Chapter 1

An Introduction to ORE

Abstract

This chapter of Object Reuse and Exchange (OAI-ORE) sets the stage for the report, providing introductions to the basic specification and basic concepts and explaining what readers will learn in subsequent chapters.

The Open Archives Initiative Object Reuse and Exchange (OAI-ORE, or simply ORE) specification defines a set of new standards for the description and exchange of aggregations of Web resources that presents an exciting opportunity for us to revisit how digital libraries are provisioned. ORE and its concept of aggregation—that a set of digital objects of different types and from different locations on the Web can be described and exposed together as a new, compound entity—may present the next major disruptive technology for librarians who develop and manage collections of digital information.

Currently, the management and presentation of digital library collections revolve mostly around the digital library systems that house them. A librarian decides what digital resources go together and then works within the capabilities of the system to present the resources in an appropriate and orderly context. The result is typically a series of webpages that human beings need to navigate to find and click on the links to the resources that meet their information needs. While the system may expose its metadata for harvesting or its index for federated searching, the digital resources themselves are tucked deeply inside proprietary silos.

ORE presents the possibility of breaking down these silos by exposing the semantics of these resources and providing hooks to retrieve them without the need for a human being to read a webpage and click on a link. Liberating digital library content from these silos for reuse and exchange may very well explode the construct of the "collection" as we know it today because it will no longer be the exclusive domain of librarians to bring together digital library resources and dictate the context of their presentation for use. Human beings and machines will be able to assemble their own "collections."

If you've looked for information on the Web or attended a presentation about the ORE standard, it is likely that within the first five minutes, you were confronted with a large, complicated diagram with circles and lines and references to a half dozen other, different technologies. If you weren't familiar with these other underlying technologies and tried to learn about them, you were probably confronted with even more diagrams and circles and lines. It can be overwhelming.

In the beginning of the ORE specification, it is suggested that the reader become familiar with:

- The architecture of the World Wide Web
- Semantic Web concepts such as RDF and RDFS
- Cool URIs and Linked Data[1]

If you hadn't already been working with these technologies or don't come from a technical background, that's a tall order! ORE can be difficult to approach because it is typically explained in terms of the various technologies that make up its foundation. The foundation is important, but someone new to the topic may quickly lose sight of the forest through the trees.

The goal of this issue of Library Technology Reports is to present a tutorial for librarians on ORE to make it more accessible and understandable. Our approach is to begin by presenting the general concepts of ORE and then work backwards to explain and fill in some of the

supporting technical details. Our goal is not to present a comprehensive account of ORE but instead to make it approachable by people who are not programmers or computer scientists. If you're interested in developing solutions with ORE for your library, this report will be a good starting point before you dig deeper into the references in chapter 5.

http://www.openarchives.org/ore is the official website for ORE. It maintains the specification and related documentation such as user guides, a primer, news releases, and other community resources.

We'll begin in this first chapter by explaining the rationale for ORE and describing the basic components of the ORE abstract data model: Aggregations, Aggregated Resources, Resource Maps, and Proxies. Chapter 2 starts with an introduction to RDF (explained later in this chapter) before walking through a practical example—the National Digital Newspaper Program at the Library of Congress—and a series of simple graphs to illustrate a Resource Map and Aggregation, metadata, Aggregated Resources, and nested Aggregations. In chapter 3, we will explore Resource Map serialization by looking at examples from the same project in RDF/XML. A selection of current ORE implementations and tools that are relevant to libraries will be presented in chapter 4, including profiles of projects at the Los Alamos National Labs, Ghent University, Microsoft, Johns Hopkins University, and the Texas Digital Library. Chapter 5 provides a list of references and further reading.

While the Semantic Web and ORE represent potentially disruptive technologies, the need for librarians to help make sense of interoperable digital information by provisioning resources with care and quality metadata and by connecting users to resources—and resources to resources—is greater than ever. In order to capitalize on these technologies, we must first understand them and be able to relate them to our professional practice of librarianship.

Basic Concepts

Speaking in generic terms, an aggregation is simply a group or collection of things. For example, you may aggregate food to prepare a meal. You can begin with recipes that include lists of ingredients and descriptions of how to prepare the dishes you've chosen to make. Some of the ingredients may come from different places. You probably have some of them locally in your fridge or cabinet, but you may need to fetch some of them from various remote locations. For example, you may pick up a loaf of bread at the bakery or a bottle of Merlot from your local wine shop. You may even be interested in a particular instance of wine, perhaps from a specific year, that has been recommended to you by a friend.

