UCLouvain

École polytechnique de Louvain

Open Source Repository for Open Educational Resources

Author: **Noémie Jamar**
Supervisors: **Yves Deville, Christine Jacqmot**
Readers: **Olivier Goletti, Kim Mens**
Academic year 2020–2021
Master [120] in Computer Science and Engineering

Abstract

In the context of UCLouvain Digital University program, an Open Educational Resources repository has been setup to ensure open long-term access to digital learning materials and to promote their dissemination. The current repository uses DSpace, a widely implemented open source platform.

The objective of this work is to assess available open-source repository frameworks to support UCLouvain Open Education initiative. Next version of DSpace repository will be compared to Islandora and Hyrax current versions associated with Fedora Commons as repository.
Improving the user experience of the OER repository, facilitating their production, enhancing their educational metadata and setting automated dissemination of resources are the improvement objectives. Comparison shows that the next version of DSpace will provide the expected functional and technical solutions.

Acknowledgements

I would like to express my sincere gratitude to Dr. Jacqmot and Pr. Deville for their constant support and their constructive remarks. Many thanks for the time they devoted to our regular meetings and for the orientations of this project which they encouraged me to formalise and explore.

I would like to thank the other jury members, Pr. Mens and Mr. Goletti for the time they accepted to devote to review this master thesis.

I am extremely grateful to all the UCLouvain professors for the learning materials that allowed me to apprehend the different domains to analyse.

I also express my thanks to my family for their continuous support.

Contents

Chapter 1

Introduction

The development of knowledge and education is key for an innovative society and cooperation is an enabler for achieving this goal. In *Open Science*, the whole educational community is committed to collaborative development of shared knowledge for the benefit of all [1]. *Open Science* relies on transparency and open access to publications and research data, collaborative sharing of works, exchange of opinions and counter-opinions to build solid and understandable theories. It aims to make knowledge and education accessible and reusable. Open access is a pillar of open science. It allows resources to be freely used, modified and shared by anyone for any purpose. It is applied, among other publications, on educational resources.

Indeed, one important project inside Open Science is Open Education. Its goal is to ensure high-quality education access to everyone without cost barriers and to promote lifelong learning opportunities [21]. It is based on the cooperative development of resources called Open educational resources (OER). Course materials, textbooks, videos, images, software codes complement since digital revolution, the historical vector of knowledge that were books. OER are defined as "Teaching, learning and research materials in any medium – digital or otherwise – that reside in the public domain or have been released under an open license that permits no-cost access, use, adaptation and redistribution by others with no or limited restrictions" by William and Flora Hewlett foundation, an Open science committed foundation [30]. They consist of digitalized educational materials that are with no additional cost available for educational communities in order to be used and reused for teaching and learning.

5Rs actions on OER are set as principles to foster cooperation :

- **Retain:** it is possible to make, own, and control copies of the content (e.g., download, duplicate, store, and manage).

- **Reuse:** it is allowed to use the content in a wide range of ways (e.g., in a class, in a study group, on a website, in a video).
- **Revise:** it is permitted to adapt, adjust, modify, or alter the content itself (e.g., translate the content into another language).
- **Remix:** it is encouraged to combine the original or revised content with other material to create something new.
- **Redistribute:** it is supported to share copies of the original content, your revisions, or your remixes with others (e.g., give a copy of the content)".

The dissemination of OER, to promote their worldwide accessibility, relies on their centralisation through well known OER harvesters and aggregators. OERCommons and Merlot ensure this role of discovering and referencing other sites' resources and organize a world-scaled access to those OERs. Sharing an OER with a Creative Commons[1] license expressing the rights of access to the resource ensures its reuse according to its author decision.

Committed to Open Science and Open Education, UCLouvain developed since several years, the *Digital University* program to promote collaborative creation, dissemination and acquisition of knowledge [28]. The university has undertaken important actions such as the establishment of an open site preserving all the contributions of the community in the field of open educational resources. Thanks to UCLouvain educational community effort, more than 700 OERs are accessible with an addition of 20 resources per month as an on-going delivery effort. Through these OERs worldwide visibility, UCLouvain's international presence and reputation is strengthened. Students have access to resources adapted to their level of learning at their convenience. Teachers have a vast base of materials at their disposal, whether for the presentation of concepts, learning labs or acquisition tests, which they can integrate or adapt as they wish. Sharing courses enriched by years of experience to improve them in collaboration with other professors is a commitment to enhanced quality and corresponds to the open values of the University. Moreover, this open access provide a channel between the university and a larger community of "post-degree" adults who want to continue their long-life development.

This educational effort is not only driven at UCLouvain: it is promoted around the world. UNESCO has clearly stated its support to a worldwide open access to free education [29] and has recently encouraged countries and institutions to devote

[1]Creative Commons:American non-profit organization and international network devoted to educational access. It releases licenses free of charge to the public. These allow authors to communicate which rights they reserve and which rights they authorise.

energies to OER development and organize global cooperation [18]. European Commission is also committed to develop a European Open Science Cloud within its Horizon Europe program [9].

In addition to this political support from society, there are also circumstances that motivate the choice to improve OER repository. The coronavirus crisis has had a positive increase impact on development of digital resources [12]. Indeed, it has deeply disrupted access to education with no in-person lessons during long periods. This has significantly accelerated the digital way to teach and learn. Educational communities have rapidly adapted their traditional teaching methods in favor of inter-mediated courses and self-learning media. Therefore, many new digital resources have been created. This opportunity is a clear incentive to improve and support OER sharing tools such as the one used at UCLouvain called OER@UCLouvain[2].

Currently, the platform is implemented on DSpace version 6. Lots of improvements are announced for the next version 7 but its release date is delayed and not yet fixed. It is therefore important to analyze which improvements are needed and whether competitors are valid solutions in case OER@UCLouvain does not want to wait any longer to obtain improvements. To keep the user experience adequate and to ensure the usefulness of the application, it is important to always evolve by providing up-to-date tools. Therefore, the objective of this work is to propose an evolution strategy for OER@UCLouvain. Here are the steps of the followed methodology:

- Functional improvements and requirements for OER@UCLouvain were listed and presented in a functionality analysis model in order to define comparison criteria.
- The platforms DSpace, Hyrax, Islandora and Fedora were examined to understand their product offerings and the available features.
- Three prototypes were set up in order to compare the different platforms. The comparison criteria resulting from the feature analysis model were evaluated according to a comparison framework. This functional analysis was accomplished through testing and documentation review.
- A technical review was performed on their architecture, external dependencies and maintenance.
- Learning metadata concepts and schemas have been discussed. Several proposals have been made to enhance their quality.

[2] `https://oer.uclouvain.be/`

- Dissemination of OER on a global scale is supported by harvesting protocols. A more in-depth analysis on those protocols was also carried out. A comparison of OAI-PMH and ResourceSync has been made. Interconnection solutions between a learning management system and the repository have been listed.

To conclude this work, a comparison conclusion and a evolution strategy recommendation are proposed.

Chapter 2

Functional Requirements

This chapter presents the current use of the OER@UClouvain platform, reviews its main objectives, its current strengths, difficulties and needs. Finally, it presents the features that an up-to-date OER repository such as OER@UCLouvain must or should have.

2.1 Context

Since 2017, in order to participate in the Open Education movement, UCLouvain has launched its OER repository with the objective of collecting and sharing valuable educational resources. Creation, sharing and dissemination of knowledge is key for the university. Thanks to this global visibility of OER, the UCLouvain's international presence and reputation are reinforced. Moreover, internally, that creation effort is beneficial for the quality of educational programs. As educational resources are open to all, this creates a link between university and a larger community composed not only of students , teachers but also of adults who wish to continue their development through lifelong learning.

The main objectives of OER@UCLouvain are to ensure preservation of OER, to organize open access to open resources for students or self-learners and so to facilitate knowledge dissemination, to provide facility tools to teachers and OER producers and finally to leverage internal creation energy.

Currently, OER@UCLouvain uses DSpace version 6, a widely-installed open-source repository. It contains more than 700 resources, with 20 resources added per month as continued delivery effort.

2.2 Strengths-Weaknesses-Opportunities-Threats (SWOT)

An analysis of strengths, weaknesses, opportunities and threats is illustrated in Figure 2.1 and presented hereafter.

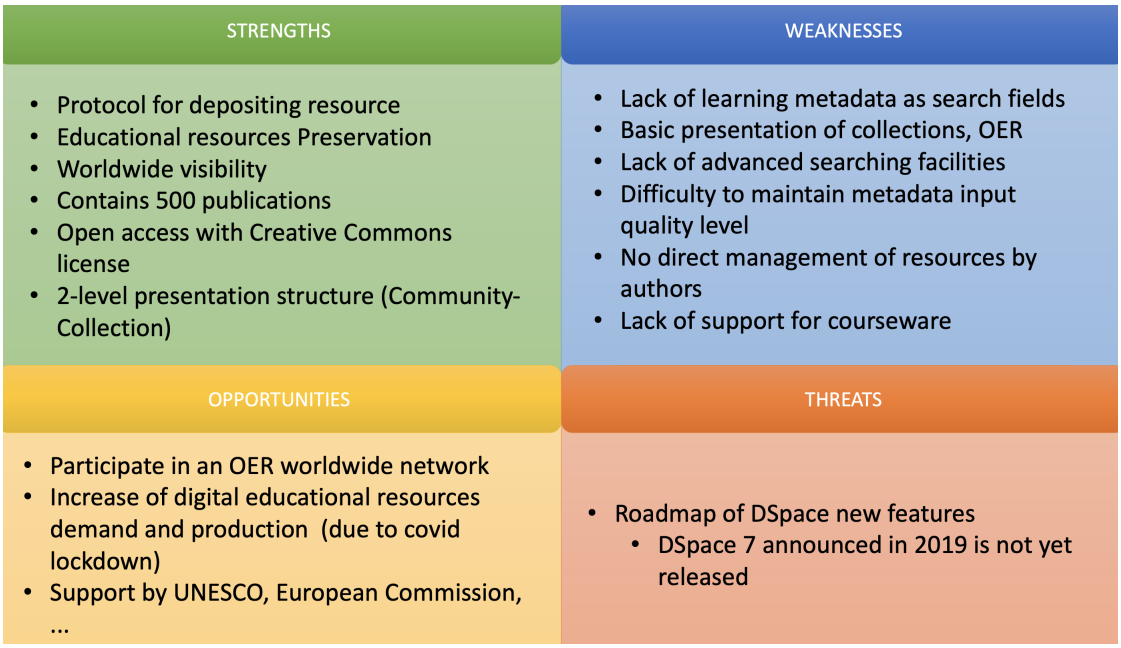


Figure 2.1: SWOT of the current DSpace v6 repository

OER@UCLouvain's main objectives are already met; several elements are considered as strengths and must be kept.

- The protocol for depositing resources is working properly. A submission interface is open to the university educational community members such as professors, assistants, researchers. They have the possibility to deposit their educational work contents in several formats and to associate information as metadata to facilitate their use and search. The submission process is finalized by a validation of a curation team in order to ensure information quality before publication.
- The resources are well preserved and are passed on to aggregators such

as OERCommons[1] or MERLOT[2] with the requested metadata set. These harvesters collect OER worldwide. They give them a global visibility and organise a peer-reviewing with rating.

- Concerning the open access search interface, it is available for any student or self-learner adult, but also for teachers so that they can discover and reuse resources in their own context. A search engine delivers, according to different selection criteria, the list of matching OER where it is possible to select an OER, to view its metadata and then to keep the resource link or to download a selected resource file. To facilitate the search, resources are referenced through a two level (community and collection) classification which reflects adequately university organisation and favors good coordination.

However, several improvements can greatly enhance the user experience whether in the public search interface, in the submission interface and in the personal OER management space for OER creators. In addition, the work of synchronisation with aggregators called harvesting could benefit from automation.

- As an improvement, the OER search interface should be reviewed. Currently, via the community and collection structure, it presents,in an ordered way, the available OERs. The already available advanced selection means are the full-text search or the selection of traditional descriptive metadata criteria: subject, title, author, date. However, the user searching for educational resources could benefit from more efficient tools better adapted to the field of OERs. His search would be more conclusive if he could use filtering criteria such as those related to the type of training, the target audience, the learning interaction mode or the learning level prerequisites. In addition, the presentation of new available and highlighted OER resources could also foster a long-term relationship with the site user. The language presentation could derived from the browsing context of the site visitor. These effects would improve the user experience in his discovery and search for educational resources journey.

- With the aim of improving OER search efficiency, it is important to provide intuitive tools such as metadata clear definition and controlled vocabulary to authors and repository content administrator to help them enhancing metadata quality level. Those metadata definition and usage facility are important features to facilitate the submission and content curation processes.

[1]OERCommons : OER aggregator website `https://www.oercommons.org`
[2]MERLOT : OER aggregator website `https://www.merlot.org/merlot/`

- The provision of a personal OER management space for authors would bring more efficiency. Currently, curators are the only ones entitled to apply changes to the content of existing OERs.
- Via the OAI-PMH harvesting protocol, it is possible to extract automatically public resources from the repository and prepare them with their metadata to be harvested by aggregators repositories. Those harvesters will collect UCLouvain's OER regularly and check their availability.

Moreover, although all current available format resources can be deposited and be downloaded from the public search interface, only some formats can be viewed such as text or video. As external point of attention, one can question the DSpace evolution roadmap as the new DSpace version 7 that was announced to be delivered in 2019 is, in April 2021, not yet released. In fact, DSpace is experiencing an extensive functional and technological review. However, this delay cannot block the improvements of the UCLouvain platform. Meanwhile, other similar products are progressing: Hyrax or Islandora associated with the repository called Fedora Commons. It is therefore important to compare the new features promised by DSpace version 7 to those of other open source solutions.

In order to set the comparison criteria, the complete list of all features that an OER repository must fulfill, is presented in the next paragraph.

2.3 OER repository features

Features that an up-to-date OER repository must or should have, have been listed in order to set the comparison criteria between DSpace next version and its competitors [20]. They are listed in five main categories such as resource storage, resource searching, harvesting, resource management and repository management according to the mind map structure presented in Figure 2.2 and explained hereafter.

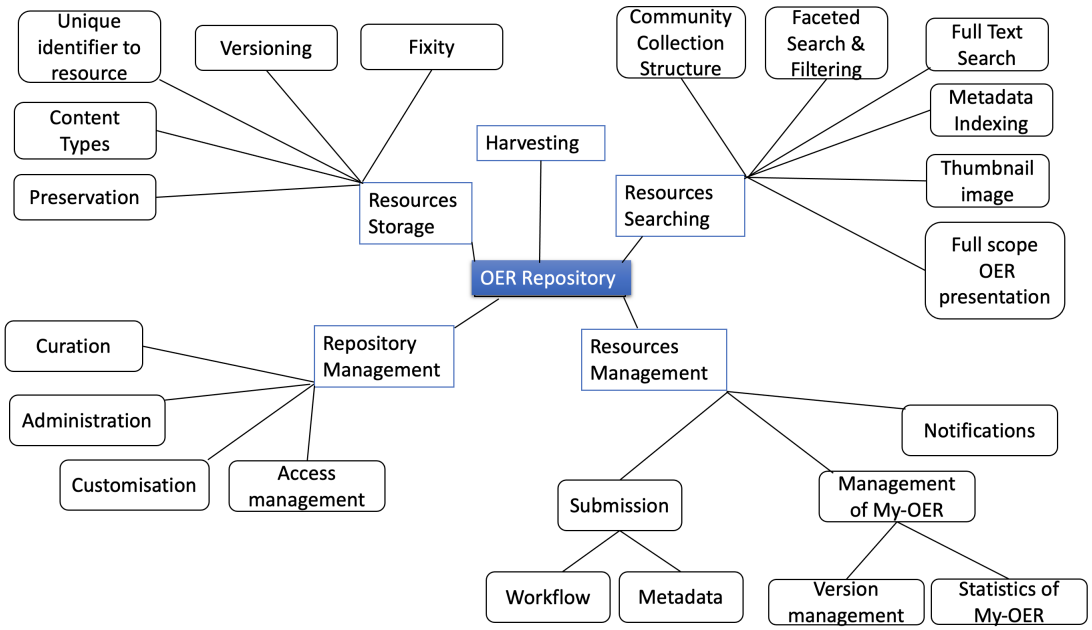


Figure 2.2: Required Features for an OER repository

2.3.1 Resources Storage

Digital course materials produced at the university have significant knowledge content value that must be preserved over the long term. This preservation feature requires that all technical informations to access the resource are kept with it [19]. Moreover, resources have to be uniquely identified. Through that unique identifier, interested users can keep an URI[3] access link to the OER from outside the repository. This allows those users to reuse educational material without having to copy it. This unique identifier can be materialised via an handle[4] in the platform or via DOI[5], both are mechanisms to set the access but DOI allows relocation of resource. In addition, all current types of content must be allowed and adding new type, in the future, is a requested capability.

From another standpoint, it is also important to support the evolutionary dynamics of OERs. It must be possible for the authors to adapt them and to re-publish them, which requires the repository capacity to maintain several visible versions. However, access to older versions must still be possible by external users.

[3]URI: Uniform Resource Identifier
[4]handle: access process organised within an application
[5]DOI: permanent identification of a published electronic resource: www.DOI.org

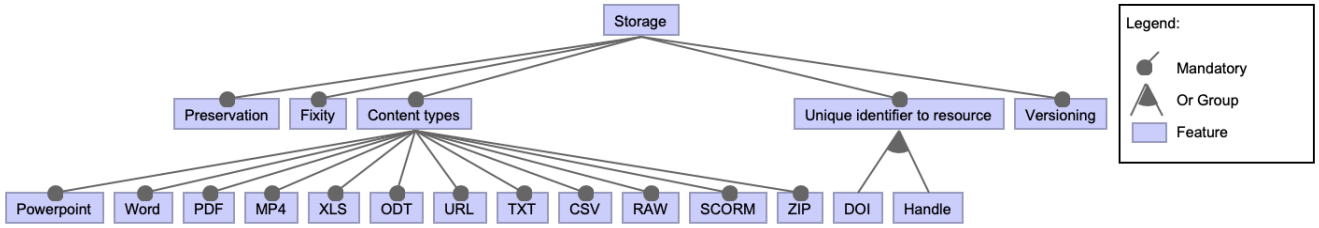


Figure 2.3: Storage features modelling

In addition, OER integrity has to be guaranteed. This control is called fixity. Using on-the-fly checksum control, it is possible to detect any inappropriate alteration on the resource and restore the correct file.

To sum up, preservation, unique identifier, fixity, versioning, download support for all content type are requested features.

2.3.2 Resources Searching

Disseminating knowledge requires powerful search tools to find the relevant educational resource. This search is facilitated by the use of adequate metadata which must be easily and universally understandable.

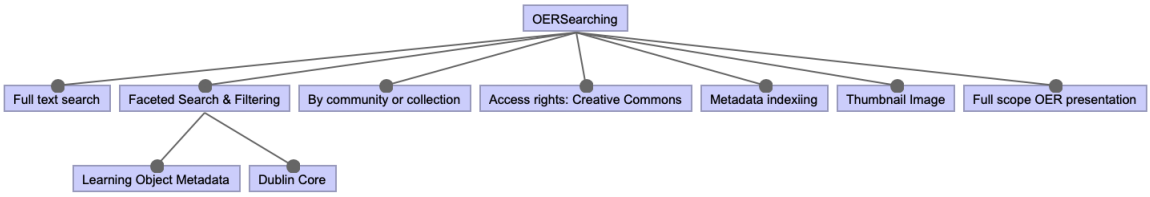


Figure 2.4: Resources searching features modelling

The search interface is a key element of the website. It is open to all. It is therefore important to define a comprehensive interface with up-to-date techniques. It must be attractive with thumbnail images to represent the OER, with an easy browseable presentation adapted in the user-context language or at least in English. Communities and collections of OER must be kept as they propose a nice discovery of the repository for site users. To make the search more interactive and efficient, in addition to a full-text search[6], a faceted selection on specific learning

[6]full-text search : facility for text format file, PDF, OCR recognised text but not for audio or image, video files

metadata criteria will optimise the search interaction by presenting relevant OER in a user-customised sorted order. For the purpose of easy selection, metadata inside value list or with autocomplete support are also valuable feature.

