Specifications for Metadata Processing Tools
Roy Tennant, California Digital Library • roy.tennant@ucop.edu

BACKGROUND
The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH, http://www.openarchives.org/) specifies a method for digital repositories (also called “data providers”) to expose metadata about their objects for harvesting by aggregators (also called “service providers”). Metadata is exposed via “sets,” or collections of metadata that data providers decide to make available for harvesting. Service providers harvest sets from data providers of interest, and provide search services for the resulting collections of metadata (for a good example of a service provider, see http://www.oaister.org/). Data providers also decide which metadata formats to expose for harvesting, beyond the one required data format of simple Dublin Core (see http://dublincore.org/).

The OAI-PMH is relatively new, and both data and service providers are still learning the best methods for exposing metadata for harvesting and gathering that metadata into centralized search services. Many of the issues that have surfaced during exploratory harvesting by CDL are outlined in the document “Bitter Harvest: Problems & Suggested Solutions for OAI-PMH Data & Service Providers” available at http://www.cdlib.org/inside/projects/harvesting/bitter_harvest.html. That paper proposes a model for service providers that includes a series of post-harvest tasks to be performed on the harvested metadata, depending on the particular requirements of the proposed search service (see Figure 1).

Figure 1. A proposed model for production metadata harvesting
The purpose of this document is to outline the specifications for a set of post-harvest metadata processing functions that will be required to create an effective search service for harvested metadata. Specific CDL projects that will rely on harvested metadata include the American West Project (see <http://www.cdlib.org/inside/projects/amwest/> for more information) and the NSDL Project (see <http://www.cdlib.org/inside/projects/metasearch/nsdl/> for more information).

In addition, our goal is to specify general functions that can be applied within other contexts, whether or not the metadata was harvested from a remote source. Specifically, these functions should be able to be used by any CDL project that has metadata processing requirements using the principles and procedures outlined by the CDL Common Framework.

**GENERAL SPECIFICATIONS**

The specifications are delineated by function, but multiple functions may be performed by one software application or one function may be performed by multiple software applications (e.g., one application may normalize dates while another normalizes place names).

**Processing Profiles**

Each of the functions specified below, with the sole exception of the analysis tool, will likely profit from using a profile for specifying how a given metadata cohort (metadata cohort being defined as any group of metadata sufficiently homogenous to be addressed with one profile) should be processed. This will enable the periodic re-harvesting of a repository and the automatic application of specific transformations. For metadata harvesting to be viable, it must be automated as much as possible. The profile should be machine-parseable but human-readable, with XML being a likely encoding solution but not the only alternative. Also, it may be fruitful to use the concept of inheritance. For example, a broadly-applicable profile could be defined that could apply to many metadata cohorts, each which would inherit those transformations but also use cohort-specific transformations identified in a cohort-specific profile.

**ANALYSIS FUNCTION**

Before metadata can be transformed it must first be understood. Therefore, an essential first step is metadata analysis. Metadata analysis should be able to answer a number of important questions, for example:

- Which metadata fields are present?
- What percentage of the total number of records have each field?
- How consistent is the metadata within those fields?
- What patterns can be detected?

**Specification:**

The metadata analysis function should be able to:

- List all elements present in a cohort and the percentage of non-empty elements of each type
- Output a list of all non-empty occurrences of a particular element (e.g., each and every date field)
- Total the number of duplicate instances of the same element (e.g., how many records have a date of 2002?)
Specifications for Post-Harvest Metadata Tools

- List all non-duplicative content for a specific element and the number of times the content occurs (e.g., 1999 = 25, 2001 = 545, etc.)
- Identify patterns across records (e.g., all records with “x” in the “y” field do not have a “z” field)

NORMALIZATION FUNCTION

Normalization is the process of standardizing the way in which information is recorded. For example, test harvesting has turned up a wide variety of methods for encoding dates. For example:

- 1991-10-01
- ca. 1920.
- (ca. 1920)
- 2001.06.08 by CAD
- Unknown
- ca. June 19, 1901.
- (ca. June 19, 1901)
- [2001 or 2002.]
- 1853.

A normalization process would make these dates conform to a specific encoding method; e.g.:

```xml
<date format="ISO 8601">1991-10-01</date>
<date format="ISO 8601" type="circa">1920</date>
```

Specification:
The metadata normalization function should be able to:
- Process the contents of specified elements to make them conform to particular specifications (e.g., the W3C date format, see the “NSDL ‘Safe’ Transforms” document at <http://metamanagement.comm.nsdlib.org/safeXform.html>).
- Strip out empty elements or those with no information value (e.g., “unknown”)
- Strip out HTML markup
- For other possible normalization routines, see the “NSDL ‘Safe’ Transforms” document.
- Perform all functions based on a profile for that harvested set

ENRICHMENT FUNCTION

Harvesting metadata removes the metadata from a particular environment and places it in an entirely new one. This loss of context alone can require us to reinsert context by adding metadata to each record (e.g., source) if it is not already present. Depending on the situation, we may find it necessary to explicitly define other information that could be implied within the context of the remote repository.

Also, if we decide that all of our metadata should be enclosed within a METS wrapper, we will need to create such a wrapper for records that do not come to us in METS.

Specification:
The metadata enrichment function should be able to:
- Insert new elements into each harvested record
- Add qualifiers to metadata elements in each harvested record (this will also likely be necessary as part of the normalization routines above)
Specifications for Post-Harvest Metadata Tools

- Mapped specified values to different values (e.g., map “UMMZ Fish Division” to “University of Michigan Museum of Zoology Fish Division” or “literature; Old Testament” to “Bible. O.T. English”)
- Optionally create new elements based on information in the record (e.g., extract place names from other elements such as <title> and place them in a geographic coverage element)
- Optionally query other information sources (e.g., a gazetteer) to enrich existing or machine-created elements (e.g., <coverage>)
- Create a METS record from the base metadata record
- Perform all functions based on a profile for that harvested set

SUBSETTING FUNCTION
OAI-PMH-compliant repositories make metadata records available for harvesting as “sets” or groups of records. How these sets are created and made available is entirely up to the data provider, which creates a wide variety of possibilities and little or no option for a service provider to pick and choose which metadata is of interest. Therefore, it is necessary for service providers to have methods for identifying and segregating the desired records after harvesting.

CDL may also need this subsetting function when creating focused search targets of records within our content management system (CMS) for searching via subject-specific metasearch portals.

Tool Specification:
The metadata analysis function should be able to:
- Search by any metadata field
- Select records individually or as a group (e.g., the entire search result set)
- Move selected records into a separate area apart from the originally harvested set (either virtually or actually)
- Subset based on a profile (e.g., once the processing parameters are established, be able to re-execute based on specific instructions)