POLITECNICO DI TORINO
Repository ISTITUZIONALE

A new architecture for cross-repository creation and sharing of educational resources

Original
A new architecture for cross-repository creation and sharing of educational resources / Favario, Leonardo; Masala, Enrico. - In: INTERNATIONAL JOURNAL: EMERGING TECHNOLOGIES IN LEARNING. - ISSN 1863-0383. - 12:2(2017), pp. 185-209. [10.3991/ijet.v12i02.6058]

Availability:
This version is available at: 11583/2669149 since: 2017-04-14T09:25:18Z

Publisher:
IAOE

Published
DOI:10.3991/ijet.v12i02.6058

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

12 October 2024

INTERNATIONAL JOURNAL OF EMERGING TECHNOLOGIES IN LEARNING

A New Architecture for Cross-Repository Creation and Sharing of Educational Resources

https://doi.org/10.3991/ijet.v12i02.6058

Leonardo Favario
Control and Computer Engineering Department, Politecnico di Torino, Italy
leonardo.favario@polito.it

Enrico Masala
Control and Computer Engineering Department, Politecnico di Torino, Italy
enrico.masala@polito.it

Abstract— A large amount of educational resources are currently available in the Internet, covering educational needs for many school grades. However, using such wealth of material, typically dispersed in several repositories, in a simple and effective way, is rather challenging due to the difficulties encountered by teachers in learning the peculiar access and operational procedures that each repository system requires. Therefore, it is important to provide the teachers with a centralized, integrated, simple system that can address most of their needs so that every operation (search, edit, share, download) can be done from a single location. This work follows this direction by designing and presenting both an architecture to integrate different repository systems using the Content Management Interoperability Services (CMIS) API and an integration layer that provides a simple web interface suitable for the needs of both the content creators (teachers) and the users of the contents (learners). Results have been evaluated both quantitatively, i.e., using performance indicators such as response time, and qualitatively, on the basis of the user experience evaluated through a questionnaire. Both type of results show that the platform adequately addresses user needs therefore it has the potential to be embraced by a large user community.

Key Words—computer uses in education, e-learning tools, CMIS, open source

1 Introduction

In the latest years the interest for e-learning systems is steadily increasing due to the many advantages they are expected to deliver, e.g., lower costs, better course adaptation for students, remote delivery, etc. [1]. All systems rely on an infrastructure providing content repository features, either generic or specifically designed for educational purposes. Examples belonging to the former category are Alfresco [2], Nuxeo [3], Sharepoint, etc., whereas for the latter category Learning Object Repositories (LORs) such as Merlot [4], MIT OCW [5], Ariadne [6] constitute a good

example. Learning Management Systems (LMS) also integrate (or can connect to) a LOR, but their distinctive characteristic is to provide content creation features. Examples are Moodle, Edmodo, Blackboard, etc.

The abundance of such software suites led to the current situation in which there is a strong fragmentation in the e-learning software suites adopted by even very similar education institutions [7]. For instance, our university relies on Alfresco while the other university in the same city is using Nuxeo. Clearly, a situation in which each institution proceeds on its own is not ideal since network and saving effects cannot be exploited. The Italian Ministry of Education, for instance, is pushing towards unification efforts by supporting projects for the creation of mini-portals and similar initiatives. However, a large amount of material has been made available in each of those institutional repositories and it would be desirable to be able to reuse it, without waiting for major infrastructure changes, through simple means, e.g., uniform access platforms and interfaces.

Some solutions trying to address the fragmentation and heterogeneity of repositories have been proposed in the past. The latest trend is to use Unified E-learning Repositories (UER) that attempt to simplify sharing resources between different institutions using a single access point and solution [8]. However, several integration issues need to be solved, in particular for distributed searches [9], [10] and communication between repositories [11].

Interoperability requires a common framework both for interfaces and queries. The Simple Query Interface (SQI) [12] layer is a tentative to introduce a standard framework which has been adopted by many LORs. Still, to correctly adhere to this approach, wrappers have to be developed to convert a query from the common language to the proprietary one and vice-versa. In a heterogeneous environment, this process could be highly consuming both in terms of time and resources.