Everything needed for your meal has been represented all together above as an aggregation, but you can also view the dishes and their recipes and ingredients as their own aggregations. Aggregations can include other aggregations. A salad may be an aggregation that includes different kinds of lettuce, tomatoes, and salad dressing. If you look at the label on a bottle of dressing, you'll see a list of the ingredients in it: another aggregation! And so on. Once you've retrieved all of the items from your shopping list, the end result is that you have everything you need assembled in your kitchen to prepare the dishes and serve the meal.

Aggregations Are Collections

This concept of aggregation is not new to librarians, who have been aggregating content into library collections for centuries. Some librarians are bibliographers who create lists of information sources on various topics of interest. Some of the sources listed in a bibliography may be in the library's local holdings, but some of them may be located elsewhere—perhaps at another library or online on a website. In the traditional analog practice of librarianship, collection management included purchasing books and subscribing to print journals, cataloging them, and arranging them on shelves for patrons to find and use. With the shift from print to digital technology, many of the same principles of collection management are now employed in the aggregation of electronic resources such as databases and e-journals. Libraries are also involved in collecting born-digital content on platforms such as institutional repositories, and many librarians are digitizing special collections and presenting and managing digital libraries.

In all of these activities, librarians define the information that constitutes a collection. This is typically guided by collection development policies that are informed by the mission of the library and the information needs of its patrons. The boundary of such an aggregation is usually established by a librarian as well, in a library's catalog, for example. In other words, what separates a book that belongs to a collection from one that does not?

In most cases, the immediate user of a library collection is assumed to be a person, and librarians have designed their interfaces for people to use. This makes perfect sense in the analog world, where print collections are classified and physically arranged in a building with signs to direct patrons in navigating and using a collection. But in the case of digital libraries, some of a library's collections may also be in electronic formats that

History of the Open Archives Initiative

In 1999, Paul Ginsparg, Rick Luce, and Herbert Van de Sompel issued a Call For Participation to bring together developers and managers of e-print repositories to explore possible collaborations. The resulting Santa Fe Convention begat the Open Archives Initiative (OAI), whose goal was stated as being: "to transform scholarly communication by providing a technical and organizational framework to facilitate interoperability among repositories."[2]

Under the leadership of Carl Lagoze from Cornell University and Herbert Van de Sompel from Los Alamos National Labs, the OAI collaboratively developed the OAI-PMH and ORE specifications and grew to include a diverse community of scientists, software developers, repository managers, publishers, and librarians who shared a common interest in facilitating scholarly communication.

The current mission statement of the OAI says that it "develops and promotes interoperability standards that aim to facilitate the efficient dissemination of content. The Open Archives Initiative has its roots in an effort to enhance access to e-print archives as a means of increasing the availability of scholarly communication. Continued support of this work remains a cornerstone of the Open Archives program. The fundamental technological framework and standards that are developing to support this work are, however, independent of the both the type of content offered and the economic mechanisms surrounding that content, and promise to have much broader relevance in opening up access to a range of digital materials. As a result, the Open Archives Initiative is currently an organization and an effort explicitly in transition, and is committed to exploring and enabling this new and broader range of applications. As we gain greater knowledge of the scope of applicability of the underlying technology and standards being developed, and begin to understand the structure and culture of the various adopter communities, we expect that we will have to make continued evolutionary changes to both the mission and organization of the Open Archives Initiative."[3]

are available over a network. The users of digital libraries can be human beings or computer programs.

The problem is that most digital libraries have been provisioned for people, not computer programs, to use. When you boil it down, the primary point of access for a digital library is usually a webpage that presents information along with links to other webpages. The language used to create webpages, Hypertext Markup Language (HTML), was developed for exactly that simple purpose: to mark up and present electronic information to human beings. It is obvious to a person who is accessing a webpage in a digital library that they are looking, for example, at a page in a book that has been digitized. Visual cues such as page numbers at the bottom of the screen or breadcrumb navigation through a table of contents conveys meaning: the semantics of the object being viewed. It doesn't matter if the page numbers are at the bottom left or bottom right of the page, people understand the construct of a "book" and can immediately recognize what it is that they are viewing and how to navigate and use it. People also understand the relationships implied in a book such as those between pages, chapters, works cited from other books, the title page, and the index at the back of the book.

Computer programs cannot understand the semantics of such Web resources unless the resources are exposed and expressed in a way that can be identified and understood by a machine. This is the goal of Object Reuse and Exchange: to provide a standard for identifying aggregations of Web resources and describing the constituents or the boundary of an aggregation.