To establish a long term interaction, if someone is interested in an OER collection, it must be feasible to subscribe to notifications when new OERs are published.

In addition to its thumbnail, an OER can be presented as a short description showing the essential metadata for the search. If the user chooses the OER, he can then see all the descriptive metadata and files that he can either download or visualize directly as PDF or video.

In particular, a specific metadata of access rights is always associated with each OER resource in order to define how the author allows user to interact with the resource. The Creative commons license permits to define exactly what the author allows. By definition, OER are open access but the author can decide if it can be referenced, be reused or remixed to create new resources.

Metadata classifications are prerequisites for smart indexing, searching, and harvesting of resources. To make the user search journey pleasant and efficient, the OER repository must use adequate standard and customisable metadata schema. Indeed, the full support of Dublin Core [7] (DC) as simple and universal standard and Learning Object Metadata [8] (LOM) as educational classification is mandatory. Those metadata schemas are presented in the chapter concerning metadata.

Moreover, it must be possible to add new metadata that can be used as index and selectable in OER description. The search interface has also to be easily modified to add selection criteria according to the chosen metadata schema.

To summarize, several features are required: new metadata indexing, full text search, faceted search, DC and LOM metadata full support, thumbnail image rendering, two level community-collection classification, full scope presentation of the OER and its metadata in a non technical form.

2.3.3 Harvesting

Another way to organise OER dissemination is through a worldwide collaborative harvesting network. Therefore, in the OER repository, the OAI-PMH protocol

[7] Dublin Core or DC: set of basic descriptive metadata
[8] LOM : set of learning content descriptive metadata

support is mandatory as it gathers all published OER, bundled with their metadata and propose them to harvesters.

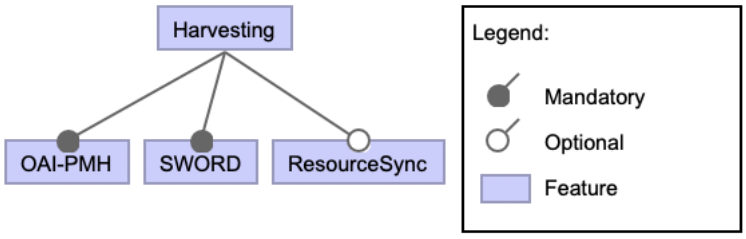


Figure 2.5: Harvesting features modelling

Other optimised protocols like ResourceSync would be of interest for future usage if OER harvesters would adopt it.

Another protocol that has to be possibly activated, is SWORD. It allows one by one transfer of resource from another learning system such as Moodle. It would give to the educational community a facility to push internal learning resources to the OER repository.

Therefore, support for OAI-PMH and SWORD are required. ResourceSync is optional.

2.3.4 Resources Management

The transmission of open access knowledge is based on the creative energy of UCLouvain educational community. It is important that administration facilities are provided to them. OER authors use a submission interface and a personal

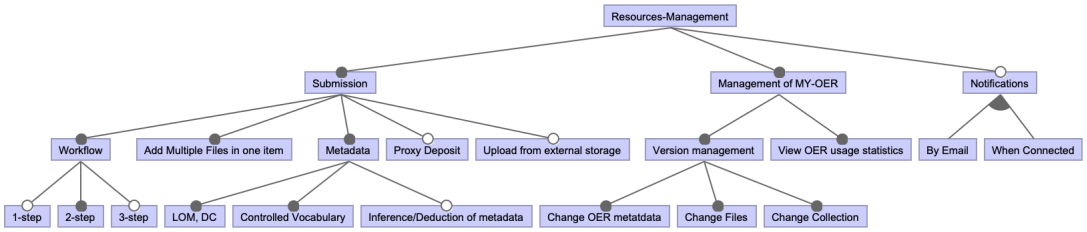


Figure 2.6: Resources Management features modelling

space to manage their own resources. In the former interface, they can submit an

OER that is composed of one or several files that are ingested in the repository. Ideally, the ingestion process is optimised so that technical metadata are inferred automatically. Moreover, some additive files can be automatically generated and proposed to the author such as thumbnail, extracted text, down-sampled audio. Other required submission facilities are zip file uploading and deposit from external storage sources such as dropbox, S3...

Moreover, a lot of facilities should be devoted to the quality of metadata input because the cause of harvesting or search problems is often misunderstanding. Several features are very useful: controlled vocabulary (metadata definition, list of accepted vocabulary, auto-complete facility) or csv uploading to easily fill the metadata. Moreover, any customisation capability that allows automated metadata deduction or inference would be an advantage.

To finalise the submission, a validation team assisted by the content responsible curators takes the decision to publish the proposed OER with its describing metadata. This current two-steps deposit workflow (submitter - validation team) matches the actual need.

The personal space at the disposal of authors allows them to manage their own learning resources i.e. to review their full-scope descriptive metadata in a non technical form, to deposit new file in an existing OER and to manage versions of the OERs that have to be adapted and re-published. In that interface, authors should have the opportunity to get feedback on access statistics over their own resources.

To ease submission and management interaction between authors and curators, a notification or mailing system would be an advantage.

To conclude, the requested features are a 2-step submission process, LOM and DC support, controlled vocabulary, multiple file ingestion, versioning management, OER access statistics feedback, my-OER space. The optional features are programmable submission process, metadata inference, proxy-deposit (submission by another person), upload from external storage source and notifications.

2.3.5 Repository management

This interface must have tools for application customisation, access management, content management and performance management.

Several customisations have to be feasible as for community and collection creation,

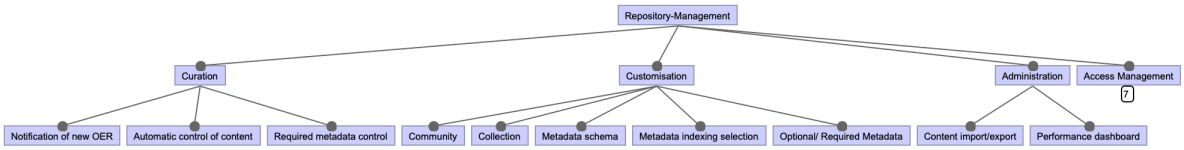


Figure 2.7: Repository management features modelling

metadata schema definition, controlled vocabulary setting, optional and required metadata selection in the submission form or in the simple OER description, metadata indexing selection.

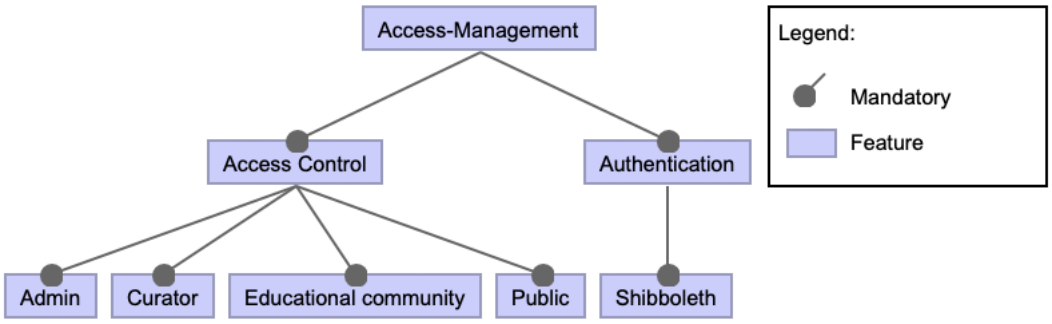


Figure 2.8: Access management features modelling

Submission process role authorisations have to be easily managed in the management interface. In addition, user authentication for the submission interface must be automatically synchronised with shibboleth to facilitate usage and avoid management overhead.

The curator's responsibility is to ensure the overall content quality. As already explained, his role is to validate the publication process after a new OER submission notification. He also manages the life cycle of metadata on the whole repository. He controls that all required metadata are available and relevant, if publishing policies are respected. Automatic scheduling of those required controls has to be present.

The administrator is in charge of the availability and performance of the site and must have all the necessary tools to do so. This is all the more important as the architecture may contain several modules such as the search engine and the resource store which must interact with each other.

Statistics on content access and site usage are valuable feedback information

for administrators and curators.

The required features are customisation capabilities (metadata schema, metadata attribute specification in submission form or presentation form, metadata indexing, community-collection), content quality control mechanisms, control automatic scheduling, performance dashboard, shibboleth authentication, authorisation control (submission access, workflow process) and content import/export treatment.

Chapter 3

Solutions review

After having established the needs of OER@UCLouvain, existing open source repository solutions or versions under construction were examined. As already explained, the new version of DSpace is in the pipeline and has no announced delivery date. It is therefore important to look at the alternatives and their available features. Fedora is known to ensure the preservation of digital content. However, it is not an all-in-one product like DSpace; it relies on other products to build the interfaces: Islandora or Hyrax. Before installing and comparing the platforms, which will be done in the following chapters, here is a quick presentation of the different products: DSpace, Fedora, Hyrax and Islandora on which the comparison will be made.

3.1 DSpace

As a digital content repository software package, DSpace is a widely deployed open source solution for institutions. It was jointly developed by MIT[1] and HP [2] [31] and is widely used in educational institutions, libraries and archives. Its main feature is the preservation of digital content, but it also provides tools for submitting and searching content, as would do a content management system (CMS) tool. In addition, DSpace is positioned as a Learning Object Repository, thanks to its various content format import/export tools .

The community development effort is supported by DuraSpace (a non-profit organisation). Since 2019, DuraSpace has mandated Lyrasis[3] to provide leadership within the support community. Its job is to facilitate a well-organised website for collab-

[1]MIT : Massachusetts Institute of Technology
[2]HP : Hewlett Packard
[3]Lyrasis: non-profit member organisation to support the community

oration inside the supporting community and to provide access to well-designed documentation and reports from the community's working groups. DSpace has about 3000 customer installations. Looking at the client base, there are few purely OER sites; typically, OER@UCLouvain is among the three examples.

DSpace version 7 reenginering project aims to bring two important features: a comprehensive interface managed in Angular and the development of REST-API interface for the backend services. Those two re-engineering targets require long development effort. The v7 release date, initially planned for 2019, is not yet fixed. A testathon has started in April 2021 with a subset of planned features. The planned target of the v7.0 release is currently in August 2021.

3.2 Fedora

Fedora (Flexible Extensible Digital Object Repository Architecture) is an open-source repository system for accessing and preserving digital content [5]. It is used by libraries, universities and digital archives often in conjunction with the Hyrax or Islandora submission or discovery user interfaces.
Originally developed by Cornell University and Virginia University[32] , DuraSpace is now leading its development. DuraSpace has asked Lyrasis/footnoteLyrasis: non-profit member organisation to support the community to organise the Fedora community. Fedora has a three hundred client base and is rarely installed without being part of the Islandora or Hyrax architecture. To ensure its development, Fedora is heavily dependent on these two products, while the latter have built Fedora-free versions for customers for whom preservation is not the primary feature. Fedora, not having a direct relationship with its customers, would like to reduce its supported base of versions and boost its support community. Fedora is by design structured internally with REST-API services and uses a RDF triplestore database to map its repository content. Their interest is presented in the chapter Technical Comparison. The RDF model is presented in appendix.

3.3 Hyrax

Built on the modular REST-API architecture, the initiative that gave birth to Hyrax chose as a principle to assemble around the Fedora repository the components offering the best services such as interface, search, etc. This multi-component architecture quickly generated several branches of differentiated products[34]. In 2017, a unification action took place. The unified Hyrax product offered a complete user interface built with Ruby-on-rails. Hyrax is positioned as a configurable and

expandable open-source repository building toolkit that accommodates multiple use cases: institutional repository, theses, library digital collections and content preservation archives. In 2019, based on Hyrax, a customisable off-the-shelf solution, called Hyku, was designed for repositories without development capabilities [24].

The support community is called Samvera. Its organisation supports workgroups but has difficulty in clearly presenting the different current versions of its products. More than 40 sites are listed as applying the Samvera framework. The community is made up of universities that have supported Samvera from the beginning, but it has been extended by several universities that have joined.

3.4 Islandora

Islandora provides an open-source digital repository solution based on Fedora and Drupal as core modules. Drupal organises the user interface of the submission and search application and Fedora manages the preservation of content. Islandora can be seen as a middleware that manages the transformation into derived content and the synchronisation of data between the two main modules. The complete architecture is positioned as an institutional repository solution.

Islandora was originally developed by the University of Prince Edward Island [33]. Its evolution is now under the responsibility of the Islandora foundation: a non-profit member organisation responsible for supporting Islandora. In 2020, Islandora is used in over 300 implementations.

3.5 Alternatives

As a potential alternative, ORI-OAI, an established solution for French academic institutions, is interesting. It uses a supLOMFR[4] metadata standard, a LOM[5] standard that has been adapted locally to meet the specific needs of higher education. In addition, it has a mechanism for interconnecting Moodle and transferring learning resources into the repository. However, since 2015, there has been no further development of its documentation site, which is why it was not selected. Hyku, a Samvera product that is a turn-the-key bundle of Hyrax features, was not selected because at the time of the choice, it did not have the OAI-PMH harvesting solution.

[4]supLOMFR local application of the LOM standard used for French educational resources
[5]LOM : Learning Object Metadata

Omeka is an alternative user interface that can interact with Fedora and DSpace. It has not been selected because it was not referenced by neither DSpace nor Fedora.

3.6 Summary

As presented in Figure 3.1, DSpace is the most installed repository among the proposed solutions. However, Hyrax and Islandora have more recent releases. In conclusion, the work of comparing new features and compatibility with OER@UCLouvain requirements is done on the basis of the current available DSpace version (v7.0 beta 4), the Hyrax v3/Fedora v4 combination and Islandora v8 using Drupal v9 and Fedora v5.

	DSPACE	Hyrax	Islandora	Fedora™
Creation Date	2002	2008 (Hydra) Hyrax launched in 2017	2006	1998
Creators	HP and MIT	Stanford University, University of Virginia, University of Hull and Fedora Commons	University of Prince Edward Island	Cornell University
Supporting community	DuraSpace	Samvera	Islandora Foundation	DuraSpace
Number of installations	+3000 installations	40 adopters known	331 installations listed	+300 installations
Recent Delivery:	DSpace 6.3 in 2018	Hyrax 2.9 in 2020	Islandora 8 in 2019	Fedora 5.1.1 in 2020

Figure 3.1: Summary of the analysed solutions

Chapter 4

Functional Comparison

Three prototypes were installed in order to compare the functionalities of DSpace, Hyrax and Islandora, the two latter products being combined with Fedora. DSpace and Hyrax installations are turn-key application with a directly usable repository. This is not the case for Islandora, it must first be adapted. This specific customisation in order to build a prototype comparable to the others has been documented in the appendix.

This chapter describes the testing methodology used and explains results from the comparison.

4.1 Methodology

Three prototypes were installed on three different computers in order to avoid incompatibilities between the different software stacks. All choices of OS are available for the three solutions.

- DSpace version 7 beta 4 was installed on a Macbook according to the installation procedure (DSpace v7.0 unreleased Backend + Frontend angular.IO installation November 2020). Since the DSpace v7.0 beta 4. version did not have all the features planned for the v7 project, the features announced but not yet delivered were summarized as a virtual DSpace v7.2 version that is not yet available.
- Hyrax version 3 linked to Fedora version 4 was installed on a Linux computer, using the docker installation package (December 2020).
- Islandora version 8 was installed on a Macbook, in February 2021, following the version of the ISLE (Islandora enterprise) docker-based package.

Islandora requires customisation to prepare a web application similar to OER@UCLouvain. Those actions such as enabling REST-API services, defining themes, views, menus, block layout, require a slight understanding of Drupal.

Features comparison are grouped in the same five categories as in the OER@UCLouvain requirements presentation: resource storage, searching, harvesting, resource management and repository management. By principle, each required feature corresponds to a test that was applied to each prototype. However, some features have been controlled as documented but not tested such as the interconnection with shibboleth, the fixity. Each test result counts for one point in case of the availability of the feature in the prototype, no point otherwise. Some difficulties arose for some features that required extra plugins. In such a case, it was noted in the test results that solutions exist but the test was not validated. In addition, some features available in one solution platform but not part of the requirement have been noted but their weight have been neutralised.

The testing results have been summarised for the three platforms and planned features have complemented the DSpace V7.0 beta4 to establish the planned DSpace v7.2. All test results were summarised in a weighed excel comparison file.

This section will resume the comparison by categories.

4.2 Resources Storage

	Max	DSPACE V7 b.4	DSPACE V7.2	Hyrax	islandora
Content Type	/9	7	9	8	8
Unique Identifier	/1	1	1	1	1
Versioning	/1	0	1	1	1
Preservation	/1	1	1	1	1
Fixity	/1	1	1	1	1
Total	/13	10	13	12	12

Figure 4.1: Resources Storage

Several types of content were ingested into the repository through the submission process: Powerpoint, video, mp4, Excel, PDF, docx, HTML, XML, zip, URL link, rawfile. A post-ingestion content check verified that the resources were correctly associated with the access metadata and that a unique URI was obtained to access

the resource directly. Multiple versions of the same resource were created and then it was verified that it was possible to restore the previous version.

As shown in Figure 4.1, the majority of the different types of content are well supported. In particular, Hyrax was penalised by a Powerpoint file test which was not correctly ingested into our prototype, the result file was empty. Islandora was penalized by the ingestion of zip files that require Drupal add-ons. DSpace beta4 did not yet allow ingestion of zip files, URL links and raw files although it is announced for DSpace v7 and it works in DSpace v6. Preservation is ensured by associating resources with access metadata. Unique identification is also ensured for each one even if the technique used differs: a handle is created at DSpace whereas Fedora manages access to resources via a REST-API service link. In addition, the three products also allow the saving of a DOI, an indirect link managed by a resolver, which makes it possible to maintain a unique DOI link but to be able to move the resource. On another hand, in terms of versioning, the DSpace beta 4 version did not allow version management and Islandora offers a two-level version management. It should be explained that in Islandora, an OER resource corresponds to a Drupal node associated with one or more resources in Fedora. Islandora provides synchronization mechanisms between the two entities but version management is done at these two levels independently, which is not very intuitive.

4.3 Searching and Discovery

	Max	DSPACE V7 b.4	DSPACE V7.2	Hyrax	Islandora
Repository Structure	/1	1	1	1	0
Search and filtering	/3	3	3	3	3
Repository presentation	/3	2	3	3	3
OER metadata standards	/3	3	3	3	2
Total	/10	9	10	10	8

Figure 4.2: Searching and Discovery testing

As presented in Figure 4.2, there are very few differentiating factor in this feature category as all three products use the same search tool Solr. Metadata indexing (DC, Addition of schema as LOM), faceted search and filtering are available to

organise the search interface in a up-to-date adaptive mode. Multi-languages support is also available. Full text search capability is also not a differentiating factor.

However, DSpace beta 4 didn't have the complete admin interface, therefore the full text search and the creation of derivatives resources could not be checked. The extraction of text or thumbnail from the initial file format are done on DSpace at the administration level. Islandora is penalized for two reasons. Firstly, in our prototype, a structure of collections with only one level was implemented. Nesting levels were not obvious to set. Secondly, the Creative Commons licence[1] could not easily be added in the prototype, yet it is documented that it is available.

In addition, but not listed as a requirement, all three products allow for the definition of Google Scholar-specific tags to support better resource discovery. This feature is an enabler to present to external search engines the repository content associated with informations to better classify the resources in the Internet. It is an alternative to harvesting in order to foster discoverability.

