Available online at www.sciencedirect.com
ScienceDirect
The Journal of Systems and Software 81 (2008) 809–822
The Journal of Systems and Software
www.elsevier.com/locate/jss

A Web services-based framework for building componentized digital libraries

Rodrygo L.T. Santos *, Pablo A. Roberto, Marcos André Gonçalves, Alberto H.F. Laender

Department of Computer Science, Federal University of Minas Gerais, 31270-901 Belo Horizonte, MG, Brazil

Received 17 July 2007; accepted 20 July 2007
Available online 8 August 2007

Abstract

We present a new Web services-based framework for building componentized digital libraries (DLs). We particularly demonstrate how traditional RDBMS technology can be easily deployed to support several common digital library services. Configuration and customization of the framework to build specialized systems is supported by a wizard-like tool which is based on a generic metamodel for DLs. Such a tool implements a workflow process that segments the DL design tasks into well-defined steps and drives the designer along these steps. Both the framework and the configuration tool are evaluated in terms of several performance and usability criteria. Our experimental evaluation demonstrates the feasibility and superior performance of our framework, as well as the effectiveness of the wizard tool for setting up DLs.

Keywords: Digital libraries; Web services; Component-based software development; Software evaluation

1. Introduction

Digital libraries (DLs) are among the most advanced and complex types of information system, going far beyond search engines, since they offer many other valued services. They are normally designed for specific user communities, which must be involved with many aspects, from specification to utilization, in order to guarantee their success (Fox and Marchionini, 1998).

Many of the existing DLs are based on monolithic architectures and their development projects are characterized by intensive cycles of design, implementation and tests (Suleman, 2002). Several have been built from scratch, aiming to meet the requirements of a particular community or organization (for instance, see Laender et al., 2004).

One way to deal with these issues is through the creation of specific software component toolkits, in which each component is responsible for a small part of the functionality of a DL and integrates with other components in order to build a complete system (Suleman, 2002). Such toolkits offer a generic, extensible and reusable framework for building DLs, allowing, for example, to reduce the necessary effort to develop them.

In this article, we present WS-ODL, a new Web services-based framework to build componentized digital libraries. The framework components operate on top of the Fedora architecture (Lagoze et al., 2006), which provides the framework repository and some basic infrastructure services. All communication among the components and between them and Fedora is done via Web services, using SOAP (Simple Object Access Protocol),[1] which provides advantages like enhanced interoperability and validation of input parameters. Besides Fedora, we make use of standard relational database technology to support the functionality of some components, mainly those providing more advanced services, e.g., searching using structured

* Corresponding author.
E-mail addresses: rodrygo@dcc.ufmg.br (R.L.T. Santos), pabloa@dcc.ufmg.br (P.A. Roberto), mgoncalv@dcc.ufmg.br (M.A. Gonçalves), laender@dcc.ufmg.br (A.H.F. Laender).

[1] http://www.w3.org/TR/soap/.

0164-1212/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2007.07.029

and unstructured data conjointly and multidimensional browsing. Since the framework configuration is not a trivial task, we have also developed a wizard-like tool that guides the user throughout the installation process.

Our experimental evaluation demonstrates that this framework is feasible and that its components present superior performance when compared to those provided by the ODL framework (Suleman, 2002), which inspired the development of ours. The experiments also show that the wizard-based approach is very effective, allowing the users to install a DL system much more easily and rapidly than by doing it via command-line. We also discuss some drawbacks of the framework in its current version, which should be considered when choosing it as a solution to build digital libraries.

In summary, the main contributions of this article are:

- The design and implementation of a software framework for building componentized digital libraries.
- The development of a wizard tool for easily setting up running digital libraries based on the framework.
- An extensive experimental evaluation of both the framework and the wizard tool comprising performance, scalability, and usability issues.

The remainder of this article is organized as follows. In Section 2, we describe related work. In Section 3, we overview the architecture of all components that make up the WS-ODL framework. The wizard-based installation tool is described in Section 4. In Section 5, we discuss the experimental evaluation of our framework. Finally, in Section 6, we present our conclusions and perspectives for future work.

2. Related work

The Open Digital Libraries (ODL) (Suleman, 2002) project was one of the first efforts to advocate a componentized approach for the development of digital libraries. ODL proposed an extension to the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) (Lagoze and Van de Sompel, 2001) to support interactions among digital library components. This framework had some advantages mainly due to the use of the OAI and the software componentry paradigms, like simplicity, openness and reuse. Nevertheless, it presents several problems with respect to performance, scalability and interoperability. Performance issues are due to specific implementation choices which led to scalability problems, mainly regarding the components that implement information retrieval functions. Interoperability issues are related to the use of the XOAI protocol, a not widely adopted OAI-PMH extension developed specifically for that framework.

To overcome some of the deficiencies mentioned above, mainly regarding interoperability, we chose to use the Web services technology in our framework. A Web service is a software system designed to support an interoperable interaction between applications in a network. It communicates with other systems via an interface described in WSDL (Web Service Description Language) and SOAP messages, that are transported using HTTP and XML. SOAP is a protocol for information exchange in a decentralized and distributed environment. Through the use of the XML technology, SOAP defines an extensible framework for message transfer on top of other low level protocols. The framework is designed to be independent of programming models and specific implementation semantics. More details on these technologies can be found on the W3C site.[2]

Fedora (Lagoze et al., 2006) is an extensible repository for storage, management and dissemination of complex objects and their relationships. These objects can have local or distributed content, and can be attached to different ''disseminations'', which makes it possible to have various dynamically created representations of a given object. The relationships among the objects are expressed in RDF (Resource Description Framework) and stored in the Kowari triplestore.[3] The architecture is implemented as a Web service and provides the basis for the construction of other applications, like a digital library, for example. Our framework supports the construction of a DL on top of Fedora while improving part of its functionality, as will be shown. It should be emphasized that other repository infrastructures could also be used, provided that they could explore Web services intra-communication facilities.

Regarding implementation choices, we chose to explore open source standards and languages such as Java and traditional DBMS technology, e.g., MySQL. In particular, we wanted to investigate how much support traditional DBMS technology could give to the implementation of the functionality of our components. In this context, Grossman et al. (1997) showed that it is possible to use the standard relational model to implement an information retrieval system and that such a system, in specific scenarios, presents good performance and scales. Moreover, it makes it possible to integrate structured and unstructured data, which provides a more powerful search mechanism, comparatively to traditional search engines. This mechanism is very interesting to a digital library, where users have needs more specific than those of a common Web user. We adopt, adapt, and extend these ideas by showing how several other DL common services can be implemented using the same technology.

With respect to the task of setting up components in order to build digital libraries, some tools have also been described in the literature. BLOX (Suleman et al., 2005) is a tool that hides most of the complexity involved in the task of configuring distributed componentized digital libraries. However, during the configuration task, users interact with this tool in an unguided manner: its interface

[2] http://www.w3.org/.
[3] http://www.kowari.org/.