Librarians invest a great deal of time and care in preparing and assigning metadata to describe digital objects and in presenting and managing objects in digital libraries. There is no question that this process adds considerable value to the digital objects. Unfortunately, most of this value is lost when the user is a computer program and not a human being because a computer program cannot identify and understand how to use the object or any of the valuable information added to it by the librarian. A computer program doesn't know it is accessing a digitized book when the book is being represented in HTML; likewise, it has no way of figuring out the title of the book, its author, and other information about the book that may be obvious to a person who can recognize and read the book's metadata.

The URL (or more generally, the Uniform Resource Identifier or URI) to a Web resource in a digital library typically does not link directly to the digital object itself but to a representation of the object. This representation is usually a "splash page" that is presented for a person, not a computer program, to read and comprehend. To illustrate the point, look at the digital object being represented in figure 1, which is a conference poster that has been deposited into a library's institutional repository.

It is obvious to you, a person, that clicking the Download button will download a copy of the conference poster. But suppose someone wanted to write a program to download all of the conference posters from the collection. The program could retrieve a list of the URIs of every object in the Library Research Publications collection (this piece of metadata could be harvested easily using the OAI-PMH), but the program would probably

What Is the OAI-PMH?

The Open Archives Initiative Protocol for Metadata Harvesting, more fondly known as the OAI-PMH, defines a protocol for exposing and harvesting metadata records. OAI-PMH data providers expose their metadata to be harvested; service providers (also known as harvesters) query data providers and selectively harvest metadata records from them.[4] Most data providers are archives of scholarly resources, such as institutional repositories, publishers, and digital libraries. A common application of the OAI-PMH is to harvest and index large quantities of metadata for the purpose of providing a portal to search across collections that are distributed in multiple remote data providers.

The OAI-PMH protocol is defined as a standard Web service. The harvester sends a request to the data provider using HTTP, in much the same way a Web browser would request a webpage from a web server. The data provider then responds with its answer encoded in Extensible Markup Language (XML). At a minimum, unqualified Dublin Core metadata records are exchanged, although other additional formats can be provided.[5] Unqualified Dublin Core provides a "common ground" for the purpose of basic metadata interoperability, although its generic nature sometimes limits its use in specialized applications.

Requests can include one of six different OAI-PMH verbs:

- Identify
- ListSets
- ListMetadataFormats
- ListIdentifiers
- ListRecords
- GetRecord

The response to the Identify verb is simply the name of the data provider. In most implementations of OAI-PMH, a set corresponds to a collection; from these, the ListSets verb returns a list and descriptions of collections hosted by the repository. ListMetadataFormats returns the metadata formats available for the object that has been requested. The oai_dc XML schema is most common. Each record has a unique identifier, and a list of these is returned by the ListIdentifiers verb. This is often (but not always) the URL of a representation (e.g., "splash page") of the digital object. The ListRecords verb returns more information about the records than simply their identifiers and supports parameters to limit the results. The most recent version of OAI-PMH, version 2, supports the use of resumption tokens to provide better flow control and avoid over-saturating data providers by requesting too much metadata at one time. Finally, the GetRecord verb returns an entire metadata record for an object from the data provider.[6]

An excellent tutorial describing the OAI-PMH is hosted by the OAI Forum,[7] and a detailed transactional approach to learning the protocol from the perspective of coding a harvester in the Perl scripting language can be found in Building OAI-PMH Harvesters With Net::OAI::Harvester.[8] The largest OAI-PMH service provider, OAIster, currently contains over 23,000,000 harvested metadata records from over 1,100 data providers.[9] These records can be searched and accessed through OCLC's free WorldCat service.[10]

How is the OAI-PMH different from ORE? Generally speaking, the focus of the OAI-PMH is on exchanging the metadata that describe digital objects, whereas the focus for ORE is on exchanging and using the digital objects themselves. ORE allows you to harvest objects and not just their metadata. Beyond harvesting, ORE enables a many-to-many web of relationships among objects to be discovered, linked, and utilized. Objects described and exposed using ORE are useful outside of an ORE context to the larger Semantic Web, for example, as Linked Data.[11] By comparison, the OAI-PMH is limited to acting in a client-server manner that requires both the service and data providers to "speak" the same specialized protocol: the OAI-PMH.