4.4 Harvesting

	Max	DSPACE V7 b.4	DSPACE V7.2	Hyrax	Islandora
ResourceSync	/1	0	1	1	0
OAI-PMH	/1	1	1	1	1
SWORD	/1	0	1	1	1
RESTful HTTP API	/1	1	1	1	1
IIIF	/1	0	0	1	1
Total	/5	2	4	5	4

Figure 4.3: Harvesting and other interoperability services

As shown in Figure 4.3, OAI-PMH is available on all three solution platforms. In addition, SWORD, the protocol for exchanging resources one by one, is available on Fedora and DSpace. It is not yet developed in DSpace v7 beta 4, but is currently announced for version 7.2. Moreover, ResourceSync, an enhanced harvesting protocol, is already available on Hyrax and announced in DSpace v7.2.

[1]Creative Commons reference: to indicate which access to the resources is accepted by the author: read, use, remix

In addition, other features are available without being part of the requirements. all three products have a REST-API service interface for developed exchanges with external applications. This feature allows OER@UCLouvain to build its own applications that can interact with the repository. All three have a tripleStore in which resources are presented in an RDF model and on which it is possible to perform SPARQL queries. This feature is important to present the linked data of the repository on Internet. It is not a requirement until harvesters request it. The IIIF[2] standard for adapting images to the presentation interface is a differentiating factor for Hyrax and Islandora, but in the field of OERs it is not an essential requirement. DSpace has discussed the IIIF standard, but it is not planned for version 7.2 so far. This feature would allow OER@UCLouvain to adapt images according to the comprehensive interface whether it is a mobile connection or not. Regarding the integration with ORCID (Open Researcher and Contributor ID) as a control authority for personnel or research project permanent identifier , DSpace and Hyrax have this functionality. Hyrax allows interconnection with Zotero[3] to define citations within OER.

4.5 Resources Management

	Max	DSPACE V7 b.4	DSPACE V7.2	Hyrax	Islandora
Submission	/11	6	8	9	6
Management of my OER	/4	3	4	4	3
Notifications	/2	1	2	1	1
Total	/17	10	14	14	10

Figure 4.4: Resources management testing

Regarding the structure of the content, there are some differences between the three products. An OER is called an item in DSpace, a work in Hyrax, and a content comprising a node associated with resources in Islandora. This is because Islandora requires two definition steps: one for the node in Drupal, and then one for the submission of the stored resource in Fedora. This decoupling is not easily understood by a user coming from DSpace. Furthermore, Hyrax offers nested works while DSpace offers the establishment of relationships between items. The construction of such relationships allows the heritage of metadata. However, it

[2]IIIF: international image interoperability framework
[3]Zotero : open source reference management tool for managing citations, bibliographies, etc.

requires specific development for the modelling of relationships. The datamodels are presented in appendix.

Collections (set of OER) are available in all three products. Nested collections are available in DSpace and Hyrax while they are discussed as target features in Islandora.

Administration over resource sets with specific access rights is called community in DSpace and administrative set in Hyrax. This notion is not available in Islandora where the administration is specific by collection. DSpace is the only solution to offer nesting of communities.

The submission process has been tested with one or more files according to the types listed in the storage category. Beta 4 and the Islandora prototype allow ingestion of one file at a time.

The metadata input grid allows to differentiate between mandatory and optional metadata. Dublin Core is supported, whereas LOM support requires the addition of this schema. Controlled vocabulary can be associated with each metadata in the form of terms list, taxonomy or controlling authority.

Hyrax offers the choice of a one-step (submitter) or two-step (submitter, editor) process defined at the collection level. The editor can comment, add metadata, validate the upload. In Islandora, only the one-step process was natively available. DSpace v7 is proposed with a three-step submission process involving several roles: submitter, editor (comment, metadata), final editor (metadata, validation). Unfortunately, adaptation of the process is not yet possible in beta 4 but is planned for v7.1. Intra-application notifications, which increase process fluidity, are also supported.

An individual interface for managing one's own OERs is available in DSpace and Hyrax. It allows for content, metadata and versioning adaptations by resource. In Islandora, this configuration requires additional customisation.

The integration of usage statistics by resource is available in Hyrax. It requires development in each other products. However, resources access statistics are at the disposal of the repository administrator.

	Max	DSPACE V7 b.4	DSPACE V7.2	Hyrax	Islandora
Curation interface	/3	0	3	3	2
Customisation interface	/5	5	5	5	4
Administrative Dashboard	/2	0	2	2	1
Access Controls	/2	2	2	2	2
Total	/12	7	12	12	9

Figure 4.5: Repository management testing

4.6 Repository Management

In DSpace, the full delivery of the administrative dashboard is postponed to v7.2. This has impacted the test of beta 4 for this category. However, although functions are not yet completed, the administration interface skeleton is built and prove the intention to have all v6 functions. DSpace v6 has standard automatic treatments and designed controls to ease curators work. Hyrax and Islandora don't have an integrated interface as they rely on Fedora command interface to do controls on the repository content.

Concerning Access management, all three products allow same granularities of access (open, from-institution or private) over the content structure elements: collection, work/item/content, or file/media. Regarding authentication, all three can connect to Shibboleth in order to integrate with the central tools of the institution.

Customisation through on/off buttons and simple appearance changes is available in all three tools for customisable elements.

Only Hyrax has designed features to allow depositing on behalf of another person or transferring ownership of resources. However, in the other products, another possible way to manage these transfers of responsibility is through the intervention of the administrator. As not requested features, lease or embargo features are available in DSpace and Hyrax. They allow to limit or postpone resource availability date.

4.7 Conclusion

According to the comparison results, it is clear that the current v7 beta4 is not sufficient to replace the currently installed version 6 of DSpace OER@UCLouvain.

	Max	DSPACE V7 b.4	DSPACE V7.2	Hyrax	Islandora
Resources Storage	/13	10	13	12	12
Searching and Discovery	/10	9	10	10	8
Interoperable Services	/5	2	4	5	4
Resources Management	/17	10	14	14	10
Repository management	/12	7	12	12	9
Total	/57	38	53	53	43

Figure 4.6: Comparison conclusion

Several elements are missing, such as a complete administration interface, versioning, zip and batch file upload, SWORD are only planned for v7.2 which is not yet scheduled. On another hand, Islandora requires more customisation to achieve a solution at the level of what is offered by the other two. Hyrax and DSpace v7-2 are globally equivalent in terms of functionalities. However the curation and repository administration tools which are a real strength in DSpace version 6 should give DSpace v7.2 a clear advantage. The quality of Hyrax documentation is also lower than that of the others. Both Hyrax and Islandora have implemented some interesting features to facilitate resource management, e.g. derivative management, but DSpace is following the trend, with a delay due to its current time-consuming user interface re-engineering project.

Chapter 5

Technical Comparison

This chapter will consider architecture of the compared platforms. The comparison will focus on the user facing interfaces, the machine facing interfaces, the added tools, the way tools and internal modules work together and the storage layer content. Open source solutions are free of charge but require resources to be implemented and maintained. This maintainability aspect must be considered when making a choice. Moreover, the decision must be valid for several years and therefore the adaptability of the solution must also be taken into account. The comparison will point differences and explain how they can affect OER@UCLouvain.

5.1 User-facing interfaces

The compared applications offer the expected interface quality with different zones of dialogue with the user, immediate filtering adaptations to his selections, thumbnail images attached to the presented resources.

The technologies used are Angular for DSpace, Ruby on Rails for Hyrax and Drupal for Islandora. Islandora requires a customization effort while the others offer directly an operational solution. In all three solutions, interface adaptations are possible. However it is not recommended to leave the standard solution and have to support alone the burden of maintenance. The choice of the interface framework should correspond to development skills of a large supporting community which will participate to the common enhancement effort. It is the case for the three choices.

The search interface is very important. In all three products, it is supported by the same external search engine: Solr. This engine indexes the metadata indicated for the resources and ensures the execution of the queries to find the

documents corresponding to the selection filtering. Solr is a good choice.

The submission interface is done in a similar way in Hyrax and DSpace through a media file deposit area and a metadata characterization area with the possibility for the administrator to control the metadata frame to be filled. The submission is done in a less intuitive way in two steps for Islandora: the creation of a Drupal node and then the integration of resources in Fedora.

The administration interface is a factor of differentiation. The choice of DSpace is to propose a centralised curation interface that federates the actions a curator has to do: control the quality of the content, validate new submission, ...
The other two products present several administration interfaces for their different modules. They apply a building blocks policy with the integration of different modules. Therefore, the administrator has to work in several tools.

Concerning the my-OER[1] interface, it is proposed by DSpace and Hyrax while it is not proposed and has to be build in Islandora. DSpace v6 proposes also such an interface.

5.2 Machine interfaces

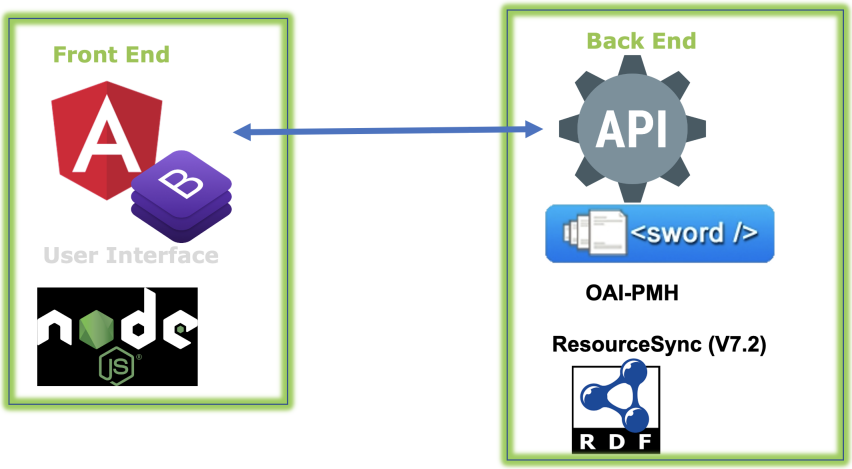


Figure 5.1: DSpace's application layer split architecture

The three solutions provide the same machine interfaces: an OAI-PMH store, SWORD, REST-API and RDF linked data. Figure 5.1 illustrates the DSpace

[1]my-OER: interface where authors can manage their own OER resources

interfaces. The OAI-PMH store interface is available in each product. As explained in the chapter Harvesting, this interface presents repository resources metadata to harvesters' queries. The selection of the metadata schema incorporated in the store is possible, several schema can be selected. The synchronization with the repository content is scheduled on regular basis. Another machine interface permits to import resources one by one. It is used when an external application submits resources to the repository. This is realized by DSpace in SWORD v1 and v2. Fedora provides SWORD v1 for Hyrax and Islandora. The version 2 permits not only the deposit of resources but also to update them. REST-API services are also ready to be used in case UCLouvain would develop external applications that use resources from the repository. Finally, all three products have an interface for accessing RDF linked data model of the repository content. This interface provides to Internet the knowledge about the resources that are part of the repository. Fedora uses the RDF model as its internal data model while DSpace uses a cache RDF representation of its internal data that are modeled in traditional database.

5.3 Internal architecture

The internal architecture of DSpace has been built in a monolithic way as most applications were when DSpace was conceived. DSpace is in the process of modifying its architecture but this effort is done step by step. The new DSpace REST-API in version 7 presents the internal services of the backend modules to the new frontend user interface. This is not a complete redesign of DSpace's internal services in microservices as those of Hyrax and Islandora, but it's an important step to insure better modularity. The design of an application in microservices structure with well designed interfaces is recommended, especially in the case of a community support. It permits cooperation between modules and the knowledge of the application can be easily shared. The Hyrax and Islandora architectures are built on the building block principle where products are chosen rather than built if possible, and then they are integrated through microservices to work together. The microservices communicate internally with each other via REST-API interfaces. This architecture permits an easier scalability. Indeed, services can be duplicated when required and a service can be replaced by a new one if interfaces are respected. They have incorporated several modules in their architecture. In Appendix, a framework of comparison has been applied to all three plateforms. Here are the conclusions of the comparison. DSpace's architecture is less modular but better integrated concerning administration tooling. It has incorporated fewer external modules and those are identical to the choices of Hyrax and Islandora. Their search engine is

Apache Solr [2] and they use a technical metadata extractor called FITS [3]. The modular architecture of Hyrax and Islandora requires robust communication tools. Islandora chooses Apache ActiveMQ[4] and Hyrax uses REDIS[5]. In addition, due to the number of modules, Islandora uses Apache Karaf [6] to orchestrate the execution of all of them. As presented in Figure 5.2, Islandora positions itself as a middleware between the application layer including the user interface supported by Drupal and the storage maintained by Fedora. On the contrary, the two other products have control over their user interface.

The other specificity of the Hyrax and Islandora architectures corresponds to the constraint of synchronisation between integrated tools. Several modules are dedicated to this. In Islandora, an OER resource is modeled via a Drupal node and stored as repository item in Fedora. The data model of the three platforms are in appendix. The link between the two objects must be established and verified for consistency. Some Islandora modules are dedicated to this control.
Other Hyrax and Islandora modules are used for the creation of derived content: optical character recognition, sampling of audio files, creation of thumbnails, extraction of files from several formats. Similarly, DSpace uses the technique of mediafilters[7] in the hand of the curators to create thumbnails and full text extractions.

In conclusion, Hyrax and Islandora, have a more modular but more complex architecture. Compared to DSpace, both offer a better quality of modification of derived content directly by authors and have an International Image Interoperability Framework (IIIF) [8] server for dynamic image management such as Cantaloupe[9] for Islandora and ImageMagick[10] for Hyrax.

[2]Apache Solr: open-source java search platform

[3]FITS : File Information Tool Set (FITS) identifies, validates and extracts technical metadata for a wide range of file formats.

[4]Apache ActiveMQ :open source Java-based message broker

[5]Redis (Remote Dictionary Server): in-memory key-value store, used as cache and message broker

[6]Apache Karaf : ecosystem that allows to manage the run-time execution of multiple microservices according to different protocols: REST/API,...

[7]DSpace : content transformation processes

[8]IIIF : International Image Interoperability Framework: standardised method of describing and delivering images to enable client /server interoperability

[9]Cantaloupe : open-source dynamic image server producing on-the-fly derivatives of high-resolution source images

[10]ImageMagick : open-source software suite to manipulate images

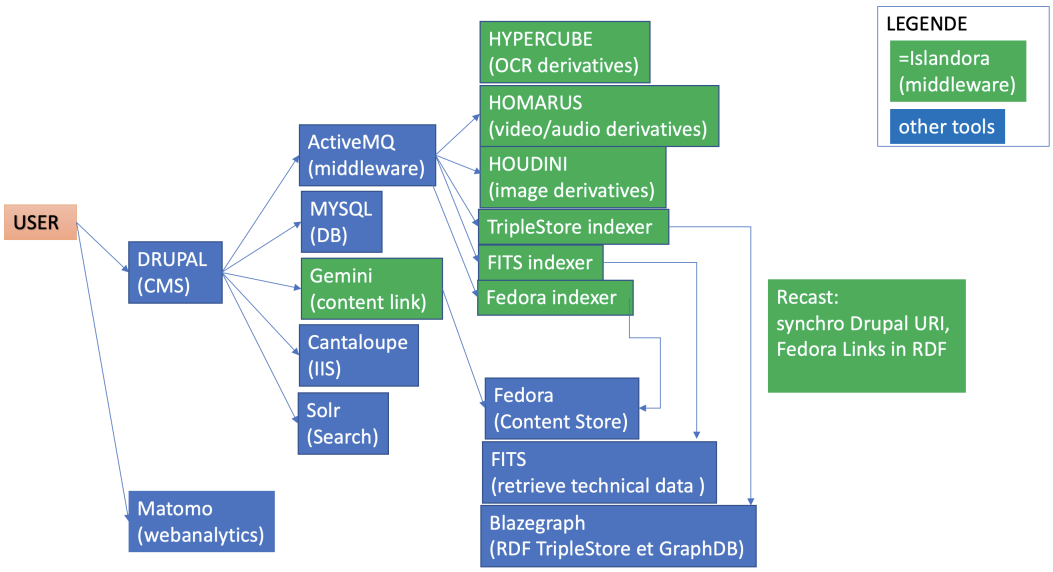


Figure 5.2: Islandora's user interface workflow

5.4 Internal storage

The difference between storage in DSpace and in Fedora has already been clarified: DSpace stores repository resources on file systems and persists metadata in a traditional relational database. On a regular basis, a cached copy is made on a triplestore according to the RDF model. In comparison, Fedora stores resources on internal or outsourced spaces in the cloud and works directly to persist the metadata in a triplestore. This approach allows direct visibility of linked data on Internet. For Islandora, a second triplestore is used to allow Fedora independence. For Hyrax, this independence is ensured by the Valkyrie solution [6]. Islandora and Hyrax have solutions to allow storage or retrieval of content from multiple external storage solutions.

5.5 Installation

In an on-premise implementation, products can be installed on expected OS (physical installation) or tenancies (virtual machine).

Much effort has been made to introduce installation procedures based on Docker or Ansible. These procedures facilitate installation, while allowing for better scalability, as the installation can verify its operation and adapt its services on demand. The classic Hyrax installation was not operational on MacOS Catalina. Fortunately,

the docker procedure, released in November 2020, allowed the prototype to be built. The Islandora installation via docker is called ISLE. It is useful considering the large number of modules that a manual installation requires. It should be noted that Islandora has a very decentralised organisation with 20 containers, whereas DSpace or Hyrax work with far fewer containers. Beta versions of DSpace v7 come also with a docker installation procedure.

5.6 Dependencies and maintenance

All three products implement, but at different scale level, a building block policy by incorporating the best-service offering external tools into their architecture. Doing so, external module dependencies must be managed. Each time a product evolves, the others have to reassess their interfaces. Supportive communities have to set up recommendations and provide some installation controls. This requires a dynamic community which needs to check and adapt quickly to dependencies changes. For instance, in version 7, DSpace uses a module called textitdependabot alert to detect vulnerabilities of third party modules.
This effort to maintain dependencies interoperability implies limiting the number of deployed versions and therefore customers are expected to follow the product releases strategy. Fedora is currently trying to convince its customer base to evolve to upper version and DSpace consider supporting only two released versions instead of three. The upgrade periodicity is a factor to consider in the product choice and is linked with the number of dependencies. This is not in favor of Islandora. However, to compensate this, Islandora has a module for administrators that lists the new recommended versions for all products of the architecture. With the use of Apache Karaf, it also has a dashboard of all modules with log tracking and alert reporting.

5.7 Security

The security of applications exposed to Internet is important even if the content of the repository is freely accessible. It is important that access controls are consistent via the user interface and via programmed access in REST-API. For example, it should not be possible to validate the submission of a resource without being a curator when submitting via REST-API commands. In addition, applications should not be subject to attacks that could compromise the integrity of the repository. To prevent this type of risk, it is necessary to conduct code robustness checks and external scans to identify vulnerabilities.

DSpace relies on multiple continuous and automatic checks. On one hand, it uses Github's Dependabot automated alerts to identify vulnerabilities in external modules. On the other hand, it has mandated an external company to continuously check DSpace's code to identify vulnerabilities or bugs while ensuring non-regression of deliveries. Moreover, a control of the tests coverage is carried out automatically during new deliveries. Particularly, the presence and validity of access control tests are verified. The other platforms have established continuous integration checks to verify non-regression and compliance with coding rules but does not indicate performing security tests automatically.

5.8 Conclusion

	DSPACE	Hyrax	Islandora
Flexibility	Customizable, developments in Angular	Customizable, developments in Ruby	Customizable with Drupal
Scalability	Monolithic in v6, Rest API in v7	Microservices	Microservices
Installation procedure	Docker, Ansible	Docker	Docker, Ansible
Dependencies: External building blocks	Angular, Solr	Ruby on rails, Solr, Fedora, FITS Redis, Blacklight	Drupal, Solr, Fedora, FITS, Cantaloupe, Matomo, Blazegraph, ActiveMQ
Maintainability effort	Small effort	integration effort	integration effort

Figure 5.3: Technical comparison summary

Figure 5.3 summarises the different aspects of the technical architecture. There are advantages and disadvantages for each of the solutions. If OER@UCLouvain's choice is to stay with DSpace, the centralised architecture allows a unified administration interface that is easier for curators. The ongoing architecture engineering is a major change that is necessary to allow DSpace to have an enhanced evolutionary capacity at the level of its competitors. If the choice is to migrate to Islandora to ensure a very customisable and scalable solution, it will require initial effort for the setup and run-time efforts to adapt continuously to all external modules dependencies.

