RODA

A service-oriented repository to preserve authentic digital objects

Miguel Ferreira & José Carlos Ramalho
University Of Minho

Rui Castro, Luís Faria, Francisco Barbedo, Cecília Henriques & Luís Corujo
Directorate-General of The Portuguese Archives

In mid 2006, the Portuguese National Archives (Directorate-General of the Portuguese Archives) launched a project called RODA (Repository of Authentic Digital Objects) aiming at identifying and bringing together all the necessary technology, human resources and political support to carry out long-term preservation of digital materials being produced by the Portuguese public administration.

As part of the original goals of RODA was the development of a digital repository capable of ingesting, managing and providing access to the various types of digital objects[1] produced by national public institutions. The development of such repository should be supported by open-source technologies and, as much as possible, be based on existing standards such as the OAIS [1], METS [2, 3], EAD [4] and PREMIS [5]. Since RODA is nearly finished, this communication aims at describing its main results.

The beginning stages of development

In the beginning of the project, a collection of functional requisites was assembled by RODA's archival team and a study on available repository platforms was conducted. In this study, DSpace [6] and Fedora Commons [7] were compared against this collection of requisites.

The results of this study allowed us to conclude that, although DSpace, as it comes out of the box, combines a broader range of ready-to-use features and user-friendly interfaces, it lacks flexibility and expansibility. One very pragmatic example is the support metadata schemas other than Dublin Core. One would have to go through a tremendous amount of work to make DSpace compatible with more complex descriptive metadata structures such as EAD [4]. Moreover, it is not entirely accurate to think of Fedora Commons as a digital repository software. It is more of a development platform, one that provides a very basic set of services on which a complete repository solution can be built upon (Figure 1). Due to its flexibility, Fedora was considered to be most adequate option to fulfil the goals of the project and the ideal platform to support the development of more specific functionality that was not present in neither of the evaluated applications.

Figure 1 depicts the overall architecture of RODA's repository. At the bottom of the figure one may find the basic services provided by Fedora. These account for elementary tasks at the Data Management and Archival Storage level. Examples of such services are *ingest*, *add a data stream to an object*, *get data stream*, *purge an object*, *find objects* and *list data streams*. For a

[1] RODA currently supports the preservation of various file formats belonging to the following object classes: text-documents, raster images, relational databases, video and audio.

complete list of the services provided by Fedora, please visit [8]. Fedora search capability is supported by Apache Lucene and its authentication procedures go through a LDAP server (Lightweight Directory Access Protocol).

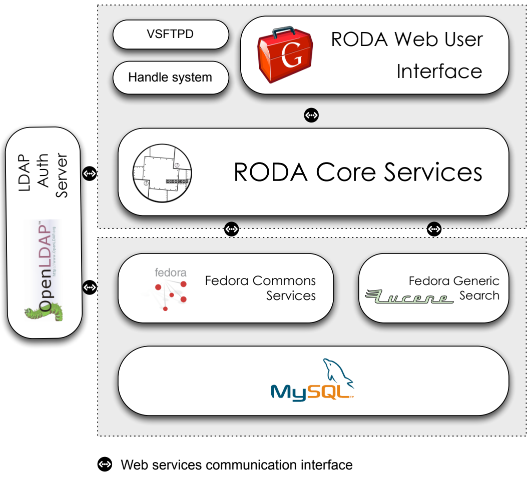

Figure 1: RODA's service oriented architecture.

RODA Core Services are responsible for carrying out more complex tasks such as the complete set of actions that compose the ingest workflow, querying the repository in more advanced and abstract ways and carrying out administrative functions on the repository. The same LDAP server previously described is used by RODA's Core Services for authenticating repository users.

On top of the RODA's Core Services lays the RODA's Web User Interface (RODA-WUI). This layer handles all the aspects of the graphic user interface for producers, consumers, archivists, system administrators and preservation experts. The RODA-WUI components are supported by the Google Web Toolkit and all communication is done via AJAX and Web services.

New entries to the repository come in the shape of Submission Information Packages (SIP). When the *ingest* process terminates, SIPs are transformed into Archival Information Packages (AIP), i.e. the actual packages that will be kept in the repository. Associated with the AIP is the structural, technical and preservation metadata, as they are essential for carrying out preservation activities.

The SIP is composed of one or more digital representations and all of the associated metadata, packaged inside a METS envelope. Producers take advantage of a small application called RODA-in that allows them to create these packages. The structure of a SIP supported by RODA is depicted in Figure 2. The RODA SIP is basically a compressed ZIP file containing a METS envelope, the set of files that compose the representations and a series of metadata records. Within the SIP there should be at least one descriptive metadata record in EAD-Component[2] format.