To be fair, the OAI-PMH and ORE were created for different purposes using different paradigms, so they cannot (and should not) be compared as apples to apples. In at least one case, the Texas Digital Library uses the OAI-PMH and ORE together in a complementary fashion.

malfunction when it tried to download the first object because it would receive this splash page instead of a PDF or other file that constituted the actual poster. The program has no way of knowing that there is an additional step (to click the Download button) to download the file. It also can't make much sense of the rest of the information being presented, such as the title of the poster, its authors, the document type (which identifies it as being a poster), the abstract, and the link to a supplementary report that provides context for the poster.

You could make an argument that, with some knowledge of this specific institutional repository and collection, you could write a program that is aware of the link behind the Download button and could accomplish this task. You might even be able to reassemble some of the structured metadata by indexing the page or applying some other heuristics, like those that Google uses for ranking relevant search results. But would this program work with a different digital library that presents different representations of its objects? Chances are it wouldn't work with any precision because the splash pages that it would encounter would be constructed

differently. For example, the Download button might be located somewhere else on the page, or instead of a Download button, the title of the object might be a link that the user is expected to click to download the object. There are some other important questions that could be asked. Could such a program be able to differentiate between conference posters and other types of objects in the collection? What if you wanted your program to download and assemble all of the posters or their supplementary files from a particular conference and those files were archived across multiple institutional repositories? What if you wanted to move a set of objects from a digital library to a preservation repository or another digital library platform without losing their semantics?

Figure 1
A typical HTML "splash page" that represents a digital object in a digital library

Object Reuse and Exchange Data Model

ORE was developed to address these kinds of issues for objects on the Web. It introduces the concept of an Aggregation of Web resources. We'll capitalize Aggregation when we're speaking about ORE Aggregations to differentiate them from the generic use of the word aggregation. For example, we could imagine an Aggregation that contains the PDF of a conference poster, its descriptive metadata, and a Microsoft Word document that is the supplementary report. These things that are being aggregated are called, simply enough, Aggregated Resources. An Aggregation has a URI that is used to identify it, just like any other resource on the Web.[12]

The ORE e-mail discussion list is maintained as a Google Group. You can search and browse the mailing list archives and subscribe at http://groups.google.com/group/oai-ore.

Unlike a Web resource, an Aggregation is a conceptual construct. Even though it has a URI, it is not tangible; you can't download it. An Aggregation is expressed by a Resource Map, or ReM for short. A Resource Map provides details about an Aggregation in a machine-readable format.[13] In our first example of aggregating food to prepare a meal, you could think of the shopping list as being like a Resource Map. The Resource Map is something tangible: you can download it, and it will reference the Aggregation and list its Aggregated Resources. A Resource Map can also express relationships and properties pertaining to its Aggregated Resources as well as metadata about Resource Map itself, such as who created it.[14]

Institutional repositories are beginning to include support for ORE. The Texas Digital Library has developed an ORE implementation for DSpace. The oreProvider project has produced an ORE add-on for Fedora, and Microsoft supports ORE natively as a part of its Zentity repository platform. See chapter 5 for a list of notable ORE implementations and tools.

A Resource Map also has its own URI that resolves to one or more serializations. You can think of serialization as being a way to write something down.

> It lets you take an object or group of objects, put them on a disk or send them through a wire or wireless transport mechanism, then later, perhaps on another computer, reverse the process: resurrect the original object(s). The basic mechanisms are to flatten object(s) into a one-dimensional stream of bits, and to turn that stream of bits back into the original object(s).
>
> Like the Transporter on Star Trek, it's all about

> taking something complicated and turning it into a flat sequence of 1s and 0s, then taking that sequence of 1s and 0s (possibly at another place, possibly at another time) and reconstructing the original complicated "something."[15]

The three formats for serialization explained in the ORE specification are RDF/XML, RDFa, and Atom XML, although there are others.[16] We'll learn more about the serialization of Resource Maps in chapter 3.

Lastly, an Aggregation can include a Proxy, which is an Aggregated Resource in the context of a specific Aggregation.[17] For example, it is not uncommon for journal articles to be republished as book chapters. For many situations, the context from which the object being included in your Aggregation may not matter (i.e., from the article in the journal or from the chapter of the book). For some applications, such as providing citations for Web resources, it may be critical. When the context does matter, you have the option of designating an Aggregated Resource as being a Proxy so that you can make assertions about it in the context of a specific Aggregation.

