Chapter 13 - Descriptive Metadata and Islandora

Many discipline-specific and community-specific metadata standards have evolved to meet the challenge of supporting data management and discovery as well as capturing and communicating information to users. Islandora's system is based on an understanding that the complexity of content-types and metadata standards, and the scarcity of metadata creation, management, and validation tools pose barriers to effective information management.

 The following chapter will outline the role of descriptive metadata in the Islandora system. If you are using Islandora, it is presumed you have a strong understanding of metadata standards that you will be working with, and the role of descriptive metadata in organizing information assets.

Islandora also preserves the use of Fedora's underlying administrative metadata (such as audit streams that indicate what has happened to an object) and relationship metadata (codified in the RELS-EXT streams of objects). Some Islandora Solution Packs also utilize the EXIFtool to extract technical metadata from objects and store it in a separate datastream.

Extensible Markup Language (XML) and Descriptive Metadata

As the proliferation of Descriptive Metadata standards continues to create interoperability and compatibility issues, XML is increasingly viewed as a key tool in automating and translating metadata, so that appropriate Metadata can continue to be produced, persisted, and managed. The benefits are widely recognized as reduced effort, greater consistency of metadata, and enhanced accuracy of metadata.

Using XML to represent metadata means that a representative document is created that represents what fields are used to describe a given object, and then these fields are mapped to a particular object, creating an XML-based metadata record. These representative documents are called schemas, and have an .xsd file extension. The use of XML for metadata also means that XML transformations (which have an extension of .xslt) can be used to translate or crosswalk metadata between schemas. Islandora leverages this XML approach to metadata to facilitate the creation of ingest forms, and the automatic crosswalking of Descriptive metadata.

Descriptive Metadata Datastreams in Islandora

Islandora utilizes Fedora's ability to represent Descriptive metadata in XML format via one or more **Datastreams** in an object. Fedora is written in such a way that any object may have multiple metadata Datastreams, which can store Metadata following any schema, such as MODS, Dublin Core, or QDC.

Metadata is added to an object in Islandora via an ingest form presented to users adding objects to the repositor, this ingest form produces an XML metadata datastream when the object is ingested. Each metadata stream will have a default **DSID**. Users fill out a form with fields corresponding to the metadata desired, and the relevant Datastream is created when the object in Fedora is created. The Datastreams that Islandora creates are defined in the Content Model Object that is affiliated with the Collection into which an item is being ingested. The Content Model prescribes, among other things, the **DSID** of a datastream containing metadata that can be viewed and edited via Islandora. Islandora's Solution Packs come with a Content Model which is ingested into your repository when the Solution Pack is installed, and which prescribes the type of metadata that will be collected on ingest, and in what Datastream that metadata will be stored. Users enter metadata via a Form that is defined in a Content Model. Solution Packs come with a set of forms pre-installed. These forms are based on our understanding of best practices, and represent a starting point for those new to the process of creating XML forms.

The system is designed to simplify the process of creating metadata for an object. One Islandora content model may have a number of forms to suit the needs of different collections. Forms can be edited, created, copied, and affiliated with Content Model objects using the Islandora Form Builder (XML Forms Modules). When you use the Form Builder, an .xsd in the Form Builder Modules can read in an externally or internally stored schema (another .xsd document), and allow for a form to be created and validated based on that external schema. Users can then associate the newly created or edited form with other content models in the repository via the User Interface (providing that they associate the form with a content model that prescribes that scheme, and permits an XML datastream corresponding to it). In order to fully utilize the Forms Builder, users will have to have an understanding of XML, Schema documents, and Xpath (the language used to navigate XML documents).

Crosswalking Metadata Datastreams in Islandora

In order to crosswalk metadata, Islandora makes extensive use of XML transformations, or .xslts. Transformation serve a number of purposes in the Islandora system. Of particular significance here is the way that Islandora crosswalks data on ingest.

Fedora requires that any object created in the system contain a default Dublin Core Stream. By extension, any object that is created in Islandora (and therefore in a Fedora repository) will have a default Dublin Core Datastream. However, Islandora's Solution Packs presume that users will often want to store an additional metadata stream outside of the default Dublin Core stream, in order to create richer descriptive metadata, and also to adhere to standards for metadata description of particular types of data and collections. For example, the MODS form that comes by default with any Solution Pack is designed to suit the most common cases for that solution pack.

This means that Islandora's Solution Pack Content Models come with a Datastream defined that allows for additional granularity and is customized to the needs of a particular data type or subject area. This also means that the Solution Packs come with a form that presents users with a richer metadata schema than Dublin Core. When metadata is created using the ingest form, an .xslt is called by the Content model. This .xslt file transforms the richer metadata schema into the default Dublin Core datastream, and stores both the richer XML based Descriptive Metadata and the Dublin Core Metadatain the object. In this way, Islandora preserves a common metadata stream that can be useful for searching and retrieving metadata objects across the repository, as well as a more granular metadata stream that describes the object as is most appropriate for the subject area or discipline to which the object refers. When a metadata field is updated through the interface, the .xslt is called to perform the transformation again, making sure that the Dublin Core datastream is kept consistent with the data in the richer Metadata Datastream.

In the end, the architecture surrounding descriptive metadata in Islandora is designed to provide out-of-the box metadata creation, but also customization. New forms can be created and affiliated with Content Models via the interface. Content Models can be written to define any number of metadata datastreams, and to call .xslt files to create new datastreams on ingest, and to update datastreams when metadata is edited. The system leverages the external community by taking advantage of .xslts that are commonly produced to serve similar use-cases for other organizations.