Hyrax offers an intermediate solution with slightly fewer modules than Islandora. The negative point of Hyrax is the difficulty of finding documentation in the community.

Chapter 6

Metadata modelling

Metadata are information about resources. Their input quality is important for a successful search and use. The description of the content, the targeted user, or the intended use of the resource give valuable information that will ease their dissemination.
The objective of this chapter is to propose relevant metadata concepts based on a comparison of the metadata used by OER aggregators. Then, learning schemas will be assessed as standard support language to allow preservation, search and harvesting. Finally, some quality enhancements to facilitate their input and to ensure resources discoverability will be presented.

6.1 Educational metadata

The purpose of this section is to review the learning metadata to be filled in by authors and curators or to be filtered by the users of the search interface. The quality of the metadata is an indicator of the quality of the OER itself [27]. Comparison with OER aggregators such as MERLOT or OERCommons shows a good understanding and adequate use of traditional descriptive metadata such as *title*, *author*, *subject*, etc [11]. The focus is set specifically on educational metadata where some differences can be noted [25]. Figure 6.1 shows some differences in OER aggregators learning criteria proposed as search selection.
The learning activity which the resource supports is an important selection criteria called *material type* by OER aggregators. Some differences can be noted in their chosen vocabularies. Currently, OER@UCLouvain uses the metadata called *type*. Unfortunately, some authors input information from the semantic context of *Media Format* such as *video*, *text*. The former clarifies the learning activity to which the resource relate, the latter specifies the technical format corresponding to the resource content. To avoid ambiguities, it is recommended to name those metadata

	OER COMMONS OPEN EDUCATIONAL RESOURCES		MERLOT Multimedia Educational Resource for Learning and Online Teaching		OER@ UCLouvain	
Concepts	Name	Vocabulary	Name	Vocabulary	Name	Vocabulary
Define the type of learning activities	Material Type	Assessment, Full course, Interactive, Reading	Material Type	Assignment, Case Study, Drill and Practice, Online Course, Open Textbook, Tutorial, Quiz/Test, Training Material	Type	Video, Learning Object, Text, Book, Article, ...
Define the level of education	Educational Level	Primary, High School, College, Graduate, Professional, Adult Education	Primary Audience	Grade School, High School, College, Graduate, Professional	Typical age range	Primary, Secondary, Baccalaureat, Master
Define the intended user	Primary User	Student, Teacher, Parent	/	/	Audience	Student, Teacher, Both

Figure 6.1: Mapping of OERCommons, MERLOT and OER@UCLouvain metadata

material type and *media format* instead of *type* and *format*. Currently, this misuse of the metadata *material type* involves inappropriate referencing in the MERLOT repository. A large part of UCLouvain's OER are referenced as *Learning object repository* instead of more specific learning activities. Moreover, in the *subject* metadata, several values from the semantic field of *material type* are currently included. It would be beneficial to clarify to authors the usage of the *Material type* so they don't have to use the field *subject* instead.
Another difference can be observed in metadata concerning the *educational level* of the targeted student. The chosen vocabularies are culturally marked. For instance, UCLouvain uses *secondary* where MERLOT uses *high school*. In addition, one word is in French and others are in English in the current vocabulary.
A resource is intended for a user such as a student himself or herself or another teacher who can integrate the resource into his or her own course. When the resource can be used by either role, it is recommended to specify repeated metadata for each role, rather than using the term *both* which cannot be understood by OER aggregators.
One concept is not used by aggregators, but is defined currently on OER@UCLouvain. The *average learning time* can be useful when a user selects a resource and when he wants to use it.

6.2 Metadata schema

The next step consists in selecting a standard mutual agreed format called schema for relevant metadata concepts. A schema is a formal description of a domain specific metadata set with the rules for encoding information. It usually lists the metadata elements, the namespace of possible values and the rules to be used between elements. Standard schemas are used internally in the repository to qualify resources or used in the harvesting protocol to prepare the presentation of the resources to be disseminated [14]. This section presents the standardised schemas that are relevant for learning resources.

6.2.1 Dublin Core

Dublin Core is the cornerstone metadata standard schema. It is used for any digital content and has an extension to learning resources. Harvesting protocol requires its usage. There is a general consensus to use DC because of its simplicity. Although the metadata namings and concept descriptions are well standardised inside Dublin Core, the vocabulary is not standardised for most metadata. Some recommendations are done to choose vocabularies inside another schema called LRMI that will be presented later.

	DC Element Name	Definition
1.	Title	A name given to the resource.
2.	Creator	An entity primarily responsible for making the resource.
3.	Subject	The topic of the resource.
4.	Description	An account of the resource.
5.	Publisher	An entity responsible for making the resource available.
6.	Contributor	An entity responsible for making contributions to the resource.
7.	Date	A point or period of time associated with an event in the lifecycle of the resource.
8.	Type	The nature or genre of the resource.
9.	Format	The file format, physical medium, or dimensions of the resource.
10.	Identifier	An unambiguous reference to the resource within a given context.
11.	Source	A related resource from which the described resource is derived.
12.	Language	A language of the resource.
13.	Relation	A related resource.
14.	Coverage	The spatial or temporal topic of the resource, the spatial applicability of the resource, or the jurisdiction under which the resource is relevant.
15.	Rights	Information about rights held in and over the resource.

Figure 6.2: Dublin Core elements

The basic DC schema provides mutual comprehensive semantic concepts. It is built on a set of 15 repeatable and optional elements as shown by Figure 6.2.

The Dublin Core Metadata Initiative (DCMI) that supports DC, has proposed some education metadata to enhance a DC record and called the new schema DCTerms (DCT).

- The DCT:Audience is proposed to specify the intended user for the resource.
- The DCT:instructionalMethod specifies the interactivity type.
- The DCT:educationlevel represents the required or expected level of education of targeted students.
- The DCT:type is used to express the material type.

This learning extension is however limited. Not all learning metadata are added.

6.2.2 Learning Object Metadata

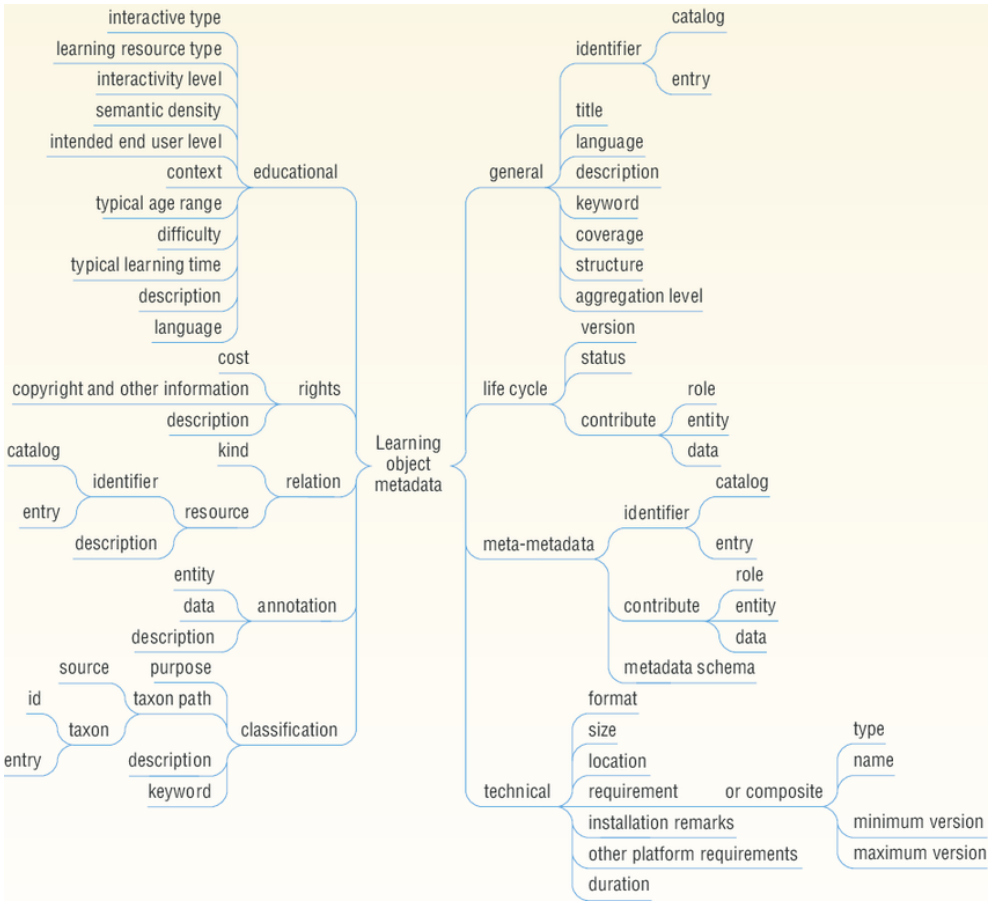


Figure 6.3: Learning Object Metadata

The Learning Object Metadata (LOM) standard is dedicated for learning resources. It was developed by the IEEE-Learning Technology Standards Committee. The

LOM has 9 categories containing 68 specific elements presented in a hierarchical tree-format in Figure 6.3 to facilitate the search for specific learning objects. The category called educational gives valuable information for the OERs: learning resource type, typical age range, intended end user role, interactivity type, typical learning time, etc.
LOM is widely used as metadata naming standard. As it doesn't specify the vocabulary to be used, various local applications are available. SupLOM.fr[1] or Normetic[2] are incompatible due to differences in vocabularies. The advantages of LOM over DCT are the availability of additional learning metadata (*context*, *difficulty*, ...) and a language typed structure for each metadata value.

6.2.3 Metadata for Learning Resources

To solve the various LOM applications incompatibility problem, Metadata for Learning Resources (MLR) has been proposed as an alternative standard. It is designed to be compatible with DC and LOM while allowing multilingualism and multiculturalism local application. This standard has started to be published in 2012 and continues to propose extension. The part *Migration from LOM to MLR* has been defined in 2017.

Recently, french researchers has been working on a specific MLR application profile for educational resources called NoDEfr-1 to replace supLOMfr. The authors working on this french normalisation [8] report the required addition of elements in order to define a consistent schema. They also mention that deployment of MLR requires a significant disambiguation effort of replacing string values with richer description structures. This transformation is required as MLR is built on RDF model. This model is presented in Appendix E. In a nutshell, RDF permits to represent concepts through a graph where objects and relations are well described in navigable structures. This effort of transformation is heavy and MLR usage in OER aggregators is not yet discussed. Therefore, it is not yet recommended to migrate from LOM to MLR.

6.2.4 Learning Resource Metadata Initiative

Search engines such as Google use Learning Resource Metadata Initiative (LRMI) as schema. To describe learning resources on the web, the LRMI project, supported by Creative Commons, started from the schema called Schema.org (RDF vocabulary used in a Web page to insert semantic content) and enriched it with terms specific

[1] supLOMfr: local application of LOM standard in French speaking graduation OER
[2] normetic: Canadian local application of LOM standard

Schema	Vocabulary
Educational Audience Role	Teacher, Student, Tutor, Parent, Team, ...
Educational Use	Lecture, Problem Solving, Problem-based, Project, Research, Review, Simulations, Homework, Model & Simulation, Brainstorming, Discovery Learning, Game, Presentation, ...
Interactivity Type	Active, Expositive, Mixed
LearningResourceType	Audio, Video, Quiz, Test, Presentation, Wiki, Lesson Plan, Lab Material, Model, Discussion, Lab, On-Line, Activity, ...

Figure 6.4: LRMI

to learning resources [7]. Therefore, although it is not used in harvesting protocols, LRMI specifies valuable vocabulary describing learning resources.Conceptual vocabularies such as *educational audience role*, *educational use*, *interactivity type* and *learning resource type* are presented on Figure 6.4.

LRMI is recommended for SEO[3]. MERLOT and OERCommons have established the mapping of their resources metadata with LRMI [3] . As recommended by DCMI, the usage of the LRMI vocabulary should be considered.

6.2.5 Schema recommendation

The recommendation is to use DC for traditional descriptive metadata and LOM for learning metadata associated with LRMI vocabularies. DC and LOM schemas can represent together adequately the concepts attached to OERs and are used in harvesting protocol. They can coexist as they complement each other. In addition, they are the choice of OER aggregators.

The *Learning resource type* vocabulary from LRMI can be used for the *Media format* concept and the *educational use* vocabulary can be used for *Material type*.

6.3 Metadata Quality Improvements

This section presents various controlled vocabulary techniques and lists how to use them to insure more coherent metadata input. The objective of this section is to improve the description of expected input by authors in order to obtain a better understanding of users of the search interface and an increased adequacy with OER aggregators. As a future extension, a resource model is explained using an announced feature of DSpace v7.0 including virtual metadata.

[3]SEO: search engine optimisation. Techniques to characterise the digital content of the website

6.3.1 Controlled vocabulary

To enhance the quality of the encoded metadata, controlled vocabulary[4] is recommended instead of free text metadata value [22]. Controlled vocabulary ensures consistency of the input data through the use of predefined and authorized terms. A variety of implementations exists. A closed terms list is the simplest. Taxonomy[5] or a thesaurus[6] provide richer selection with synonyms or narrower versus larger concepts. The control can be implemented locally in the repository or via an external controlling authority. In that case, it can be referred to this authority to select the possible metadata values. For instance, the Library of Congress (USA)[7] is a control authority[8] for languages or for subjects metadata values.

Material type, *Media format*, *Primary User* metadata are candidate for terms list controlled vocabulary.
In addition, the metadata *education level* would benefit from a taxonomy solution in order to set equivalence of level between different educational system. Indeed, the *education level* is geographically context dependant as high school refers to English speaking education and secondary school relates more to our local system. An extended vocabulary with synonyms or equivalencies would be an improvement for the search interface. Moreover, a thesaurus for the subject metadata would give wider selection when used to extend the selected semantic of the search criteria. For instance, when a user specifies *Newton laws*, it is possible to verify the selection with a larger concept as *Mechanics*. The solution of control authority for *subject* and *language* metadata with the Library of Congress, need to be evaluated.

[4]Controlled vocabulary : use of predefined and authorized terms. The control can be based on a simple term list or a complex machine-readable ontology.

[5]Taxonomies : is a hierarchical classification system with parent/child or broader/narrower relationships between items. It also allows for synonymy relationships.

[6]Thesaurus : is a kind of dictionary that lists all concepts in a specific domain and labels them with a preferred identifier. They contain variants and broader and narrower terms but also related terms.

[7]US library of Congress: `https://www.loc.gov`

[8]Authority control : is a central authority control that organizes an open controlled vocabulary service. This facilitates the selection of the unique term.

As explained, duplicates names for the same representation should be avoided. The difficulty with free-text metadata formatted as a string is that the control must be done manually by the curators. A Professor may be referenced with two different name spellings as in Figure 6.5. Therefore, it is difficult to find all his OER production in one set. It is therefore proposed to define an author as a unique entity object. Instead of a metadata string value, a relation with an existing person entity inside the repository will bring no problem of spelling. But, this requires a preliminary definition of authors and an additional on-demand process of definition for external contributors.

Figure 6.5: Different spellings for one author name

6.3.2 Configurable Entities and virtual metadata

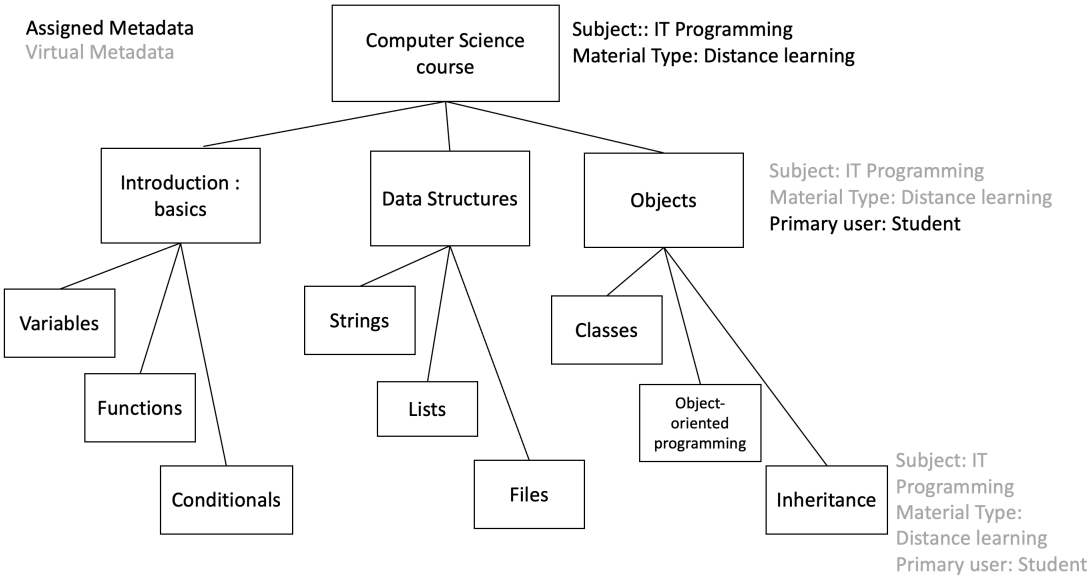


Figure 6.6: Entity relation applied to a CS course

Until now, OER have been considered individually or classified in a collection. It should be interesting to set relations between OERs. Those links could map a course structure as presented in Figure 6.6. Therefore, resources could gain virtual metadata from the top description level.
To organise such a graph of oriented relations between resources, a solution called configurable entities is announced in version 7 of DSpace [10]. DSpace has an

optional extension defining different types of repository items. They are called entities. Relations can be set between them. As an example, DSpace has built as a prototype an entities-relations model describing a journal composed of monthly publications. Each one contains several articles. The presentation of such compound objects can be customised to show the different hierarchy relations in one screen. Furthermore, the metadata definition of an entity type can use its related entity metadata. This avoids metadata redundancy and ease curation tasks. To do so, a virtual metadata concept has been introduced in the model to allow related entities to derive metadata. According to the documentation of DSpace, this requires some customisation to set the entities definition but most treatments are reusing core DSpace treatments.

6.4 Conclusion

Concept	Namespace	Control/Facility	Vocabulary example
Authors	dc:author	Controlled list	All educational community members of UCLouvain
Subject	dc:subject	Controlled list	Library of Congress Subject Headings
Material Type	dc:type	Controlled list based on LRMI	Educational Use vocabulary: Assessment, Lecture, Drill & Practice, Brainstorming, Comparing, Cooperative Learning, …
Primary User	lom:intended EndUserRole	Controlled list	Student, teacher, self-learner
Educational Level	lom:educational. TypicalAgeRange	Taxonomy of education level equivalence	Primary school, secondary school, bachelor, master
Typical Learning Time	lom:educational. TypicalLearning Time	Free text	Class-period, Quarter, Year
Media Format	dc:format	Controlled list	Video, Downloadable docs, Text,...

Figure 6.7: Enhancements of OER@UCLouvain metadata

To enhance the current OER@UCLouvain platform, the metadata concepts supporting the description and the search of a learning resource have been compared with their usage by OER aggregators. The available learning metadata schemas have been presented in order to define how to use them. The selected schemas are DC and LOM as they permit to express all required concepts and are useful for harvesting interoperability. Some LRMI vocabularies have been selected to extend the description of metadata. The future extension of virtual metadata proposed in DSpace has been discussed considering OERs in a more linked structure such as a course path. Figure 6.7 summarizes main relevant metadata concepts metadata with their proposed controlled vocabulary enhancements.

