Dynamic Generation of OAI Servers

J. Alfredo Sánchez, Antonio Razo, Juan Manuel Córdova, Abraham Villegas
CIRIA/CENTIA/ICT
Universidad de las Américas Puebla
Cholula, Puebla, México
+52-222-229-2257
{alfredo.sanchez, antonio.razo}@udlap.mx, {juancordovab, abraham.villegas}@gmail.com

ABSTRACT

We describe *Voai* and *Xoai*, two software environments that facilitate the automatic construction of OAI servers for collections managed via relational and XML databases, respectively. We have used Voai and Xoai to generate OAI servers for diverse collections. We use freely available tools and do not impose programming requirements upon the users. By making this software publicly available, we aim to facilitate the process of joining the OAI community and becoming data providers.

Categories and Subject Descriptors

H.3.7 Digital Libraries; H.2.5 Heterogeneous Databases

General Terms

Standardization, Management, Documentation, Design.

Keywords

Metadata servers, OAI-PMH, XML databases, Dublin Core.

1. INTRODUCTION

The OAI Protocol for Metadata Harvesting (OAI-PMH) has become popular as a mechanism for publicizing and sharing digital collections. In this approach, servers that implement OAI-PMH are built for each collection and added to a central registry as data providers. Applications and value-added service providers thus can query OAI data providers in a uniform fashion regardless of the varying internal structure of the underlying digital collections.

Since collections differ in organization and structure, OAI-PMH servers need to be constructed specifically for each of them. Oftentimes, staff at institutions that are eager to share their digital repositories is knowledgeable in database design and construction but lack expertise (or time) needed to build an OAI server. This becomes an obstacle towards the OAI goal of generating an open, extensible federation of highly diverse collections. Major changes in OAI-PMH also have resulted in the need to develop new metadata servers. At the time of this writing, over 90 data providers that complied with version 1.1 of OAI-PMH appear as "purged" from the community of OAI-PMH 2.0 data providers. Clearly, tools are needed to support the process of constructing OAI servers and joining (or staying in) the OAI interoperability movement in a dynamic fashion. *Voai* and *Xoai* are software environments we have developed to facilitate the construction of OAI-compliant servers by just using basic knowledge about the target collections.

2. RELATED WORK

A number of projects have been undertaken that focus on providing tools to completely or partially automate the process of building OAI-compliant servers. A survey of such projects is presented in [1]. All of these projects consider collections stored on relational databases. Some of them support collections stored on multiple tables, whereas others rely on an intermediate table with a structure known to the OAI server. Some involve only a minimum amount of programming whereas others require complex configuration by the user or even manually changing the code they generate. We are not aware of servers that can be generated automatically for collections that are stored on XML DBMSs. Our *Voai* server generator is able to generate code that will work for diverse collections stored in multiple tables on relational databases. It does not impose any programming requirements on the user. Considering that XML databases are increasingly being used for storing digital collections, we also developed *Xoai*, which generates OAI servers for this class of semi-structured repositories (relying on the XML:DB interface).

3. VOAI—THE RELATIONAL CASE

In a conventional approach, OAI servers would be constructed for every distinct collection. With Voai, specifications for each collection are received and used to automatically generate the code of OAI servers. Upon execution of each server code, the metadata of the corresponding collections become available via OAI-PMH requests for applications and service providers.

Conceptual design. For each of the verbs comprised by OAI-PMH, the administrator provides required information directly or indicates how metadata can be retrieved from the database. General information includes database connection parameters or identification data for the server to be generated, such as the name and email address of its administrator. The administrator also provides SQL queries needed to obtain metadata, such as title, author and language, depending on the metadata standard being used. When this data gathering phase is completed, a code generator will proceed to build a completely functional OAI-PMH server. Along with the generated source code, a "deployment description" is also produced with the specifications of the resources needed to make up an application. The resulting object code and the deployment description are organized into an

JCDL'06, June 11-15, 2006, Chapel Hill, North Carolina, USA.

appropriate file structure and placed in the storage space requested by the user so the newly created OAI server becomes accessible to the public.

Implementation. Voai is currently available (from http://ict.udlap.mx/software) as a set of Java *servlets*. Form-based interfaces allow for the specification of server parameters and database queries, as well as for code generation. Templates are provided for Dublin Core, which is a basic requirement of the OAI-PMH protocol, but mappings from the collections to other metadata standards may also be specified in flexible manners. The generated code is placed in a web server's servlet container. Our tests have been conducted with the Tomcat web server. Although we have used mainly MySQL so far, we have relied on the JDBC connectivity standard, so switching RDBMSs is possible without any changes to the code generated by Voai.