[2] An EAD record does not describe a single representation. In fact, EAD is used to describe an entire collection of representations. In the SIP is included only a segment of EAD that is sufficient to describe one representation, i.e. a <c> element and all its sub-elements. The team has called this subset of the EAD an EAD-Component.

One may also find preservation and technical metadata inside a submission package, although this last set of metadata is not mandatory as is seldom created by producers. Nevertheless, it was felt important that RODA should support those additional SIP elements for special situations such as repository succession, e.g. when ingested items belong to a repository that is to be deactivated.

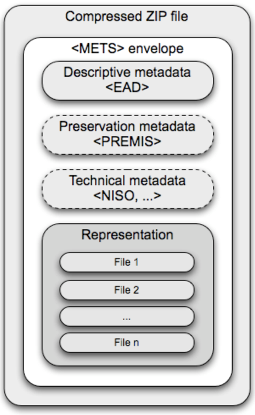
Compressed ZIP file
<METS> envelope
Descriptive metadata <EAD>
Preservation metadata <PREMIS>
Technical metadata <NISO, ...>
Representation
File 1
File 2
...
File n

Figure 2: Structure of a Submission Information Package.

Before SIPs can be fully incorporated into the repository they are submitted to a series of tests to assess its integrity, completeness and conformity to the ingest policy. After decompressing the SIP, the validation process performs the set of tasks described in Table 1.

Table 1: SIP validation steps during ingest.

Step	Name	Description
1	Virus check	SIPs are checked for viruses. Clam anti-virus is being used under the hood to perform this task.
2	Envelope syntax check	Verify that the METS envelope is well formed.
3	SIP completeness check	Check if all files referred in the METS envelope exist within the SIP.
4	File integrity check	Files included in the SIP are accompanied by a checksum string. This information is used to check if any of the files have suffered corruption of some sort.
5	Descriptive metadata check	Verify that an EAD-component is included in the SIP and that its syntax is correct.
6	Preservation metadata check	Check if a PREMIS record has been included in the SIP and that its syntax is correct.
7	Representation check	Verify that at least one representation exists within the SIP.
8	Representation check	Depending on the type of the representation in the SIP, a series of more specific tests are conducted to verify if the representation is complete and format-wise compliant with the ingest policy in place.
9	Specific representation check	Depending on the type of the representation in the SIP, a series of more specific tests are conducted to verify if the representation is complete and format-wise compliant with the ingest policy in place.
10	Normalization	Representations whose format does not conform to the preservation formats defined by the preservation policy are automatically converted to the correct format. The original representation is maintained by the repository for diplomatic reasons.

After a successful validation, the SIP is then taken apart and each of its constituents is integrated into the repository. After this procedure, the SIP becomes an AIP and is ready to be disseminated to potential consumers that have clearance to access that information.

The consumer is able to browse over available collections to view or download digital representations kept in the repository. Depending on the type of the digital object, different viewers or disseminators are used. For example, text documents are delivered to consumers without resorting to any particular artefacts. They are delivered in PDF format, so the consumer should use its favourite PDF viewing application. Documents composed of several images (such as digitised works) on the other hand are displayed in a special Web viewing applications that allow consumers to navigate through the pages of the representation.

RODA's content model is atomistic and very much PREMIS-oriented (Figure 3). Each intellectual entity is described by an EAD-component metadata record (DO nodes in Figure 3). These records are organized hierarchically in order to constitute a full archival description but are kept separately within the Fedora Commons content model. Relationships between EAD-components are created using Fedora's own RDF linking mechanism.

Additionally, each leaf record (i.e. a file or an item) is linked to a representation object (RO nodes in the figure), i.e. a fedora object that embeds all the files and bit-streams that compose the digital representation. Finally, each of these objects are linked together by a set of PREMIS entities that maintain information about the digital object's provenance and history of events (PO nodes).

Each preservation event that takes place inside the repository is recorded as a new preservation-event node (i.e. PO event nodes in the figure). Special events, like format migrations, establish relationships between two preservation-representation nodes. These are called linking events. Each preservation event is executed by an agent, whether this be a system user or an automatically triggered software application. The agent that triggered the event is recorded in PO agent nodes.