Chapter 7

Harvesting

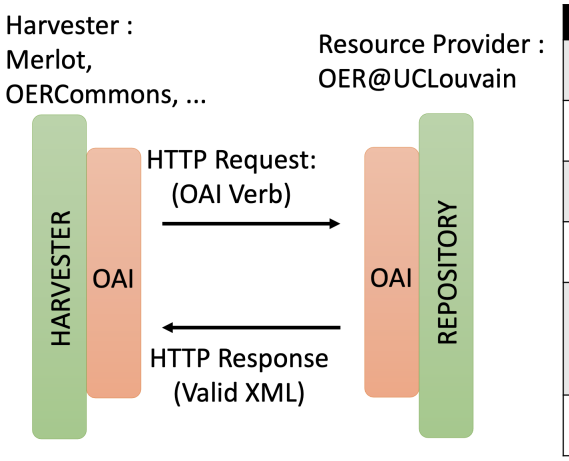
Harvesting allows resources to be disseminated and collected. In the case of OER@UCLouvain, two directions are considered. The first is export, which allows UCLouvain's learning resources to be propagated to OER aggregators sites that give OER greater visibility, and thus relay the university's reputation. The second is the exchange of resources between the OER repository and Moodle, a learning management system (LMS), in order to facilitate and automate exchanges as Moodle contains the majority of the teaching resources used at UCLouvain. This work of importing into OER@UCLouvain would be an improvement that would allow the gathering of course content.

7.1 Exporting to other OER platforms

UCLouvain open educational resources are intended to be freely and widely accessible in order to be shared. Their dissemination in global learning resources sites such as OERCommons or MERLOT is key. Currently, the transfer is manually scheduled. But, automatisation is possible, UCLouvain could define in the repository platform, a provider store that would contain the metadata of the resources in an exportable XML structure. This way, harvesting sites can use OAI-PMH[1][17], the most widely used harvesting protocol to gather those formatted resources on a regular basis. After gathering, harvesters will qualify and provide on their own sites the resources access links. This dissemination of access links avoids deduplication while favoring content sharing.

[1]OAI-PMH : Open Archives Initiative Protocol for Metadata Harvesting

7.1.1 OAI-PMH



HTTP VERBS	DESCRIPTION
Identify	Information about the OAI Store
ListMetadataFormats	Queries the list of available metadata form
ListSets	Requests the list of sets available in the store
ListIdentifiers	Retrieves the list of available identifier
ListRecords	Returns a list of records corresponding to the different parameters (dates, set) requested
GetRecord	Retrieve a given record (metadata)

Figure 7.1: OAI-PMH protocol

As a provider, OER@UCLouvain has to gather its published resources. Each resource is presented in the form of one record containing a formatted structure of associations of a metadata label and its specific value. There are several records for one resource if there are several metadata schema. A record is an XML-encoded byte stream that includes a header, a manifest and an optional and repeatable field used for rights statements. The header contains a unique identifier of the resource itself with useful information for selective harvesting such as the date of creation of the resource or its membership in a set of records. The grouping of records in sets is mandatory for large repositories. This grouping corresponds usually to collections. The manifest contains metadata in the format of the Dublin Core or LOM schema as explained in the previous chapter.

On their side, harvesters make OAI-PMH service requests to harvest these metadata via six HTTP verbs or services, as shown in Figure 7.1. With the first identify request, harvesters will negotiate the provider exchange capabilities. They can then use 4 verbs either to get list of supported metadata formats, list of sets or to request the full list of records or resource identifiers from the provider. In addition, a get request allows them to retrieve a specific record. Harvesters can limit their harvesting requests to portions of the metadata available in a repository by using two types of criteria that can be combined in an OAI-PMH query: datestamps and set membership. The flow control is realized via a resumption token that contains context information on the flow, to restart the transfer. A provider store will limit the transmitted volume through sending this resumption token. The harvester has to wait for a period of time before sending a new request with the resumption token.

The main advantage of OAI-PMH is its wide use and its simplicity. By building such an OAI store, UCLouvain would open the access to its OER.

However, OAI-PMH protocol has some attention points as listed by COAR[2] in its next-generation repository study[23].
Firstly, the difficulty lies in the persistence of resource identifiers. The identifier that the harvester has obtained should ideally not change. However, this is not always the case despite the FAIR[3] principles [35] . So, harvesters are forced to regularly check that resources are still accessible. To avoid the problem of resource deletion, UCLouvain has established the policy that resources must be preserved and therefore never deleted. However, if the repository moves to another platform, the identifier of a resource would change. The solution is to use a DOI[4] that allows the location of resources to be changed while maintaining a stable single access via an authority that provides the link.
Secondly, OAI-PMH harvesting allows the export of metadata but not of the resource as such and therefore the association of metadata with the resource is not guaranteed. Harvesters needs to regularly verify the relation between metadata and their referenced resource.
In addition, harvesting frequency is done on a regular basis but not continuous. Ideally, it would be interesting if synchronization could be done when change occurs and would require only modified resources exchange. Moreover, it is worth knowing that the exchange protocol is penalised for extremely large repositories with a sequential restart protocol when error occurs.
In the early days of OAI-PMH, search engines like Google also supported this protocol to reference all Internet resources. But soon, they preferred other referencing solutions forcing those who want to optimise the presence of their resources on the net to build other support to explain their contents. By default, robots continuously analyse the content of sites and deduce from what they find the meaning of the resources. To optimise referencing with a better description of the content, site managers prepare content description files for the robots. This is done with sitemap[5] file that is written in schema.org vocabulary. Therefore, a parallel reference format listing could be organised by UCLouvain to present in a sitemap its resources to robots that scan the site. What COAR suggests for next generation repository, is that this parallel effort of gathering resources metadata for OAI-PMH and sitemap would be mutualised.
In terms of new upcoming features, COAR expects protocol evolution because har-

[2]COAR: Confederation of Open Access Repositories
[3]FAIR: Findable, Accessible, Interoperable and Reusable
[4]DOI: Digital Object Identifier
[5]Sitemap : a XML-file present on website to present its resources to search engines robots

vesting on the next generation repositories will, in the future, include notifications, annotations, comments, peer-reviews, usage statistics on resources.

7.1.2 ResourceSync

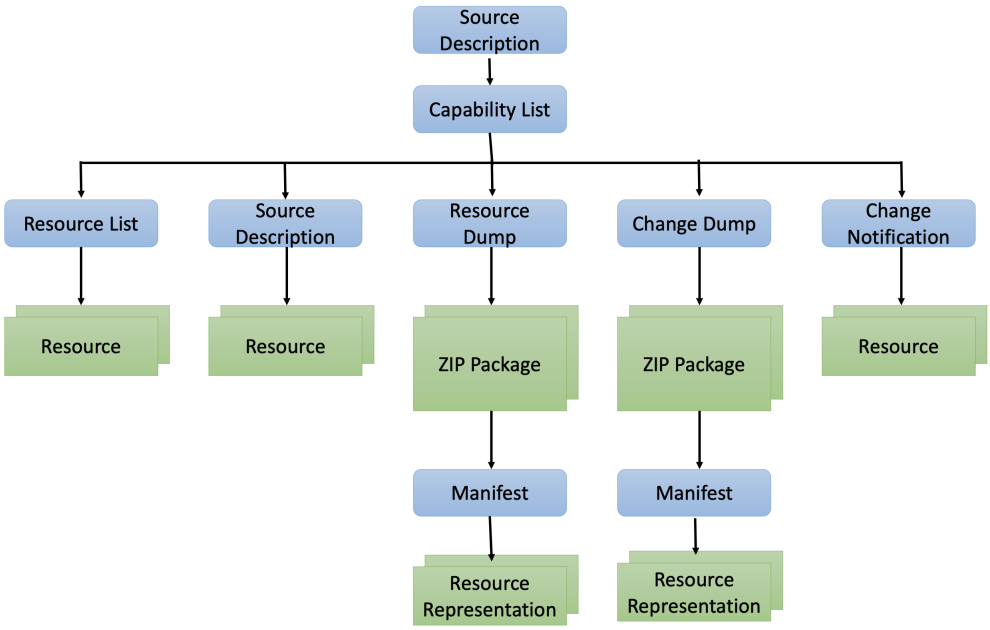


Figure 7.2: ResourceSync: capabilities

To improve OAI-PMH, COAR recommends the evolution towards other solutions like ResourceSync [4]. This new harvesting protocol has been strongly imposed to the different US libraries by the National Library in the United States. Currently, this protocol is implemented by Hyrax and is in the DSpace 7.2 features list.

The main advantage is that ResourceSync harvesters can gather the full-scope resource list or synchronize only modified resources to allow a faster and more reactive dissemination. In addition, the protocol allows to avoid one by one resource requests as in OAI-PMH, by using already prepared packages (called dump) that can be retrieved as a whole.
As defined in the protocol, a negotiation allows the harvester to discover the choices implemented by the provider. The possible provider capabilities as drawn in Figure 7.2, are to prepare the list of resources and the dumps of resources or the list of modifications and the dumps of modified resources. In the case of resources are coming from other sources, it is also possible for the provider to inform the harvester and avoid multiple harvesting of the same resource. As for OAI-PMH, resource

sets are also available. Furthermore, to share publishing efforts, ResourceSync is based on the XML sitemap format. Therefore, it combines the effort of referencing metadata and content for both harvesters and Web search engines.

Despite these improvements [16], OERCommons continues to use OAI-PMH and do not plan to change the protocol at this time.

7.2 Importing from courseware

Currently, educational resources are developed in the university's learning management system, Moodle. To be preserved and disseminated, resources have to be transferred in the OER@UCLouvain platform. It would be an appreciated facility for the teaching community to use an automatic link between the two platforms.

This part is based on the documentation of the Moodle platform and selected repository platforms. However, it is purely theoretical and no practical validation was done.

Firstly, the export capabilities in Moodle itself have been analysed. Moodle works with educational resources in its own format and with specific metadata. It is a proprietary structure for course descriptions and Moodle does not have a native export solution in a standard format. It can however import courses built with the Sharable Content Object Reference Model (SCORM) standard, but can't export in SCORM. The only interoperability solution offered by Moodle is via the Learning Tools Interoperability[6] protocol [13] which allows the publication of courses that are then available via other LMS[7] having the corresponding interface. But, this interface is not implemented on DSpace nor on Hyrax or Islandora.

Therefore, as no native export mechanisms are available, the Moodle plugin list has been analysed. The required plugin must be able to address two issues: the construction of a set of metadata on the resource in a XML format that can be understood by both platforms and a network transfer solution.

There are two transfer solutions referenced with DSpace; either one via OAI-PMH or one via SWORD[8]. OAI-PMH can be used as import solution where Moodle organises a store and the repository organises the harvesting. SWORD is

[6]Learning tools interoperability (LTI): allows, for example, Moodle to cooperate with another LMS to provide courseware.

[7]LMS: learning management system

[8]SWORD: Simple Web-service Offering Repository Deposit

a one by one resource transfer solution that can be used between Moodle and the repository. Although not documented in Moodle, as Fedora is also compatible with these protocols, those solutions could also fit for Islandora or Hyrax.

7.2.1 OAI-PMH and OAI-ORE

This export requires two OAI interfaces : one provider store on Moodle and one harvester server on the repository side.

On the latter side, DSpace uses the OAI harvester interface. This interface, in addition to OAI-PMH, uses another protocol called OAI-ORE[9] that describes aggregation standards of web resources that are exchanged [15]. An aggregation is a combination of multiple media types files such as texts, images, data, and videos.

On the Moodle site, there are two possible plugins that allow to export in OAI-PMH: one takes the resources of the whole site and the other exports only those indicated as exportable by their authors.

The most recent Moodle plugin is developed in 2018 by Humboldt-Universität zu Berlin's. It has been tested on Moodle 3.4. According to the documentation, the Moodle course owner defines which resources can be uploaded to the OER repository, they fill in the metadata for the resources. A XML package is then uploaded in the OAI store of Moodle so that the resources can be harvested by DSpace through its OAI-ORE/OAI-PMH Harvester module.
Finally, the harvesting interface deposit resources inside the repository, but the curator still need to review metadata and adds the resources to the right collection.

Another solution using OAI-PMH, has been developed by some french representative. This solution has been tested on Moodle 2.7 to Moodle 3.4. The difference with the University of Berlin modules is that this solution provides a site-wide sharing repository engine including metadata support.

7.2.2 SWORD

The other available interconnection plugin on Moodle is based on the protocol SWORD[26], that stands for Simple Web-service Offering Repository Deposit. It allows a solution to deposit content one by one from Moodle to the repository.
It is used by the Up2University [2] platform, an European Commission initiative,

[9]OAI-ORE : Open Archives Initiative Object Reuse and Exchange

which has recently built an ecosystem based on Moodle as LMS and DSpace as OER repository.

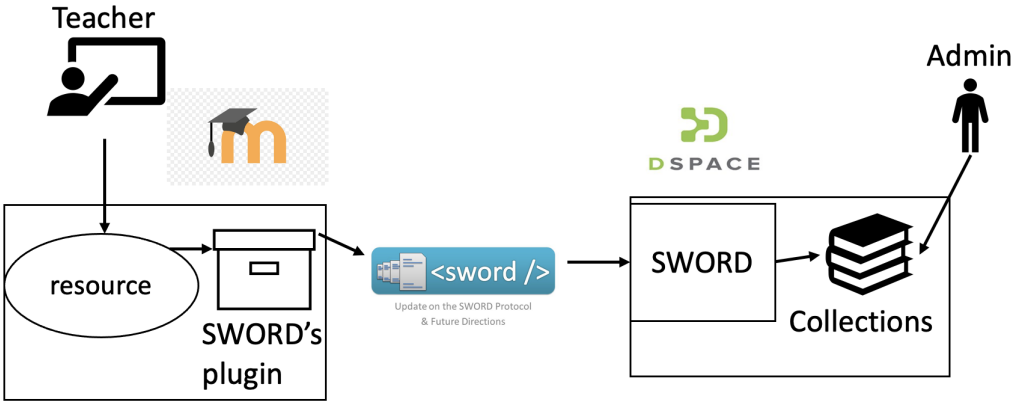


Figure 7.3: SWORD Plugin

As shown in Figure 7.3, with this plugin, the course owner chooses in Moodle which educational resource can be sent to the repository. He fills in the necessary metadata, and then validates his submission. The resource is uploaded to the repository using the SWORD module. Finally, the curator checks the metadata and adds the resource to the right collection in the repository. On the repository side, DSpace provides SWORD interface in its version 6 and intends to do so in its version 7.2. Fedora has also the SWORD interface. For OER@UCLouvain, this would provide a basic solution to store and preserve OER.

7.3 Conclusion

The implementation of an OAI-PMH store and its automatic and regular loading from the repository content is a solution for OER@UCLouvain to replace manual transfers to OER harvesters. The creation of an OAI store is native in the DSpace solution. Regular scheduling must be added to enable the capture of newly published resources and their associated metadata. Access is directly in place for OER aggregators. The deployment of ResourceSync is not yet to be considered.

The Moodle to OER Repository linking solution is feasible. Such a solution should be validated. However, it may increase the curation load. Such an exchange is dependant on both platforms, Moodle and the repository. Therefore, it must be maintained accordingly.

Chapter 8

Conclusion

This study has established several ways to improve the OER@UCLouvain platform in order to facilitate the production and boost the dissemination of educational resources from the University. First, the search interface is proposed to be uplifted to integrate educational characteristics as selection criteria. This will give a better experience to visiting users. They will be able to express their needs according to material type or education level. Secondly, the exchange process with harvesters for the dissemination will benefit from automatisation. Then, several metadata management facilities can be made available to resource creators in order to improve their input quality and doing so, improve the discoverability of the UCLouvain educational resources.

Based on the needs and proposed improvements, a list of criteria was established in order to compare what DSpace, the current hosting platform, proposes as an evolution compared with its competitors Hyrax and Islandora associated with Fedora. The functional, technical and support comparison was based on versions proposed from November 2020 to March 2021. It was determined that the DSpace beta 4 version was not sufficient to fulfill the requirements. Therefore, it has been decided to consider the theoretical DSpace v7.2 (delivery not yet planned) with documented future features.

The result of the functional comparison is to the advantage of Hyrax and DSpace v7.2. In those two platforms, the mandatory features are present. Especially, the quality of their interface in terms of user experience brings the expected improvements. The facilities in terms of metadata management are also available and can provide useful assistance to the educational community. DSpace has an advantage over Hyrax concerning the integration of the management interface.

The technical comparison that has been conducted to analyse how the platforms

were built, demonstrate a current more modular architecture for Hyrax and Islandora with Fedora. They allow the integration of multiple products and ensure an easier evolution as interfaces are well structured. However, DSpace is currently adapting to these new modular techniques and well structured interfaces. Some choices are identical for the three platforms, as for example, the use of Solr as the indexing and searching engine. On the contrary of DSpace and Hyrax that are built as turn-key solutions, Islandora requires a lot of customisation. There is a lot of work to set up the application whereas the two other products bring directly coherent and efficient solutions. Islandora customisation level is perceived as a disadvantage and has penalised it in our tests due to lacking directly installed features. However, it is fair to notice its remarkable flexibility of adaptation with easy click buttons. The main reason for its non-selection is its strong dependency on Drupal which requires a sustained update rate and the decoupling of the OERs into two different entities to be coordinated between the Drupal (node) and Fedora (repository element) architecture.

In terms of support and documentation, DSpace and Islandora are very well provided while Hyrax has still progress to do. Its documentation is not complete and mixes the different versions. As an example, the version 3 release does not have a documentation space outside its project space.

As a comparison conclusion, since Hyrax and DSpace 7.2 are equivalent in terms of functionalities, since DSpace moves in the direction of a more modular architecture and since its documentation and community are more supportive, I would advise to renew the choice of DSpace for OER@UCLouvain but I recommend to wait for DSpace 7.2 at least.

While waiting for the release of DSpace v7, there are several actions to do on DSpace v6. Indeed, the loading of the OAI-PMH store for harvesters can be automated and the effort to disseminate resources can be reduced. Secondly, resource thumbnails that give a more pleasant presentation, can be generated and proposed to authors with the assistance of a curator. In addition, the implementation of learning metadata criteria in the search process requires the creation of new indexes in Solr. That operation is also already possible in DSpace v6. Those actions will already propose a more user-friendly interface that can be uplifted when the version 7.2 will be released.

This work allowed me first of all to put into practice the project methods I learned during my studies. It was an interesting work of structuring, discussion and choice to establish the requirements and the improvements of the OER@UCLouvain plat-

form in order to boost the production and the dissemination effort. The selection of the solutions to be compared required a lot of reading and research. Setting up the prototypes has been instructive. I learned a lot from multiple installations and the analysis to verify their functioning and interaction. Working with beta software with incomplete documentation was not always easy. The Islandora prototype required to define and customise the basic elements of a site. This step of creation, even on a small scale, was a good learning experience for me. The test plan based on the requirements allowed me to establish a comparison framework. The key elements of differentiation and synthesis were established to give an informed opinion on what could be an adequate solution for the university. At the same time, I was able to deepen my understanding of harvesting and metadata management mechanisms.

This study has been instructive and exciting. I hope to have established sound advice for the future of OER@UCLouvain because I am convinced that learning resources are treasures to be disseminated and promoted in the world.

Appendix A

Requirements specification

This appendix summarizes all the functionalities required for the proper functioning of an OER repository for UCLouvain.