4. XOAI—THE XML DATABASE CASE

Even though a myriad of XML documents are available from multiple sources, building OAI servers for sharing their metadata requires significant work. Our Xoai project focuses particularly on collections stored on so-called native XML databases (for XML-enabled databases, a tool such as Voai could do the job). The approach of Xoai is different from that of Voai mainly in that Xoai does not generate new code for servers to be used with each distinct collection. Instead, a canonical OAI server is provided as part of the software that is distributed for installation at the data provider's site. This server accesses metadata in a predefined format, which is generated by Xoai.

Conceptual design. The key phase for generating OAI servers in Xoai is mapping the structure of heterogeneous collections to a format that is known by a process that serves OAI-PMH requests. This format consists of a "required" section, which includes the attributes considered by Dublin Core, plus an "optional" section, which covers for the specification of any other metadata standard. The generation of an OAI server in Xoai starts by mapping the description of each document in a collection onto the predefined format. This is accomplished by applying a set of XPath queries specified by the collection's administrator. The next step consist of generating a configuration specification, which includes four major aspects: (1) general information (such as the repository's name, the administrator's email address, etc.), (2) responses to OAI-PMH verbs, which involve XPath and XQuery statements, (3) error handling, and (4) metadata definition, which specify the metadata formats that will be handled by the server.

Implementation. The canonical OAI server provided by Xoai was implemented in Java and uses the XML:DB interface specification. This allows Xoai to generate OAI servers for collections stored on a wide range of XML database systems, as long as an XML:DB interface is applicable. Form-based interfaces allow users to provide data for both the mapping and the configuration phases. XML stylesheets play an important role in producing the necessary responses to each of the six OAI-PMH types of requests for the specified metadata formats, and also in error handling.

5. RESULTS

Both Voai and Xoai are freely available at http://ict.udlap.mx /software. We have demonstrated their functionality and potential by producing metadata servers for collections that are now listed as OAI data providers. Voai has been used to generate OAI-PMH servers for three diverse collections: a digital theses repository (http://ict.udlap.mx: 9090/Tales/Oai_tesis) and a collection of over 130 digitized rare books (http://ict.udlap.mx:9090/CIText/Oai_citext), both of which are managed by the University of the Americas Puebla, Mexico, as well as a collection of incunabula, which resides at University of Valladolid, Spain (http://gutenberg.dcs.fi.uva.es:8080/ OAI_UVA/Oai_incunables). In all three cases, collections are modeled and stored as relational databases using the MySQL DBMS.

Voai supports server generation for collections of arbitrary structures, with metadata modeled and stored as relational databases. Target collections may consist of multiple tables and the database schema does not need to be modified in order to share metadata. The user who builds an OAI server with Voai does not require to write any programs or to handle complex configuration files. The resulting server code becomes immediately accessible from the global network and can be validated to become an official data provider.

Xoai was initially tested with an XML version of the thesis collection described above. After its release for evaluation, it has been used successfully used to generate an OAI server for a collection of manuscripts held by the Historic Library of the University of Valladolid, Spain (http://gutenberg.dcs.fi.uva.es: 8080/uvaoai/proveedor), which has been added to the community of OAI data providers. Specific details of the server generation for both of these collections are available in [2]. In essence, we were able to generate these OAI servers in an expedite manner by using Xoai's user interface to extract metadata from documents, store them in a pre-defined format, and provide configuration data.

Xoai and Voai are contributing to our goal to provide highly dynamic mechanisms for constructing OAI servers for collections stored in the most popular database models.

6. ONGOING AND FUTURE WORK

We are currently working on a number of areas to improve Voai and Xoai. Voai and Xoai are currently linked to version 2.0 of OAI-PMH and would need to adapt to changes so it can support the process of upgrading to new versions of the protocol. We are considering the formulation of a meta-description of the protocol that may be used in the generation process. That meta-description would be the only component of our generation facility which would need to be updated in the event of a version change. We also plan to use XUpdate in Xoai to more easily reflect changes in the organization of XML collections.

ACKNOWLEDGMENTS

Project supported by the CUDI-Conacyt Program (2004-2005) and CYTED (Project VII.9 RIBIDI).

REFERENCES

1. Villegas, A. 2005. Voai: un generador automático de servidores de metadatos bajo el protocolo OAI-PMH. B. Eng. Thesis. Dept. of Comp. Systs. Eng., UDLA Puebla. (Spanish).

2. Córdova, J. M. 2005. Xoai: Un generador automático de servidores OAI para colecciones en Bases de datos XML. B. Eng. Thesis. Dept. of Comp. Systs Eng., UDLA Puebla. http://ciria.udlap.mx/tesis. (Spanish).