Building Interoperable Digital Libraries: A Practical Guide to Creating Open Archives

Hussein Suleman (hussein@vt.edu), Virginia Tech

1 Outline

1.1 Introduction to the Open Archives Initiative

The Open Archives Initiative (OAI) is dedicated to solving problems of digital library interoperability by defining simple protocols, most recently for the exchange of metadata.

The work of the OAI was initiated in connection with a meeting of representatives of various electronic pre-print and related archives (e.g., NDLTD, arXiv, NCSTRL) in Santa Fe, USA in October 1999. From this meeting emanated an agreement among the archivists to support a common set of principles and a technical framework to achieve interoperability. This started the process of defining standards, broadened to include digital libraries other than pre-print archives. Community comments were collected at meetings in San Antonio and Lisbon during the following year.

A technical working group provided input into the process of creating and testing standards under widely differing conditions. This process culminated in the announcement of a new protocol for interoperability, the Open Archives Initiative Protocol for Metadata Harvesting, in January 2001. These standards are now being disseminated to all interested parties who wish to adopt a low-cost approach to interoperability, with support from the existing members of the Open Archives community.

This tutorial is aimed at introducing individuals to the concepts underlying the OAI and the harvesting protocol, as well as providing sufficient information to allow attendees to almost immediately implement the standard on their own archives. In addition, attendees will be introduced to issues that need to be addressed when building new systems, either in the capacity of being providers of data, users of data or both.

1.2 Definitions and Concepts

The OAI has currently adopted the harvesting approach to interoperability, one that may not be well known or understood by digital library practitioners. As such, it is crucial to explain, discuss, and disambiguate the concepts and terminology used among OAI implementers. Among these are the following issues:

- Basic Principles
    - What is a Repository?
    - What is an Open Archive?
    - Why choose **harvesting** over **federation**?
    - Metadata interoperability vs. Data interoperability
    - **Data providers** and **Service providers**

- Underlying technology
  - HTTP requests and XML responses
  - XML, XML namespaces and XML Schema
- Protocol policies
  - Unique identifiers and guarantees of persistence
  - What are **sets** and how and when do we use them?
  - What is a **record** in the context of OAI, and how does it differ from the popular definition?
  - How is **metadata** encapsulated into records?
  - Support for multiple metadata formats
  - Datestamps, and how they are used for harvesting and self-containment
  - Flow control mechanisms

1.3 Requirements to be a Data Provider

In order to be a data provider, an existing archive needs to satisfy a list of technical requirements. New archives can be built with these requirements in mind to ease the path to becoming OAI data providers at a later stage. Included among these are:

- Maintaining add/modify/delete datestamps for records
- Keeping track of deleted records
- Having unique identifiers for records
- Mapping the internal data format to Dublin Core, and optionally other existing standard metadata schemata, or creating new metadata schemata

Coupled with the list of archive policy decisions, any data provider also needs to have the technology to implement the OAI protocol. Pointers and information will be provided for those tools that have been specifically useful to past implementers, as well as those tools that have been created by the community to support new data providers.

1.4 Description of the Harvesting Protocol

The harvesting protocol defined by the OAI is a request/response protocol with 6 request types. The syntax and semantics of the protocol requests will be discussed, specifically:

- *Identify* – self-description of an Open Archive
- *ListMetadataFormats* – metadata formats supported by an archive
- *ListSets* – addressable sub-divisions contained within an archive
- *GetRecord* – metadata for a single record
- *ListRecord* – metadata for all records within specified constraints

- *ListIdentifiers* - identifiers for all records within specified constraints

1.5 Answers to Common Questions

Some of the most common problems and questions will be discussed along with the solutions that have been implemented by existing archives. This will include discussion of the following issues:

- What if an archive does not have datestamps for records?
- How can we synthesize unique identifiers?
- What if we want more metadata than is provided by just Dublin Core?

1.6 Requirements to be a Service Provider

A basic introduction will be presented on the issues pertinent to being a service provider, a user of the data provided by data providers. This will cover some of the practicalities of implementing harvesters and incorporating them into existing digital library systems, with brief discussion on the issues of acceptable usage, ownership, metadata mapping, and the building of multi-tiered hierarchical digital libraries.

1.7 Implementation Details

Pseudo-code (and where applicable and appropriate, fragments of actual code) will be used to give practical insight into the programming techniques that may be used to implement the harvesting protocol.

1.8 Testing with the Repository Explorer

The Repository Explorer (http://purl.org/net/oai_explorer) is a tool that may be used to browse through an archive using only the OAI harvesting protocol. This tool is primarily used for testing new implementations and displaying the metadata in the format seen by service providers. The use of this (or a similar tool) is vital in ensuring compliance because of the wide range of possible protocol and encoding errors that are detected when using the explorer. A number of examples of such will be displayed as illustrative of both errors in implementation and as exemplar testing techniques.

1.9 Building Communities within the Framework of the OAI

Beyond the basic use of the harvesting protocol, many communities desire extensions to support specific functionality. Some examples of features and how they could be implemented within the overall protocol framework will be discussed. For example:

- Retrieving full-texts, thumbnails, etc.
- Layering the protocol when complexity is not needed