	Category	Feature	Description	required	optional
Resources Searching	Long term access	Preservation	OER Content is stored with its metadata to be accessible on long-term	X	
		Fixity	OER Content integrity is guaranteed. Alteration is detected and solved	X	
		Unique identifier	OER content is accessible on long-term through a URI link from anywhere	X	
		Versioning	Ability to upload a new version of the OER and keep previous	X	
	Digital Content	Content types	All content types are supported to be uploaded	X	
Resources Storage	Repository Structure	2-level presentation	Capability to structure the repository in layers: Communities-collection => A collection is a subset of the community	X	
	Search and filtering	Full-text search	Ability to search on the text of a file	X	
		Faceted search	Capability to add specific, relevant filtering options to your results page using easy user-selection	X	
		Search on LOM	Specific learning criteria selection	X	
	Repository presentation	Thumbnail	Thumbnail must be automatically generated	X	
		Full Scope OER presentation	See resources with their associated metadata in a non-technical way	X	
		Multilingual capabilities	English and French	X	
	OER metadata standards	LOM	Define the public (education level), the learning time, ...	X	
		Dublin Core	Define the author, the subject, keywords, ...	X	
		Creative Commons	Define which rights the public have on the OER	X	
Harvesting	Export OER	OAI-PMH	Harvesting protocol	X	
		ResourceSync	Harvesting protocol (not used by OER harvesters at the moment)		X
	Import OER from Moodle	SWORD	A plugin exists to interconnect repository with Moodle and use SWORD. So, the repository needs at least SWORD protocol	X	
Resources management	Submission	Workflow: 1-Step	The submission is directly accepted by the repository		X
		2-Step	Curator reviews the metadata	X	
		3-Step	Submission –> review of metadata – > final acceptance		X
		Multiple Files in one submission	Several files relating to the same OER, for example the tutor book and the student book	X	

Resources management		Metadata: LOM DC	Defining the LOM in the submission form Defining the DC in the submission form	X X	
		Controlled vocabulary	Ability to add a list of authorised terms on metadata	X	
		Inference of metadata	Inferring metadata on technical aspects (size of the file, type of the file)	X	
		Proxy Deposit	Ability to upload an OER for another person (and to transfer the OER to this person)		X
		Upload from external storage	Ability to upload files that are stored in an external storage such as Dropbox, OneDrive, etc		X
		Optional and required metadata	Ability to define which metadata are required and optional in the submission form (for the admin). A check is made at submission.	X	
	Management of my OER	Version management: -Modification metadata	Ability for an OER owner to change the metadata value of his OER	X	
		-Modification Files	Ability to insert, update the files, select thumbnail image	X	
		-Modification collection, community	Ability to add the OER in a collection, or to change in which collection the OER is stored	X	
		View OER usage statistics	Ability for an OER owner to consult the statistics of his OER	X	
	Notifications	By email When Connected	Ability to be informed if a new OER is uploaded on a selected collection, either by mail or when connected on the repository		X X
Repository management	Curation	Notifications of new OER	Event notification during the submission process	X	
		Automatic control of content	Processes for checking content	X	
		Metadata control	Processes to enhance the quality of metadata	X	
	Customisation	Community	The second level classification	X	
		Collection	The first level classification	X	
		Metadata schema	Ability to add new schemas	X	
		Metadata indexing selection	Ability to search on new metadata (such as LOM)	X	
		Optional/Required Metadata	Ability to define which metadata are required and optional	X	
	Administration	Content import/export	Ability to migrate content	X	

<table>
<tr><td rowspan="3">Repository management</td><td></td><td>Performance dashboard</td><td>Ability to see repository and infrastructure event volumetry, alerting control</td><td>X</td><td></td></tr>
<tr><td>Access management</td><td>Access control: Admin, Curator, Educational Community, Public</td><td>Define the rights of all members
<table><tr><td></td><td>Search</td><td>Upload</td><td>Metadata review</td><td>All</td></tr><tr><td>Public</td><td>X</td><td></td><td></td><td></td></tr><tr><td>Educational community</td><td>X</td><td>X</td><td></td><td></td></tr><tr><td>Curator</td><td>X</td><td>X</td><td>X</td><td></td></tr><tr><td>Admin</td><td>X</td><td>X</td><td>X</td><td>X</td></tr></table></td><td>X</td><td></td></tr>
<tr><td></td><td>Authentication: Shibboleth</td><td>user’s connection of UCLouvain, member of educational community</td><td>X</td><td></td></tr>
</table>

Appendix B

Tests

This appendix includes the tests applied to each product: DSpace 7 beta 4, Hyrax and Islandora. A table for DSpace 7.2 shows the features that will theoretically be present in that version.

DSpace beta 4

	Category	Feature	Comments	State
Resources Storage	Long term access	Preservation	Association of content and metadata to insure continuous access	1
		Fixity	There is a checksum control	1
		Unique identifier	Handle and DOI can be generated	1
		Versioning	Not yet available	0
	Digital Content	Content types:		
		PowerPoint	Upload OK – download OK	1
		Video mp4	Upload OK – download OK	1
		Excel	Upload OK – download OK	1
		Pdf	Upload OK – download OK	1
		Word	Upload OK – download OK	1
		Html, xml	Upload OK – download OK	1
		Zip	Batch upload NOK	0
		URL link	Not yet available	0
		raw file	Upload OK – download OK	1
Resources Searching	Repository Structure	2-level presentation	Community and collection structure done by administrator	1
	Search and filtering	Full-text search	Available on search interface, mediafilter technic	1
		Faceted search	By default on DC, customisation possible (Solr)	1
		Search on LOM	Customisation: add index on Solr	1
	Repository presentation	Thumbnail	Not yet available	0
		Full Scope OER Presentation	Customisation of item html form	1
		Multilingual capabilities	English and French available	1
	OER metadata standards	LOM	Add registry (schema), customisation of item metadata template	1
		Dublin Core	By default	1
		Creative Commons	License definition: initial customisation by administrator	1
Harvesting	Export OER	OAI-PMH	The OAI server is set-up by default - Planned synchronisation required	1
		ResourceSync	Not yet available	0
	Import OER from Moodle	SWORD	Not yet available	0

Resources management	Submission	Workflow: 1-Step	Not yet available	0
		2-Step	Not yet available (default in beta 5)	0
		3-Step	The workflow by default in beta 4	1
		Multiple Files in one submission	Manual submission upload OK	1
		Metadata: LOM	Initial customisation by administrator	1
		DC	By default	1
		Controlled vocabulary	Customisation on DSpace.cfg	1
		Inference of metadata	Not available	0
		Proxy Deposit	Not available	0
		Upload from external storage	Not available	0
		Optional and required metadata	Customisation item metadata template	1
	Management of my OER	Version management:		
		Modification metadata	New/update/delete metadata	1
		Modification Files	Can modify the files (don't generate a new version)	1
		Modification collection, community	Possible to change in which collection the resources are stored	1
		View OER usage statistics	Not yet available	0
	Notifications	By email	Not yet available	0
		When Connected		1
Repository management	Curation	Notifications of new OER	Not yet available	0
		Automatic control of content	Not working (curation task mark as failed)	0
		Metadata control	Not yet available	0
	Customisation	Community	Created in admin interface, test of differentiated user access	1
		Collection	Created in admin interface	1
		Metadata schema	Possible to add new schemas (Registry)	1
		Metadata indexing selection	Customisation DSpace.cfg	1
		Optional/Required Metadata	Customisation item metadate template	1
	Administration	Content import/export	Available but not working (corrected in beta5)	0
		Performance dashboard	Not yet available	0
	Access management	Access control: Admin, Curator, Educational Community, Public	Possible to add even more role: we can define a different curator for each collection or community.	1
		Authentication: Shibboleth	Documented as possible (not tested)	1

Hyrax

	Category	Feature	Comments	State
Resources Storage	Long term access	Preservation	Association of content and metadata to insure continuous access	1
		Fixity	There is a checksum control	1
		Unique identifier	Ok, DOI available	1
		Versioning	Add new version and old versions are preserved	1
	Digital Content	Content types:		
		PowerPoint	Upload OK – download NOK	0
		Video mp4	Upload OK – download OK	1
		Excel	Upload OK – download OK	1
		Pdf	Upload OK – download OK	1
		Word	Upload OK – download OK	1
		Html, xml	Upload OK – download OK	1
		Zip	Upload OK – download OK	1
		URL link	Upload OK – download OK	1
		raw file	Upload OK – download OK	1
Resources Searching	Repository Structure	2-level presentation	Administrator defines administrative set that contains collection	1
	Search and filtering	Full-text search	Automatic (done via derivatives)	1
		Faceted search	By default on DC, index on LOM to be added (Solr)	1
		Search on LOM	add property Class GenericWork (code)	1
	Repository presentation	Thumbnail	By default (done via derivatives)	1
		Full Scope metadata OER Presentation	Customisation metadata edit form	1
		Multilingual capabilities	English and French available	1
	OER metadata standards	LOM	Can be added (Class GenericWork) (code)	1
		Dublin Core	By default	1
		Creative Commons	By default	1
Harvesting	Export OER	OAI-PMH	Blacklight OAI provider must be set up	1
		ResourceSync	Available according to documentation, not tested	1
	Import OER from Moodle	SWORD	Available according to documentation, but only V1, not tested	1

Resources management	Submission	Workflow: 1-Step	Collection customised, direct submission	1
		2-Step	Submission with curator validation	1
		3-Step	Not possible	0
		Multiple Files in one submission	Manual submission and batch upload	1
		Metadata: LOM	Customisation of Submission form	1
		DC	By default	1
		Controlled vocabulary	Class GenericWorkForm (code)	1
		Inference of metadata	Not available	0
		Proxy Deposit	Tested via two different users	1
		Upload from external storage	Tested on Google drive (via Hyrax Sandbox)	1
		Optional and required metadata	OK (Class GenericWorkForm (code)	1
	Management of my OER	Version management:		
		Modification metadata	New/update/delete metadata	1
		Modification Files	Can modify files, upload a new version	1
		Modification collection	Possible to change in which collection the resources are stored	1
		View OER usage statistics	Number of views and downloads	1
	Notifications	By email	The notifications process is working only if the user is connected to the repository	0
		When Connected		1
Repository management	Curation	Notifications of new OER	When connected	1
		Automatic Control of content	Not integrated in admin interface - run in command line	1
		Metadata Control	Automatic rules control - run in command line	1
	Customisation	Community	Administrative set created in admin interface, test of differentiated user access	1
		Collection	Created in admin interface	1
		Metadata schema	Class GenericWork (code)	1
		Metadata indexing selection	Modification in the code needed (Class GenericWork)	1
		Optional/Required Metadata	Modification in the code needed (Class GenericWorkForm)	1
	Administration	Content import/export	Admin interface (csv import, bulk import and export)	1
		Performance dashboard	System status	1
	Access management	Access control: Admin, Educational Community, Public	User acces control tested via interface	1
		Authentication: Shibboleth	Documented as possible (not tested)	1

DSpace 7.2

	Category	Feature	Comments	State
Resources Storage	Long term access	Preservation	Association of content and metadata to insure continuous access	1
		Fixity	There is a checksum control	1
		Unique identifier	DOI and handle can be generated	1
		Versioning	Available according to documentation	1
	Digital Content	Content types:		
		PowerPoint	Upload OK – download OK	1
		Video mp4	Upload OK – download OK	1
		Excel	Upload OK – download OK	1
		Pdf	Upload OK – download OK	1
		Word	Upload OK – download OK	1
		Html, xml	Upload OK – download OK	1
		Zip	Available according to documentation	1
		URL link	Available according to documentation	1
		raw file	Upload OK – download OK	1
Resources Searching	Repository Structure	2-level presentation	Community and collection structure done by administrator	1
	Search and filtering	Full-text search	Available on search interface, mediafilter technic	1
		Faceted search	By default on DC, customisation possible (Solr)	1
		Search on LOM	Customisation: add index on Solr	1
	Repository presentation	Thumbnail	Available according to documentation (mediafilter)	1
		Full Scope OER Presentation	Customisation of item html form	1
		Multilingual capabilities	English and French available	1
	OER metadata standards	LOM	Add registry (schema), customisation of item metadata template	1
		Dublin Core	By default	1
		Creative Commons	License definition: initial customisation by administrator	1
Harvesting	Export OER	OAI-PMH	Set-up by default	1
		ResourceSync	Available according to documentation	1
	Import OER from Moodle	SWORD	Available according to documentation (V2)	1

Resources management	Submission	Workflow: 1-Step	Available according to documentation	1
		2-Step	The workflow by default in beta 5	1
		3-Step	The workflow by default in beta 4	1
		Multiple Files in one submission	Manual submission upload OK	1
		Metadata: LOM	Initial customisation by administrator	1
		DC	By default	1
		Controlled vocabulary	Customisation on DSpace.cfg	1
		Inference of metadata	Not available	0
		Proxy Deposit	Not available	0
		Upload from external storage	Not available	0
		Optional and required metadata	Customisation item metadata template	1
	Management of my OER	Version management:		
		Modification metadata	New/update/delete metadata	1
		Modification Files	Can modify the files (don't generate a new version)	1
		Modification Collection	Possible to change in which collection the resources are stored	1
		View OER usage statistics	Available according to documentation	1
	Notifications	By email	Available according to documentation	1
		When Connected		1
Repository management	Curation	Notifications of new OER	Available according to documentation	1
		Automatic Control of content	Available according to documentation	1
		Metadata Control	Available according to documentation	1
	Customisation	Community	Created in admin interface, test of differentiated user access	1
		Collection	Created in admin interface	1
		Metadata schema	Possible to add new schemas (Registry)	1
		Metadata indexing selection	Customisation DSpace.cfg	1
		Optional/Required Metadata	Customisation item metadate template	1
	Administration	Content import/export	Available according to documentation	1
		Performance dashboard	Available according to documentation	1
	Access management	Access control: Admin, Curator, Educational Community, Public	Possible to add even more role: we can define a different curator for each collection or community.	1
		Authentication: Shibboleth	Documented as possible (not tested)	1

ISLANDORA 8

	Category	Feature	Comments	State
Resources Storage	Long term access	Preservation	Association content and metadata to insure continuous access	1
		Fixity	There is a checksum control	1
		Unique identifier	OK, DOI available	1
		Versioning	Versions management in Drupal (node) and in Fedora (media files)	1
	Digital Content	Content types:		
		PowerPoint	Upload OK – download OK	1
		Video mp4	Upload OK – download OK	1
		Excel	Upload OK – download OK	1
		Pdf	Upload OK – download OK	1
		Word	Upload OK – download OK	1
		Html, xml	Upload OK – download OK	1
		Zip	Module to be added	0
		URL link	Upload OK – download OK	1
		raw file	Upload OK – download OK	1
Resources Searching	Repository Structure	2-level presentation	Not available, one level collection only	0
	Search and filtering	Full-text search	Done via derivatives after customisation	1
		Faceted search	Done via customization of facets	1
		Search on LOM	Done via adding of LOM in content type template and adding of LOM in Solr indexes	1
	Repository presentation	Thumbnail	Done via derivatives after customisation	1
		Full Scope OER Presentation	Customisation of views	1
		Multilingual capabilities	English and French available	1
	OER metadata standards	LOM	Via adding of LOM in content type template	1
		Dublin Core	By default	1
		Creative Commons	Not tested, module to be added	0
Harvesting	Export OER	OAI-PMH	Customisation of OAI views, required scheduling of OAI server synchronisation	1
		ResourceSync	Not available	0
	Import OER from Moodle	SWORD	Available (V1) in Fedora	1

Resources management	Submission	Workflow: 1-Step	By default	1
		2-Step	Not tested, requires workflow module	0
		3-Step	Not available	0
		Multiple Files in one submission	Submission of file one by one only	0
		Metadata: LOM	Taxonomy customisation, content metadata template customisation	1
		DC	Taxonomy customisation, content metadata template customisation	1
		Controlled vocabulary	Term list, taxonomy, control authority	1
		Inference of metadata	Not available	0
		Proxy Deposit	Not available	0
		Upload from external storage	Module to be added	1
		Optional and required metadata	Tested via customisation of views	1
	Management of my OER	Version management:		
		Modification metadata	Possible (Modification of a node in Drupal)	1
		Modification Files	Possible (Modification in the Fedora repository)	1
		Modification Collection	Possible to add in multiple collection, remove from one, ...	1
		View OER usage statistics	Statistics in an external module	0
	Notifications	By email	Not available	0
		When Connected		1
Repository management	Curation	Notifications of new OER	Not tested, Requires workflow module	0
		Automatic Control of content	Automatic control of customised rules	1
		Metadata Control	Automatic control of customised rules	1
	Customisation	Community	Not available	0
		Collection	Tested, difficulty of controlling embedded views	1
		Metadata schema	Possible to add new one directly in the app	1
		Metadata indexing selection	Customisable in Solr	1
		Optional/Required Metadata	Customisable	1
	Administration	Content import/export	Not in admin interface	0
		Performance dashboard	System status, general log events (automatic upgrade proposition)	1
	Access management	Access control: Admin, Educational Community, Public	Different users access tested	1
		Authentication: Shibboleth	Documented as available, not tested	1

Appendix C

Islandora Set-up

After installing Islandora, Drupal, Fedora, it is necessary to make customizations. Those adaptations require an understanding of how Drupal works but do not require development expertise. The goal of this prototype was to realize an operational platform on which the tests could be realized.

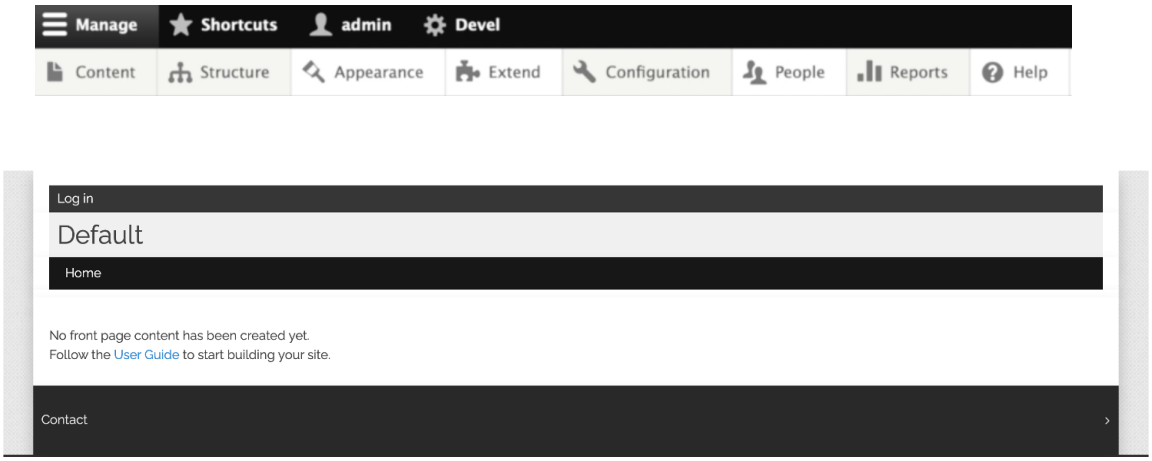


Figure C.1: Islandora first screen without customisation

C.1 General settings

Many customizations are easily done with a simple button. When multiple actions are needed for the administrator, Islandora gives design tips and at any time a dashboard presents the status of the services and report errors. Another dashboard allows to see the available upgrades for all elements of the framework.
Many Drupal concepts allow to design a responsive interface that adapts size of

images, language to the context of the user.
The site is structured in pages or views structured in blocks, which are positioned in different spaces of the page. There is a block menu, one sidebar, one for the content, ...
The theme was adapted and the OER@UCLouvain logo image was easily uploaded.

C.2 Collection structure

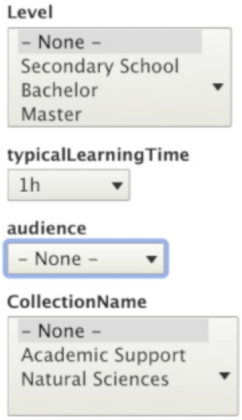
In order to replicate the collection structure seen in OER@UCLouvain, two collections have been created. A page has been created per collection and a reference to the page added in the home menu. The page was designed to present all content nodes that were member of that specific collection. In the repository, a content item node that was typed as collection has been created. The other possible content types were articles or basic page (static site information). Some OER content items were submitted and linked as member of the collection. However, when viewing the full content of the repository, the specific node of the collection itself also appeared. It was then necessary to add a specific field describing the content type in order to exclude the nodes representing collections top level. Those manipulations allowed to present content items inside one collection. But, the test of a two-level collections was not conclusive.