Preservation management within RODA is handled by scheduled events. The preservation expert defines the set of rules that trigger specific preservation actions and when these should take place. Preservation actions comply to a common API, so creating and installing new preservation actions in the repository is as easy as copying the programme file[3] to the correct directory on the server. Preservation actions include format converters, checksum verifications, reporting tools (e.g. to automatically send SIP acceptance/rejection emails), etc.

RODA has been developed to be a complete digital repository providing functionality for all the main units that compose the OAIS reference model. RODA fully implements an Ingest workflow that not only validates SIPs, but also takes care of the whole negotiation process between the archive and the producers of information. RODA also accounts for Access providing different ways to search and navigate over available metadata as well as visualizing and downloading stored digital objects. Administration components were also developed allowing archivists to change the descriptive metadata and define rules for preservation interventions such as scheduling integrity checks on all stored digital objects, initiate a migration process, or control which users or groups are authorized to perform certain actions within the repository. All actions performed in the repository by any user are logged for security reasons. Every user must be authenticated in order to use the repository.

[3] In the context of RODA programme files are typically JAR files.

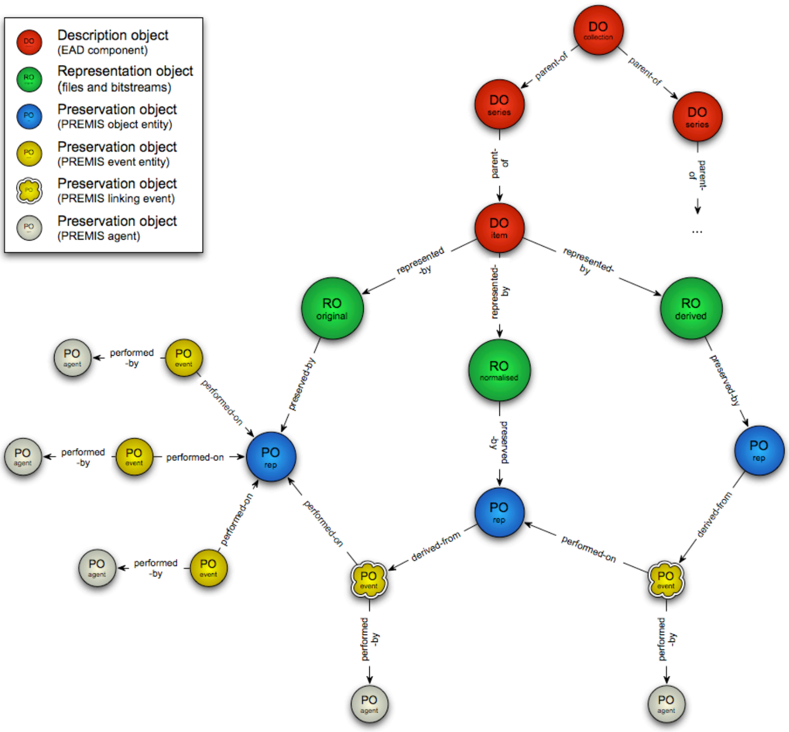

Figure 3: RODA's molecular content model featuring EAD-components, digital representations and PREMIS objects.

References

1. Consultative Committee for Space Data Systems, *Reference Model for an Open Archival Information System (OAIS) - Blue Book*. 2002, Washington: National Aeronautics and Space Administration.

2. Library of Congress. *METS - Metadata Encoding & Transmission Standard.* [cited 2008 2008-04-21]; Available from: http://www.loc.gov/standards/mets/.

3. McDonough, J.P., *METS: standardized encoding for digital library objects*. International Journal on Digital Libraries, 2006. **6**(2).

4. Library of Congress. *EAD - Encoded Archival Description*. 1998 [cited 2008-04-21]; Available from: http://www.loc.gov/ead/.

5. PREMIS Working Group, *Data dictionary for preservation metadata: final report of the PREMIS Working Group*. 2005, OCLC Online Computer Library Center & Research Libraries Group: Dublin, Ohio, USA.

6. Hewlett-Packard Company and MIT Libraries. *DSpace Web site*. [cited 2008 2008-04-21]; Available from: http://www.dspace.org.

7. University of Virginia and Cornell University. *Fedora Commons Web site.* [cited 2008 2008-04-21]; Available from: http://www.fedora.info/.

8. Fedora Commons. *Fedora Access and Management Web Services - API Documentation.* [cited 2007-11-05]; Available from: http://www.fedora.info/definitions/api/.

9. Ramalho, J.C., et al. *Relational Database Preservation through XML modelling.* in *Extreme Markup Languages*. 2007. Montréal, Québec, Canada.