

Towards a Repository-independent Implementation of Digital Object Prototypes

Kostas Saidis¹, George Pyrounakis²
saiko@di.uoa.gr, forky@libadm.uoa.gr

¹ Department of Informatics and Telecommunications

² Libraries Computer Center

University of Athens

University Campus, Athens, 157 84, Greece

Extended Abstract

According to Kahn and Wilensky [1], a digital object refers to a human generated artifact consisting of four components, namely its metadata, files, relationships and behaviors. Several digital object encoding formats exist, such as METS [4], FOXML [3] and MPEG21 DIP [5], that are capable of storing any type of digital object, theoretically, representing any possible variation of the aforementioned object's constituent parts. However, the respective digital object typing information, that is, what parts constitute each object and how each one behave, is not realized in a manner suitable for effective usage by the higher level components of the Digital Library (DL). This forces the DL users (cataloguers, designers and developers) to interpret typing information manually.

In order to provide a uniform resolution of digital object typing issues in an automated manner, we've introduced the notion of Digital Object Prototypes (DOPs) [2]. DOPs operate on top of the underlying digital object repository and storage layer and provide a tangible realization of digital object types, enabling DL designers to generate *user-defined* types of digital objects. Individual digital objects are treated as instances of their respective prototype and automatically conform to their type-specific specifications. Since all digital object typing information is expressed in terms of prototypes, the addition of a new object type is performed by the generation of its corresponding prototype, requiring no further code development or custom implementation.

Our current implementation of DOPs is deployed in Pergamos, the Digital Library of the University of Athens. Pergamos contains more than twenty (20) disparate digital object types, belonging to five (5) distinct collections, namely Theatrical Collection, Medical Images Collection, Historical Archive, Folklore Collection and Papyri Collection. Each individual object type is realized in its corresponding digital object prototype: `uoadl.medical.image`, `uoadl.theatre.album`, `uoadl.folklore.notebook`, `uoadl.folklore.section`, `uoadl.folklore.page`, and `uoadl.theatre.photo` to name a few. Pergamos high level DL services, such as User Interface, Browsing, Cataloging, Batch Content Ingestion and Automated Content Conversion reside in a single, uniform implementation that is able to operate on all Pergamos material and adapt to each individual object's specifications and requirements by utilizing the object's respective prototype.

DOPs are currently implemented on top of Fedora [6], the repository used in Pergamos. Our presentation will provide an overview of the deployment of DOPs in Pergamos and will focus on our current effort to generate a repository-independent DOPs implementation that could operate on top of any digital object repository system.

References

1. R. Kahn and R. Wilensky. *A Framework for Distributed Digital Object Services*. Corporation of National Research Initiative - Reston USA, 1995. Available at <http://www.cnri.reston.va.us/k-w.html>.
2. K. Saidis, G. Pyrounakis, and M. Nikolaidou. On the effective manipulation of digital objects: A prototype-based instantiation approach. In *Proceedings of the 9th European Conference on Digital Libraries (ECDL 2005)*, pages 26–37, 2005.
3. *Introduction to Fedora Object XML*. Fedora Project. Available at <http://www.fedora.info/>.
4. *METS: An Overview & Tutorial*. Library of Congress.
5. T. Staples, R. Wayland, and S. Payette. Using mpeg-21 dip and niso openurl for the dynamic dissemination of complex digital objects in the los alamos national laboratory digital library. *D-Lib Magazine*, 10(2), February 2004.
6. T. Staples, R. Wayland, and S. Payette. The fedora project: An open-source digital object repository management system. *D-Lib Magazine*, 9(4), April 2003.