C.3 Content view and edit form

The submission of content item requires two steps: the creation of the node (a concept necessary for Drupal) and then the addition of a repository item to add content files in Fedora. This two step process is not intuitive. The work to hide this separation is important and not foreseen in the basic customization. Moreover, lots of metadata are to be filled in by default during submission . Fortunately, it is possible to customize the metadata list with a mandatory and optional attribute and to customise their presentation association. Some metadata can be set visible in the complete view of the metadata associated to a content and others are presented in a teaser when the content item is part of a browsable list. As there are two concepts: content (Drupal node) and media (content file repository in Fedora), there are also two configurable spaces of metadata. Fortunately, Islandora uses FITS which allows the ingestion of a content file to automatically deduce the technical metadata. Unfortunately, it was not possible to adapt the metadata that contains the creative commons access license. According to the documentation this is possible but it requires an additional plugin which is not part of the standard installation.

C.4 LOM metadata adding

An important criterion for improving the current operation of OER@UCLouvain is the possibility to search on educational metadata values (LOM). It was easy to add them to the metadata description associated with a content (node). In order to be able to use them in the search, it was necessary to find the access to the configuration of Solr which is the search engine and to specify the indexing to be carried out on these new metadata. It was also necessary to add them for the visualization of the contents during the browsing list.



C.5 OAI-PMH

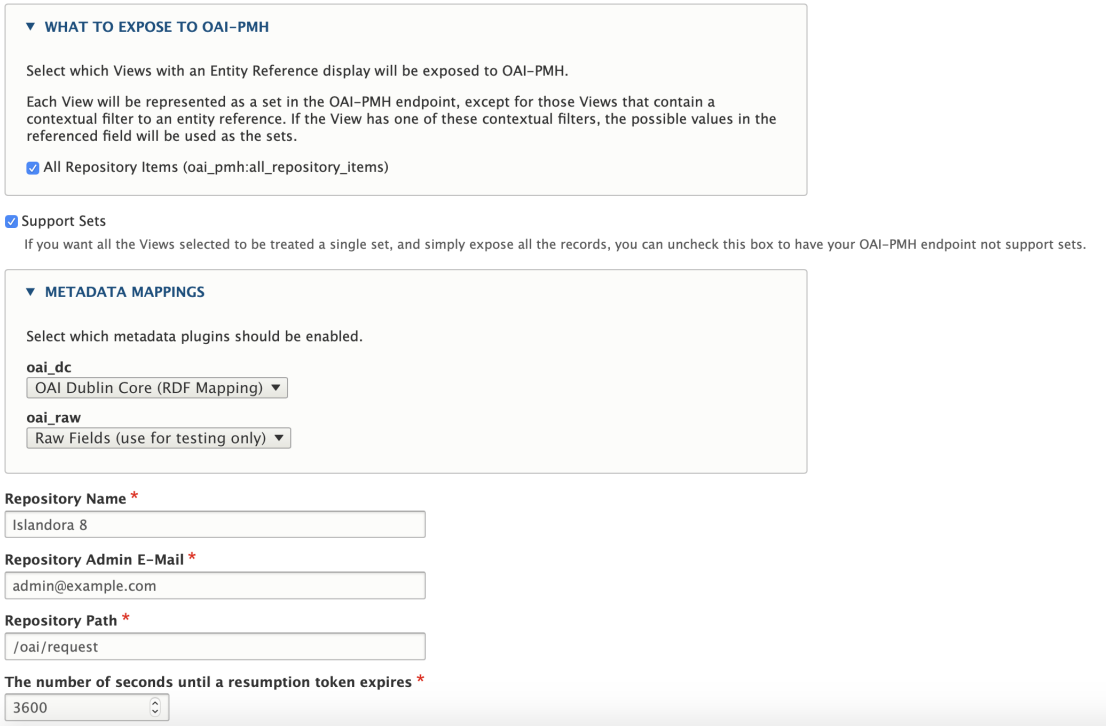
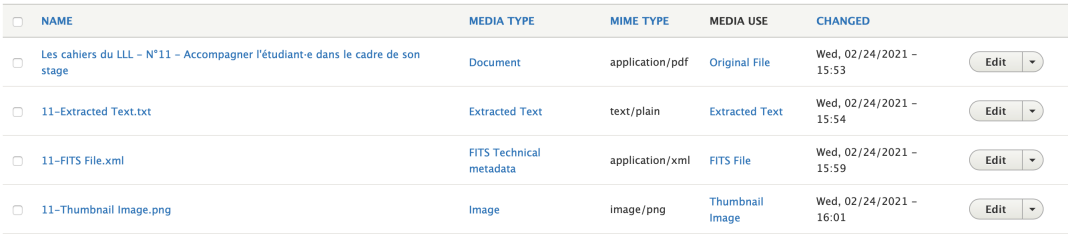


Figure C.2: Set-up of OAI-PMH in Islandora

Harvesting is an important requirement. That's why it was necessary to verify its operation. First of all, as standard, it is not allowed. You have to activate

the rest-api for it. It is associated with a view that allows to specify what can be harvested from the site and which metadata fields must be specified. This allowed to verify that the LOM metadata was also part of it.

C.6 Thumbnail and Full-text extract generation



NAME	MEDIA TYPE	MIME TYPE	MEDIA USE	CHANGED	
Les cahiers du LLL – N°11 – Accompagner l'étudiant-e dans le cadre de son stage	Document	application/pdf	Original File	Wed, 02/24/2021 – 15:53	Edit
11-Extracted Text.txt	Extracted Text	text/plain	Extracted Text	Wed, 02/24/2021 – 15:54	Edit
11-FITS File.xml	FITS Technical metadata	application/xml	FITS File	Wed, 02/24/2021 – 15:59	Edit
11-Thumbnail Image.png	Image	image/png	Thumbnail Image	Wed, 02/24/2021 – 16:01	Edit

Figure C.3: Extracted files from a pdf

In order to improve the rendering of the site by adding images, the test of the thumbnail autogeneration during the ingestion of content has been realized. The principle in Drupal is based on actions that are executed in response to events through contexts. It is therefore possible to specify to Islandora that it must generate a thumbnail when ingesting a media image or pdf among others and this is done automatically. These additive elements are linked in the media list of the content (node) and can be managed by the author.

C.7 Vocabulary - Taxonomy

Regarding metadata facilities, Islandora automatically offers controlled vocabulary solutions: default value, list of value with one choice only or multiple choice. It allows also the definition of taxonomy that modelises the list of terms in the triplestore using RDF model and allows the selection of authority source such as the Library of Congress subject Headings. In the taxonomy space, it was also necessary to create the authors of the contents in order to be able from the predefined relationships in Islandora to establish the association between them and their resources. Again, this way of doing things does not correspond to current usage but is necessary for the approach of semantic web which establish a formal link between representation objects.

C.8 Conclusion

This customization work was necessary in order to have a prototype as the other frameworks offer. It would have taken a little more time to go deeper but this effort allowed to perform most of the tests.

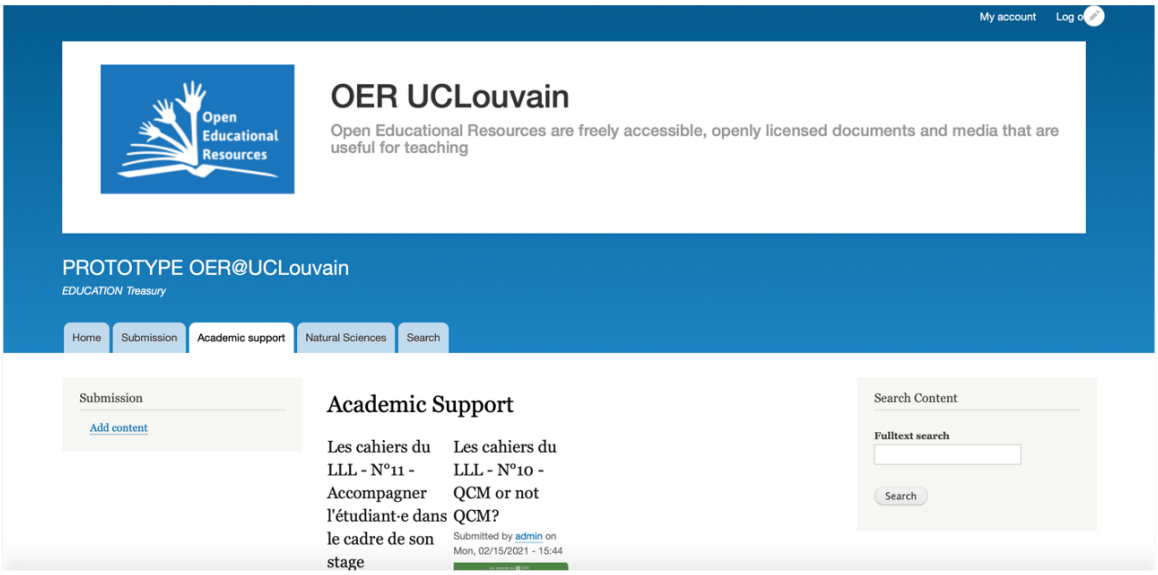


Figure C.4: Final UI of Islandora

Appendix D

Architecture maps

This appendix establishes a comparison framework that allows the different modules to be identified according to their role in the different platforms architecture. This framework makes it possible to identify the external modules and therefore the dependencies to be considered. This analysis was used to establish the conclusions that are presented in the technical comparison chapter.

D.1 Architecture framework

In order to set an architecture comparison frame, a five levels model has been used. The top level is a the application layer and is split in user-facing interface and machine interfaces that are used by other applications. The lower level is for storage, either internal or external storage. The level in the middle is for the internal business logic and the external building blocks that are used. Between the storage level and the business logic, there is a middleware level that ensure synchronisation between the upper level and the storage level. It also materialise resources transformation : either creating derivatives or using edition capabilities. The level between the application level and the business logic stands for the communication capabilities and the control of the run time environment.

The objective of this comparison is to assess the modularity of the architecture, to detect its specificities and dependencies.

D.2 DSpace architecture

DSpace has organized its components inside three layers. In order to be able to compare with the other products, the three-layer architecture of DSpace has been complemented in Figure D.1 by two layers of middleware corresponding to reliable

communication, runtime environment orchestrator, synchronisation modules and derivative tools/editors.

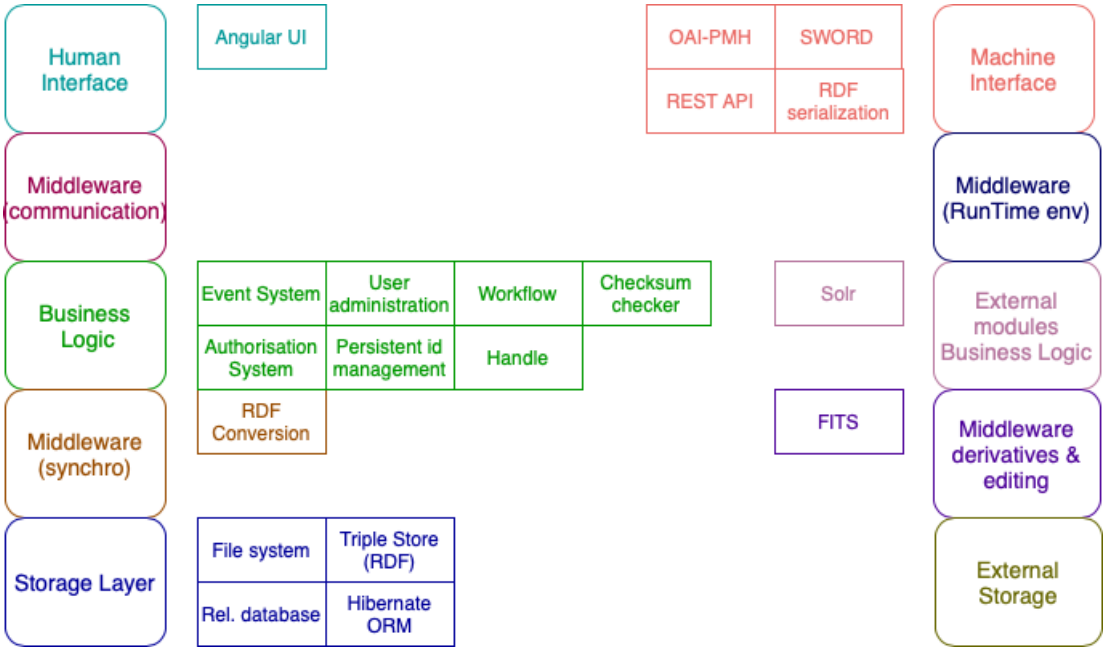


Figure D.1: DSpace's architecture

At the top level, the application layer includes interfaces for human and machine access. As of version 7, the application layer is divided into a front-end (user interface) and a back-end (server API) consisting of machine interfaces, as shown in Figure 5.1.

The DSpace front-end (all user-facing functionality) consists of user interfaces built on Angular. They use Typescript on the client side and node.js, npm and yarn as node package manager on the server-side. As defined with Angular, the interface is structured with modules that are part of components whose role is to display data in html templates, and using services that retrieve data for the components.
The DSpace backend consists of a API Server, built on Spring Boot. This backend server provides all machine-based interfaces, including the REST API (specified with well-defined contracts), OAI-PMH, SWORD (v1 and v2) and RDF.

On the third level, there is the internal Business Logic Layer, also called DSpace Java API layer, which provides the core business logic for all the various application interfaces: access control, workflow, checksum checker, handle (active access to a resource item), history recorder, ...

Finally, at the last level, the storage layer is a subset of DSpace-api dedicated to the management of content storage (metadata, relations, bitstreams) for all objects of the business layer. Hibernate ORM (persistent objects) is used on top of a relational database (Postgres or Oracle).Stored files (bitstreams) are supported on filesystem storage or Amazon S3 storage.
On the storage layer, there is also a triplestore such as Apache Fuseki, to support linked data. The metadata and relationships of public items are cached using RDF modelling. The triplestore can be queried using SPARQL and can export its contents in an RDF serialization that is interfaced with the public. A metadata RDF mapping file describes how to convert DSpace's internal metadata fields to RDF during the caching transfer.

D.3 Hyrax architecture

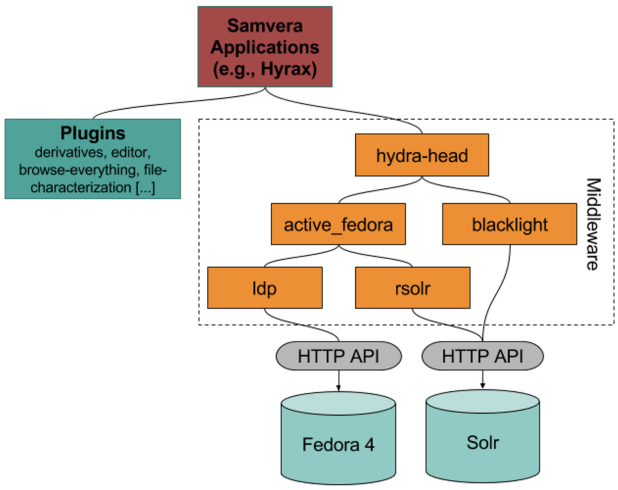


Figure D.2: Hyrax's simplified architecture

As shown in the simplified architecture Figure D.2, Hydra-Head is a Ruby-on-Rails module containing the core code for the web application.

Fedora is the historical persistence layer for content and related metadata storage as linked data. As of version 3, Hyrax offers a Valkyrie module to store resources either in Fedora or in another storage.

Hyrax uses Solr, an external open source module, for indexing and search ca-

pability. A middleware module called "Active_Fedora" indexes all storage contents into Solr. Hyrax interaction with Fedora and Solr are written using HTTP API.

Figure D.3 shows a summary of the component architecture.

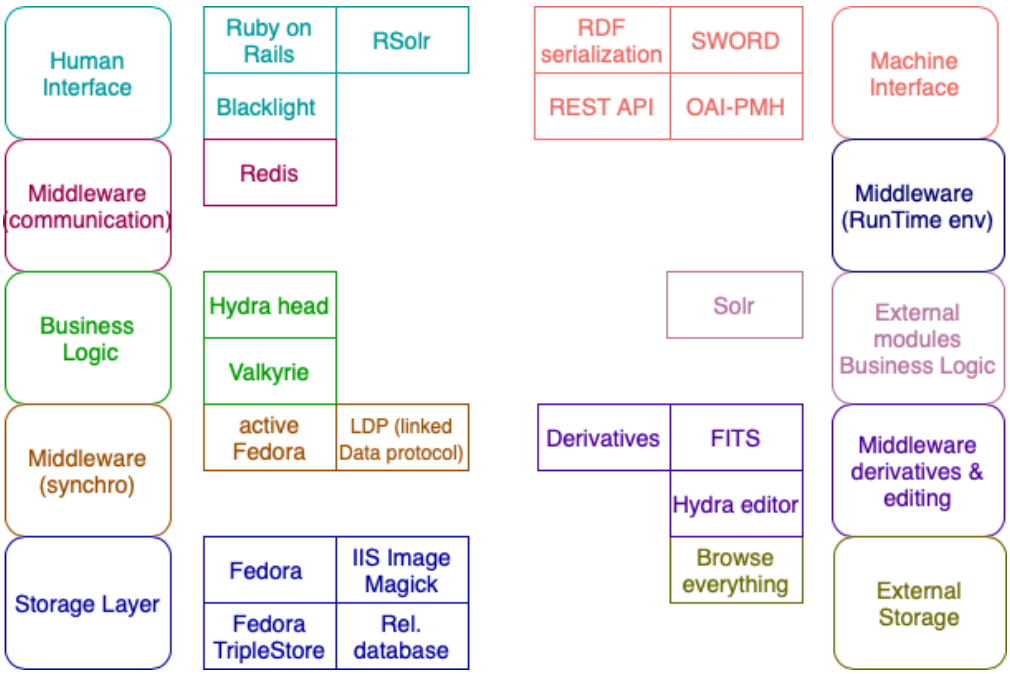


Figure D.3: Hyrax's architecture

Hyrax-suit contains several modules to ensure synchronisation between the storage and the search engine, but also to generate derivatives for uploaded contents.

- LDP (Linked Data Platform) is used for interaction with Fedora 4
- Rsolr is a ruby client for Solr.
- Blacklight is used to organise search and display with plugins such as Spotlight (for virtual exhibits) or GeoBlacklight (for geospatial data).
- Derivatives allows, for uploaded content,to generate, for example, thumbnails for large images, down-sampled audio and video for web steaming, or thumbnail snapshots of PDF documents.

- Hydra-file infers technical metadata using FITS (external module) to characterize files (kind of image encoding, Size,...)
- Hydra-editor provides a simple editor for objects.
- Browse-everything is a rails engine insuring access to files stored in the cloud (Dropbox, Skydrive, Google Drive, Box, and a server-side directory share).

D.4 Islandora architecture

Islandora is a middleware between the CMS Drupal (open source content management system providing a comprehensive user interface) and Fedora commons used as a central digital content repository. Starting from the user interface in Drupal, all the interactions between the modules are summarised in Figure 5.2.
Islandora components are microservices whose roles are either to create derivatives:

- FITS: to create technical metadata
- Homarus : to generate video and audio derivatives
- Houdini: to generate image-based derivatives
- Hypercube for OCR

or to synchronise between nodes in Drupal and resources items in Fedora to maintain consistency of a single content object.

