biblCit format="DVN">
          Davis, Kingsley, 1970, "World urbanization, 1950-1970",
     <distStmt>
          <distrbtr>Odum Institute Dataverse Network</distrbtr>
          <distDate date="2007-11-30">2007-11-30</distDate>
```

```
        </distStmt>
        <biblCit format="DVN">
              Davis, Kingsley, 1970, "World urbanization, 1950-1970",
                http://hdl.handle.net/1902.29/D-488,
                Odum Institute [Distributor]
         </biblCit>
          <holdingsURI="http://arc/study?globalId=hdl:1902.29/D-488"/>
       </citation>
    </docDscr>
  </metadata>
</record>
```

Figure 2: An excerpt of the serialized version of a typical metadata object in Dataverse.

For example, a file that contains the numerical results of a survey would be paired with metadata that contains the survey's location, description, location, sample size, date range, keywords, citation requirements, and even the contents of the survey questions. Researchers can find files on Dataverse by requesting keyword searches of the metadata.

Under the hood, Dataverse stores its information in a filesystem directory and a relational database, which are both hidden from the users. It is technically possible to back up an entire Dataverse archive by just copying these underlying files and database tables. These **raw backups** are risky, however, because their format might not be compatible with future versions of Dataverse. Although a raw backup contains all of the information in the digital library, recreating an archive from the backup might require inventing an ad hoc conversion process between formats. To prevent this problem, the Dataverse developers have provided two standard interfaces for exporting information from the library: the metadata can be downloaded using the OAI-PMH protocol, and (as mentioned above) the files can be downloaded through the HTTP protocol. OAI-PMH is a protocol for metadata harvesting that transfers XML over HTTP [2]. Although the metadata on a Dataverse website looks like a flat association list, such as the one below, its internal structure can conform to one of many existing metadata standards, including USMARC, Dublin Core, and DDI [1, 4].

The methods described in this paper can be applied to any of the metadata formats that Dataverse can export, but the particular examples are designed for DDI metadata. This metadata is a tree structure serialized as an XML document, like all metadata transferred through OAI-PMH.

2.2. The destination: iRODS

In iRODS, each metadata element is a list of three strings, called the **attribute**, **value**, and **unit**, together called an **AVU**. iRODS users can associate any AVU with any file in the iRODS data grid (assuming they have permission to do so).

Like Dataverse, iRODS can perform keyword searches by generating lists of objects whose metadata match arbitrary string expressions. But unlike Dataverse, iRODS does not store any hierarchical structure on the set of metadata elements associated with a file. Compare the serialized iRODS metadata in figure 3 to the serialized Dataverse metadata in figure 2: in Dataverse, a file's metadata is a hierarchical tree of metadata elements that can have arbitrary text fields associated with them, An iRODS file's metadata, in contrast, is an unstructured set of AVUs. The main challenge of this project is finding a way to use iRODS to both preserve the hierarchical structure of the metadata in a Dataverse archive while exposing its contents to iRODS features, such as keyword searching.

3. Methods

We split the goal of this project into two parts and addressed each one separately.

3.1. Part one: Preservation

iRODS's ability to preserve data is a result of its abstraction of storage: each data object can be reduntantly stored in many geographically separated machines. Therefore, for the sake of preservation, it makes sense to serialize the contents of a Dataverse archive and then store them as data objects in iRODS, exposing them to the replication and integrity-checking features of iRODS.

Our automatic transfer script, called **Dataverse-to-iRODS**, takes advantage of Dataverse's ability to export serialized versions of its files and metadata through the HTTP and OAI-PMH protocols (respectively). The transfer of data objects depends on the transfer of metadata, because each metadata object contains URL references to the files it describes. After Dataverse-to-iRODS has run, iRODS has a collection containing each XML metadata object and the files it refers to. Here are the steps of the transfer process:

1. Dataverse-to-iRODS uses OAI-PMH to request a list of identifiers, which are the unique names of the metadata objects in the Dataverse.
2. The list of identifiers is an XML document. Dataverse-to-iRODS uses SAX to extract each identifier and use it to request a metadata object through OAI-PMH.
3. Each metadata object is an XML document. Dataverse-to-iRODS uploads the XML document into an iRODS collection corresponding to its identifier.

4. For each metadata object, Dataverse-to-iRODS also uses SAX again to extract the URL of each data object that the metadata refers to. It then downloads each object by HTTP and uploads it into the same collection as the metadata that refers to it.

As seen in steps 2 and 4, it is the metadata itself that lets Dataverse-to-iRODS discover the URLs of data objects and identifiers of other metadata objects. This discovery process is only possible because the metadata conforms to well-documented standards. The initial list of metadata identifiers conforms to the OAI-PMH protocol, which allowed us to write the XML parser that extracts each identifier. Similarly, each metadata object itself conforms to the DDI standard, which specifies the contents and hierarchy of the metadata precisely enough that we can parse out the URL of every file that the metadata refers to.

3.2. Part two: Exposure

We wanted the contents of Dataverse archives to be as accessible to iRODS users as possible. Dataverse websites let users browse for studies categorically and find them by keyword searches. In this project, we used iRODS's metadata catalog to re-implement these keyword search capabilities over a Dataverse archive after it has been serialized and transferred into iRODS. For each XML document containing Dataverse metadata, we automatically use its contents to compute the producer, notes, topic classification, explicitly-listed keywords, and about ten other fields that pertain to the metadata and its related files. We then ingest these fields into the iRODS metadata catalog and associate them with both the XML file containing the original metadata and the data objects that the metadata refers to. As a result, iRODS users can use the iquest program to perform the same keyword searches that are available through Dataverse. For example, the following command, when typed on one line, will return a list of iRODS collections containing all objects whose distributor contains the string "Odum Institute".