To summarize, the ORE Data Model is made up of four entities: Aggregations, Resource Maps, Aggregated Resources, and Proxies. An Aggregation contains a

Examples of Aggregations and Applications of ORE

Examples of Aggregations

- A simple unordered set, or bag, of Resources, such as a collection of favorite images from various web sites.
- A multi-page, HTML document where the pages are linked together by hyperlinks that provide "previous page" and "next page" access.
- Information available from "social networking" sites, which contain content and related social activity around that content. An example is Flickr, where each participant has an entry page providing access to images in multiple sizes and resolutions that are organized in sets and collections. All of these entities are separate Resources. These are then linked to additional Resources that are comments and annotations about the images.
- A scholarly publication stored in an ePrint repository such as arXiv or in a DSpace, ePrints, or Fedora repository. Such a publication may appear on the Web as multiple Resources, each with an individual URI. The set of Resources typically consists of a human readable "splash page", that links to the body of the publication in multiple formats such as LaTeX, PDF, and HTML. In addition, the publication may have citation links to other publications, each existing as one or more Resources.
- An overlay journal issue that aggregates multiple scholarly publications as described above, each located in their origin repository, into an issue. Issues may be recursively aggregated themselves into volumes, and then into the journal itself.
- A semantically-linked group of cellular images—each available as a Resource resident in repositories from research laboratories, museums, libraries, and the like—in the manner implemented in the ImageWeb Project.
- Published scientific results such as those envisioned by Clifford Lynch that, in addition to the features of the scholarly publication described above, incorporate data plus the tools to visualize and analyze that data.

Examples of Applications

- Crawler-based search engines could use such descriptions to index information and provide search results sets at the granularity of the aggregations rather than in addition to their individual parts.
- Browsers could leverage them to provide users with navigation aids for the aggregated resources, in the same manner that machine-readable site maps provide navigation clues for crawlers.
- Other automated agents such as preservation systems could use these descriptions as guides to understand a "whole document" and determine the best preservation strategy for the document Compound Object.
- Systems that mine and analyze networked information for citation analysis/bibliometrics could achieve better accuracy with the knowledge of aggregation structure contained in these descriptions.
- Institutional repository applications could use them as the basis of interoperability for exchange and service interaction with other institutional repositories.
- These machine-readable descriptions could provide the foundation for advanced scholarly communication systems that allow the flexible reuse and refactoring of rich scholarly artifacts and their components Value Chains.

—Excerpt from the *ORE User Guide: Primer*, http://www.openarchives.org/ore/1.0/primer.

Resource Map plus one or more Aggregated Resources, which can also be Proxies. In the next chapter we will introduce the Resource Description Framework (RDF), which forms the foundation of the Semantic Web and gives us a language to use for talking about Aggregations in greater detail. We'll also visually explore Aggregations, Aggregated Resources, and Resource Maps by using graphs to illustrate how they relate to one another.

Notes

1. Carl Lagoze et al., "ORE User Guide: Primer," Oct. 17, 2008, Open Archives Initiative Object Reuse and Exchange website, http://www.openarchives.org/ore/1.0/primer (accessed March 6, 2010).
2. Herbert Van de Sompel and Carl Lagoze, "The Santa Fe Convention of the Open Archives Initiative" *D-Lib Magazine* 6, no. 2 (Feb. 2000), http://www.dlib.org/dlib/february00/vandesompel-oai/02vandesompel-oai.html (accessed March 6, 2010).
3. Open Archives Initiative, "Mission Statement," About OAI page, http://www.openarchives.org/OAI/OAI-organization.php (accessed March 6, 2010).
4. Open Archives Initiative Protocol for Metadata Harvesting website, http://www.openarchives.org/pmh (accessed March 6, 2010).
5. Ibid.
6. Ibid.
7. Susanne Dobratz, Friederike Schimmelpfennig, and Peter Schirmbacher, "OAI for Beginners: The Open Archives Forum Online Tutorial," 2002, http://www.oaforum.org/tutorial (accessed March 6, 2010).
8. Ed. Summers, "OAI-Harvester-1.0," 2004, CPAN Search website, http://search.cpan.org/~esummers/OAI-Harvester-1.0 (accessed March 10, 2010).
9. The OAIster Database," OCLC website, http://oaister.org (accessed March 10, 2010
10. WorldCat website, http://www.worldcat.org (accessed March 10, 2010).
11. Linked Data website, http://linkeddata.org (accessed March 10, 2010).
12. Lagoze et al., "ORE User Guide: Primer."
13. Ibid.
14. Ibid.
15. Marshall Cline, "What's This 'Serialization' Thing All About?" Serialization and Unserialization: C++ FAQ Lite, 1991–2009, http://www.parashift.com/c++-faq-lite/serialization.html#faq-36.1 (accessed March 10, 2010).
16. Lagoze et al., "ORE User Guide: Primer."
17. Ibid.