- Millimer : to convert Drupal node entities into Fedora resources
- Recast: to remap Drupals URI and fedora links in the Fedora associated RDF store

As shown in Figure D.4, the Islandora installation deploys also other external open source components such as:

- ActiveMQ as message broker
- Karaf as OSGi runtime environnement
- Solr as Search and discover layer
- Blazegraph as triplestore and graph database
- Cantaloupe as IIIF image server for on-demand generation of high-resolution source image derivatives

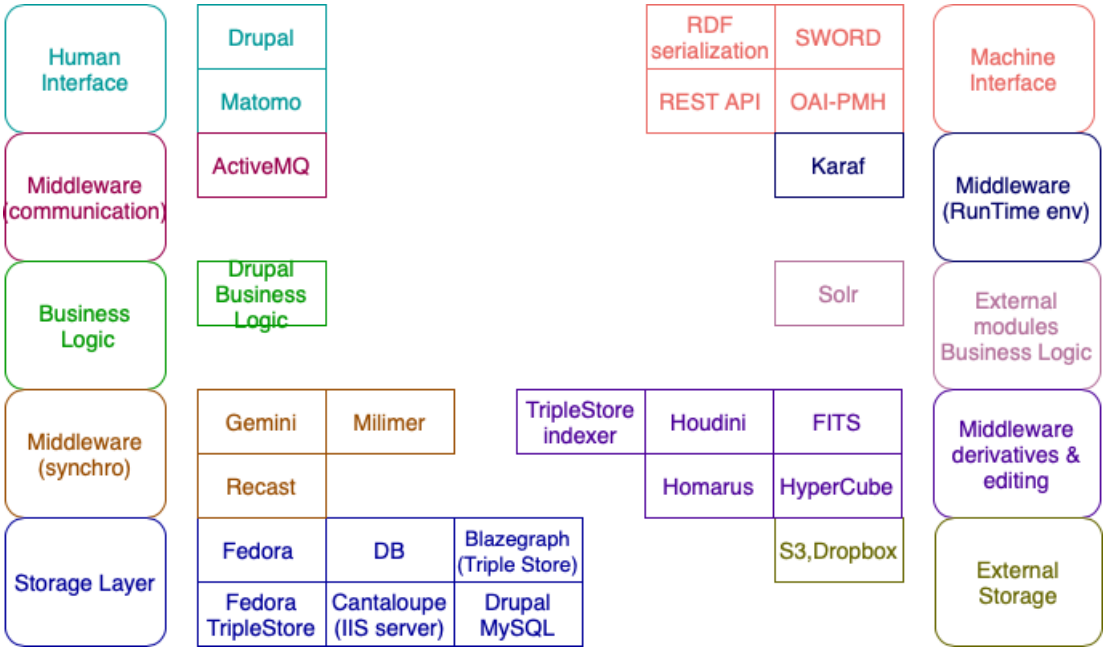


Figure D.4: Islandora's architecture

- Matomo as web analytics application to generate usage statistics and dashboards
- MySQL or PostgreSQL as relational Database management system.

Although Fedora is the recommended repository for content preservation, Islandora can also work with other storage systems.

Appendix E

RDF

This section presents the main aspects of RDF model. RDF model is used by all three platforms; as internal data structure for Fedora and as a copy of the repository content by DSpace. The graph format is called a RDF model. It relies in a triplestore[1] where triplets are a relation representation that associates :

- the *subject* which represents the resource to be described
- the *predicate* which represents a type of property applicable to this resource
- the *object* which represents a data or another resource as value of the property.

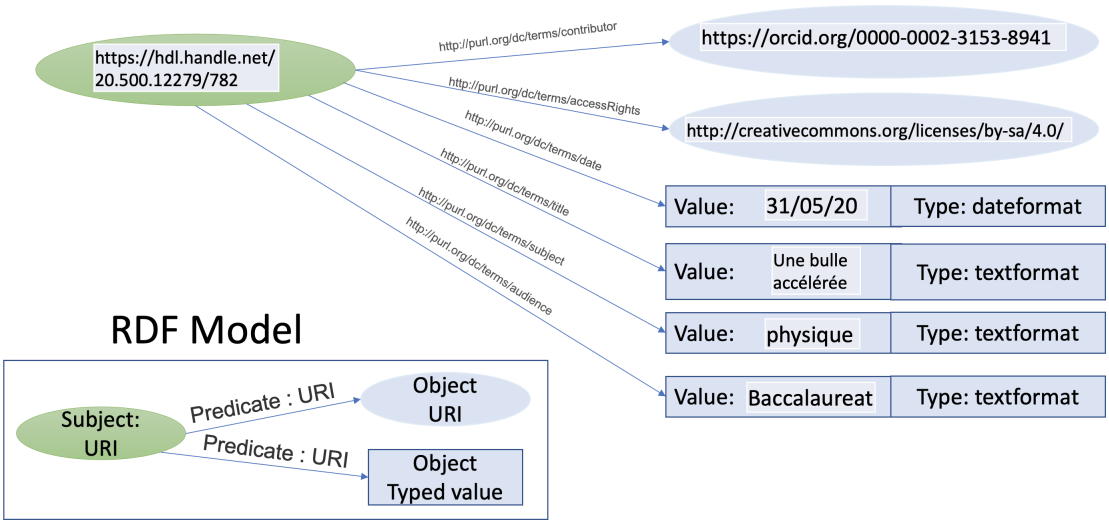


Figure E.1: Metadata OER RDF format

[1]Triplestore is RDF store specifically built for triplets

The set of triplets corresponds to a labeled oriented multigraph. Each triplet is an oriented edge whose label is the predicate, the source node is the subject and the target node is the object. When the controlled vocabulary is done through an authority control using RDF, the controlled value or term is an URI[2], a link to a digital object, rather than a string. This URI provides additional information about the term such as the vocabulary from which it originates, and its relationship to other terms. This RDF format evolution is valuable because it supports machine handling without requiring human contextualization capabilities.

Figure E.1 presents some metadata of an OER example. As you see, the relations are materialized by URIs. The URI links to informations describing how a relation or an entity has to be understood and manipulated. The effort going from a traditional hierarchical model to a RDF model requires to transform simple typed values into new entities that can be more complex but also more powerful to ensure more definition and less ambiguities.

[2]URI: identification of a published electronic resource

Appendix F

Data Model

This description of the data models demonstrate the difficulty that Islandora has in keeping the Drupal node in sync with the resources stored in Fedora. it presents the specificity of DSpace to ensure sub-communities and the ability of Hyrax to nest resources.

F.1 DSpace datamodel

The DSpace internal data model is composed of a community, a sub-community, a collection, an element having a one-to-one relationship with a DC record, and a set of bitstreams having a bytestream type.

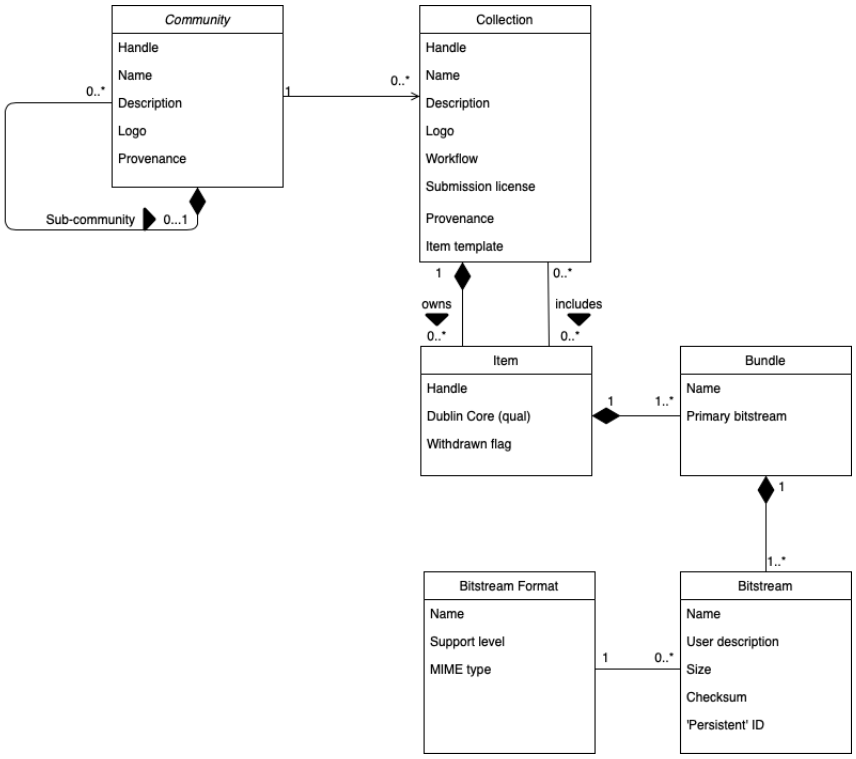


Figure F.1: DSpace datamodel

F.2 Hyrax datamodel

The internal Hyrax datamodel is composed of AdministrativeSet, collection and work that is composed of files.

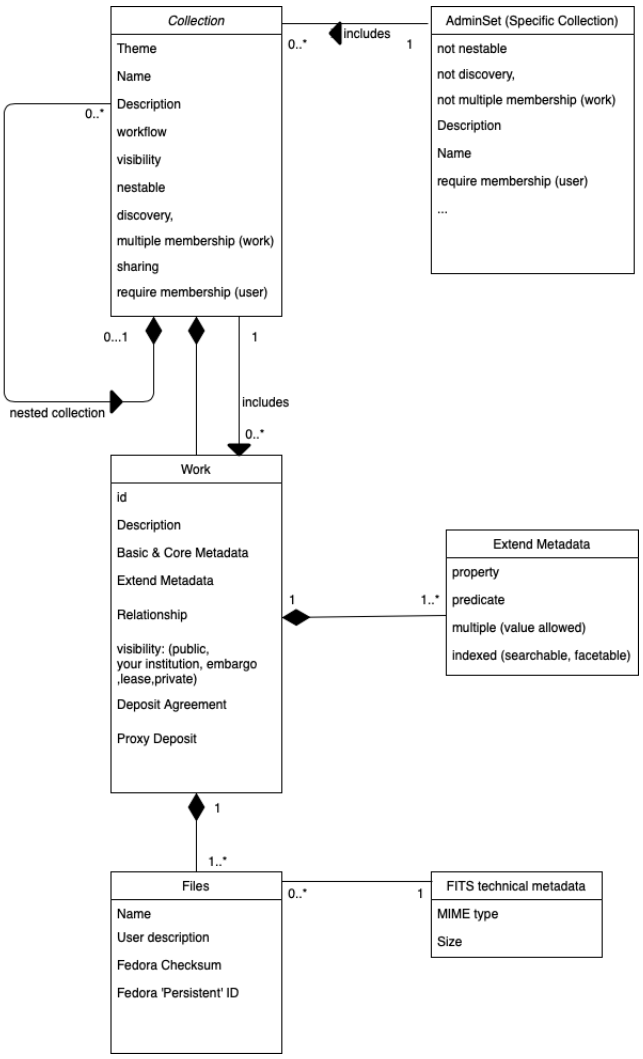


Figure F.2: Hyrax datamodel

F.3 Islandora data model

The Islandora data model consists of collection and contents that are each composed of a drupal node and media which are contained in Fedora or in another Storage system.

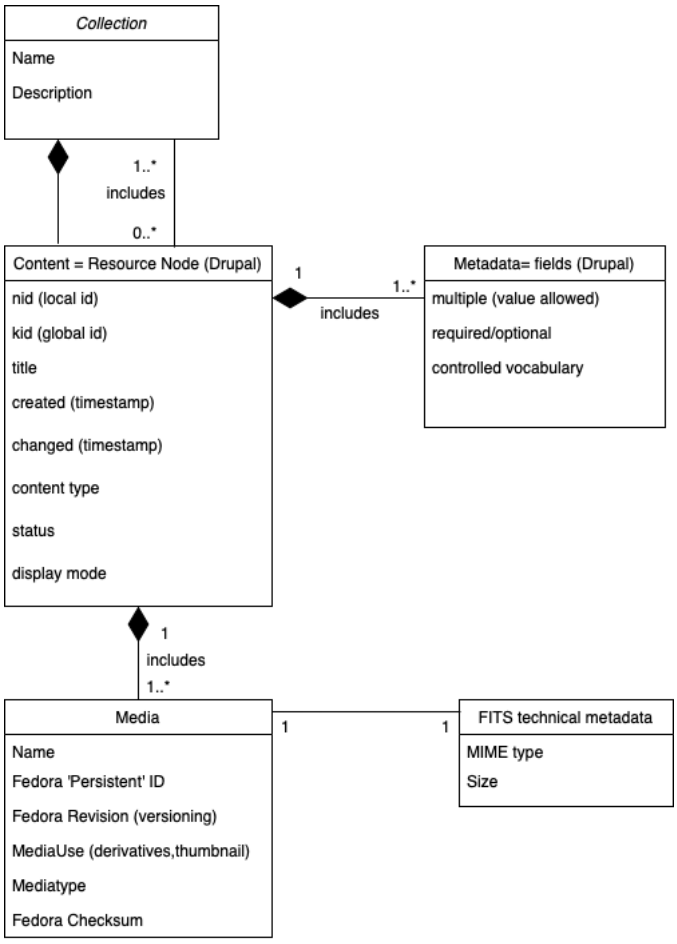


Figure F.3: Islandora data model

Bibliography

[1] Helene Brinken, Martin Mehlberg, and Lambert Heller. The open science training handbook: Written by 14 international experts during the foster book sprint. In *Open Education Global Conference (oeglobal2018), Delft, The Netherlands, 24-26 April 2018*. Zenodo, 2018. accessed date: 2021-06-7.

[2] Supported by European Commission. Up2u european commission project: learning up to university. accessed date: 2021-02-26. URL: `https://up2university.eu`.

[3] Lorna M Campbell and Phil Barker. Lrmi implementation: Overview issues and experiences. *Bolton, UK: Cetis*, 2014. accessed date: 2021-05-28.

[4] Todd Carpenter, Michael Nelson, Bernhard Haslhofer, Shlomo Sanders, Richard Jones, Robert Sanderson, Martin Klein, Herbert Van de Sompel, Graham Klyne, Paul Walk, Carl Lagoze, Simeon Warner, Stuart Lewis, Zhiwu Xie, Peter Murray, and Jeff Young. Resourcesync framework specification. accessed date: 2021-04-17. URL: `http://www.openarchives.org/rs/1.1/resourcesync`.

[5] Fedora community. Fedora features. accessed date: 2021-04-12. URL: `https://duraspace.org/fedora/about/features/`.

[6] Esme Cowles. Valkyrie, reimagining the samvera community. accessed date: 2021.27.03. URL: `https://library.princeton.edu/news/digital-collections/2018-06-05/valkyrie-reimagining-samvera-community`.

[7] DCMI. Learning resource metadata initiatitive specification. accessed date: 2021-05-02. URL: `https://www.dublincore.org/about/lrmi/`.

[8] Nicolas Delestre, Yolaine Bourda, Erwan Le Gall, Rosa-Maria Gomez de Regil, Yohan Colmant, Marie Peterlongo, and Romuald Verrier. Nodefr-1: Métadonnées pédagogiques et web des données. In *Environnements Informatiques pour l'Apprentissage Humain*, 2019. accessed date: 2021-04-13.

[9] Content Directorate-General for Communications Networks and Technology. Shaping europe's digital future:european open science cloud, 2021. accessed date: 2021-06-06. URL: https://digital-strategy.ec.europa.eu/en/policies/open-science-cloud.

[10] Lieven Droogmans. Dspace 7 - the power of configurable entities, November 2019. accessed date: 2021-05-26. doi:10.5281/zenodo.3554102.

[11] Institute for the Study of Knowledge Management (ISKME). Oer commons learning object metadata element set. accessed date: 2021-05-26. URL: https://www.micollegesonline.net/uploads/ckeditor/files/OERCv4-MetadataApplicationProfile.pdf.

[12] Víctor J García-Morales, Aurora Garrido-Moreno, and Rodrigo Martín-Rojas. The transformation of higher education after the covid disruption: Emerging challenges in an online learning scenario. *Frontiers in Psychology*, 12:196, 2021. accessed date: 2021-06-07.

[13] IMS Global. Learning tools interoperability for learning management system. accessed date: 2021-04-07. URL: http://www.imsglobal.org/activity/learning-tools-interoperability.

[14] Mokhtar Ben Henda. Etude sur les portails et agrégateurs des ressources pédagogiques universitaires francophones en accès libre. *arXiv preprint arXiv:1703.01072*, 2017. accessed date: 2020-11-23.

[15] Open Archives Initiative. Oai-ore: Open archives initiative object exchange and reuse. accessed date: 2021-03-07. URL: https://www.openarchives.org/ore/.

[16] Petr Knoth, Matteo Cancellieri, and Martin Klein. Comparing the performance of oai-pmh with resourcesync, 2019. accessed date: 2020-12-10.

[17] Carl Lagoze, Herbert Van de Sompel, Michael Nelson, and Simeon Warner. The open archives initiative protocol for metadata harvesting. accessed date: 2021-04-17. URL: http://www.openarchives.org/OAI/openarchivesprotocol.html.

[18] Fengchun Miao, Sanjaya Mishra, Dominic Orr, and Ben Janssen. *Guidelines on the development of open educational resources policies*. UNESCO Publishing, 2019. accessed date: 2021-03-29.

[19] Tomasz Neugebauer, Pierre Lasou, Andrea Kosavic, and Tim Walsh. Digital preservation functionality in canadian repositories. 2019. accessed date: 2021.03.18.

[20] Confederation of Open Access Repositories. Coar community framework for best practices in repositories., October 2020. accessed date: 2020-12-08. doi:10.5281/zenodo.4110829.

[21] Benoît Raucent, Céline Mathelart, Yves Deville, Françoise Docq, Christine Jacqmot, and Pascale Wouters. *Open Education*. Louvain Learning Lab, 2019. accessed date: 2021-06-07.

[22] Jenn Riley. Understanding metadata. *Washington DC, United States: National Information Standards Organization (http://www. niso. org/publications/press/UnderstandingMetadata. pdf)*, 23, 2017. accessed date: 2021-05-07.

[23] E Rodrigues, A Bollini, A Cabezas, D Castelli, L Carr, L Chan, and K Yamaji. Next generation repositories. *Behaviours and Technical Recommendations of the COAR Next Generation Repositories Working Group*, 2017. accessed date: 2020-12-10.

[24] Samvera. Differences hyrax-hyku. accessed date: 2020-12-10. URL: https://wiki.lyrasis.org/pages/viewpage.action?pageId=85530575.

[25] Gema Santos-Hermosa, Núria Ferran-Ferrer, and Ernest Abadal. Repositories of open educational resources: An assessment of reuse and educational aspects. *International Review of Research in Open and Distributed Learning: IRRODL*, 18(5):84–120, 2017. accessed date: 2021-04-04.

[26] SWORDAPP.org. Simple web-service offering repository deposit specification. accessed date: 2021-03-14. URL: http://www.swordapp.org/about/.

[27] Mohammadreza Tavakoli, Mirette Elias, Gábor Kismihók, and Sören Auer. Metadata analysis of open educational resources. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, LAK21, page 626–631, New York, NY, USA, 2021. Association for Computing Machinery. accessed date: 2021-05-27. doi:10.1145/3448139.3448208.

[28] UCLouvain. Digital university, oer motivations. accessed date: 2020-09-30. URL: https://uclouvain.be/en/digital-university/oer.

[29] UNESCO. New unesco recommendation will promote access to educational resources for all. accessed date: 2021-03-25. URL: https://en.unesco.org/news/new-unesco-recommendation-will-promote-access-educational-resources-all.

[30] H Wickline. Open educational resources: Breaking the lockbox on education. *California, USA: William and Flora Hewlett Foundation*, 2013. accessed date: 2021-06-07.

[31] wikipedia. Dspace wikipedia presentation. accessed date: 2021-04-12. URL: `https://en.wikipedia.org/wiki/DSpace`.

[32] wikipedia. Fedora wikipedia presentation. accessed date: 2021-04-12. URL: `https://en.wikipedia.org/wiki/Fedora_Commons`.

[33] wikipedia. Islandora foundation: presentation. accessed date: 2021-02-18. URL: `https://en.wikipedia.org/wiki/Islandora`.

[34] wikipedia. Samvera (hyrax) wikipedia presentation. accessed date: 2021-04-12. URL: `https://en.wikipedia.org/wiki/Samvera`.

[35] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3(1):1–9, 2016. accessed date: 2021-06-08.