```
iquest "SELECT DATA NAME, COLL NAME
where META DATA ATTR NAME like
      '%Study Distributor'
and META DATA ATTR VALUE like
      '%Odum Institute%' "
```

Since the iquest program allows arbitrary metadata queries, it is not as user-friendly as Dataverse for basic keyword searches. It would be easy, however, to recreate Datavere's ease of searching by writing a web interface that translates the contents of an intuitive web form into an iquest query.

This metadata ingest process occurs automatically during each Dataverse-to-iRODS transfer, with the following steps:

1. After uploading each XML document containing metadata, Dataverse-to-iRODS calls an iRODS rule.
2. The iRODS rule calls the msiXSLTransformationApply microservice, which applies an XSL Transformation to the XML-encoded metadata to extracts the parts of the metadata that we think users will want to search for by keyword.
3. The rule writes the resulting transformed XML to a temporary file. This transformed XML is a list of attribute, value and unit triples.

```
<metadata>
    <AVU>
        <Attribute>title</Attribute>
        <Value>World urbanization, 1950-1970</Value>
        <Unit></Unit>
    </AVU>
    <AVU>
        <Attribute>handle</Attribute>
        <Value>hdl:1902.29/D-488</Value>
        <Unit></Unit>
    </AVU>
    <AVU>
        <Attribute>distributor</Attribute>
        <Value>Odum Institute Dataverse Network</Value>
        <Unit></Unit>
    <AVU>
</metadata>
```

Figure 3: A serialized set of iRODS metadata objects attached to a single file. The "attribute, value, unit" triples are not arranged in any hierarchical structure, except for being attached to the same file.

4. The rule calls the msiLoadMetadataFromXml microservice, which ingests these AVUs into the iRODS metadata catalog.
5. The rule associates the AVUs with each object that they apply to – the XML file containing the metadata that the AVUs were derived from, as well as the data files that the metadata refers to.

Although each AVU can be attached to many files, the iRODS metadata system only stores one copy of it in its internal database.

As in section 3.1, the success of this process relies on the fact that Dataverse and iRODS conform to documented standards. Because Dataverse's exported metadata conforms to both OAI-PMH as well as the DDI specification, it was possible to write the XSL Transformation that extracts the parts of the metadata that we wanted iRODS users to be able to search for. The metadata ingest also exploits iRODS's ability to both perform XSL transformations and deserialize an XML list of AVUs into the metadata catalog.

4. Conclusions

This paper has described how our script, Dataverse-to-iRODS, automatically creates a copy of a Dataverse archive inside iRODS, exposing it to iRODS's long-term preservation mechanisms and metadata-based search features. Dataverse-to-iRODS uses existing standards, namely OAI-PMH, XML, and any OAI-PMH-compatible metadata specification, to copy the data into iRODS data objects for preservation and ingest selected metadata fields into the metadata catalog to allow for keyword searching.

5. References

[1] Gary King, Merce Crosas, Ellen Kraffmiller, Leonid Andreev, Gustavo Durand, Robert Treacy, Kevin Condon, Michael Heppler, and Akio Sone. The Dataverse Network Project/Features. Retrieved Februrary 26, 2010, from http://thedata.org/software/features.

[2] Carl Lagoze and Herbert Van de Sompel. The open archives initiative: building a low-barrier interoperability framework. In *ACM/IEEE Joint Conference on Digital Libraries*, pages 54–62, 2001.

[3] Arcot Rajasekar, Michael Wan, Reagan Moore, Wayne Schroeder, Sheau-Yen Chen, Lucas Gilbert, Chien-Yi Hou, Christopher A. Lee, Richard Marciano, Paul Tooby, Antoine de Torcy, and Bing Zhu. *iRODS Primer*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers, 2010.

[4] Mary Vardigan, Pascal Heus, and Wendey Thomas. Data Documentation Initiative: Toward a Standard for the Social Sciences. *The International Journal of Digital Curation*, 3(1), 2008.

[5] Jewel H. Ward, Antoine de Torcy, Mason Chua, and Jonathan Crabtree. Extracting and Ingesting DDI Metadata and Digital Objects from a Data Archive into the iRODS extension of the NARA TPAP using the OAI-PMH. In the *5th IEEE International Conference on e-Science*, Oxford, UK, December 2009.