Scenarios which only include homogeneous repositories are, of course, much easier to handle and in the best case they can seamlessly interconnect forming a robust network [13]. However, many of those solutions lack the possibility to allow each teacher to use any available resource in the network and structure them as desired into their own course. Other solutions provide such characteristics but pose constraints on both servers and software, and require dedicated network such as in [14].

For the case of several heterogeneous general purpose repositories, easy integration procedures are clearly difficult to achieve. The Content Management Interoperability Services (CMIS) specification [15] aims to overcome such difficulties by providing a standardized API that is now supported by several content management systems, including Alfresco and Nuxeo. The CMIS API allows to perform all the basic operations such as querying, inserting, and deleting and a limited management of metadata attributes. Although not all features and attributes are available through this interface, this seems to be a good starting point to develop a unifying interface for the creation of an integrated LO repository which is able to provide a seamless access to any of the content available in any of the repositories in the network.

Motivated by the lack of a solution that effectively addresses the issues of a scenario involving heterogeneous repositories, as well as by the recent availability of

such an interoperability layer, we designed our Free Architecture for Interoperable Repositories (FAIR). Such architecture contributes to fill the gaps in achieving the following objectives, which are inspired partly by [16] and partly by the requirements of the Italian Ministry of Education about its preference for open source solutions in the public administration:

1. use of open source software only;
2. all contents available under a permissive license (Creative Commons BY-SA 3.0 IT [17]);
3. simple web user interface both for teachers and learners;
4. transparent web-based handling, manipulation and sharing of objects in different repositories;
5. quality of content (achieved, e.g., through a review process);
6. optimized performance (including producing different formats suitable for various types of devices).

Objectives #1 and #2 are dictated by our aim to make it one of the e-learning reference platforms at the national Italian level. According to [16], the issues mentioned in #3, #4, #5 and #6 are key factors that can either motivate or act as barriers for teachers in using online repository systems for educational purposes. Therefore, we carefully took them into account during the design and evaluation of our architecture.

Our proposal aims to connect together, with minimal effort, many of the existing institutional repositories so that the available resources can be easily and uniformly accessed by both teachers and learners. This approach has the advantage of making available trusted contents which have been typically produced or revised by competent people in the institution. Moreover, a uniform access and management interface ensures that teachers only need to learn one simple system and not the many systems currently available in each single institution. Note that this approach is also suitable to integrate contents originating not only from schools but also from museums and science associations that might have content repository systems containing resources that are basically ready to be used and shared [18].

Note that currently there seems to be no solution with all these characteristics, as it will be discussed in more details in Section 2 that presents a throughout comparison with existing systems. Also, as part of objective #6, our architecture is designed so that each newly created resource automatically includes a ready-to-go HTML5-based presentation of the content which can be used directly within the browser in both online and offline mode. This is a feature that has been particularly appreciated by the users as reported in Section 4.2.

We experimentally evaluate our proposed system in terms of both technical performance parameters and user feedback. The technical evaluation focuses mainly on the response times of the web application, whereas users' opinions are collected and analyzed through a questionnaire. Some qualitative characteristics such as usage simplicity and intuitiveness of the interface can, in fact, only be evaluated by directly asking to the users. Given the good technical results and user opinions we believe that our architecture has the potential to be a starting point for a larger project involving

several school grades and more institutions. Moreover, this project has been proposed for consideration at the Italian Ministry of Education to become one of the possible platforms for a national initiative in online learning for Italian schools.

The paper is organized as follows. Section 2 presents the related work in the field. In Section 3 the architecture of the proposed system is presented in details, followed by some experimental results, both quantitative and qualitative, in Section 4. Conclusions are drawn in Section 5.

2 Related Works

A large number of educational contents in digital format have been created in recent years. Such contents are generally referred to as learning objects (LOs). Borrowing from the work of the IEEE Learning Technology Standards Committee (LTSC) [19] and the Centre for Excellence for Teaching and Learning in Reusable Learning Objects (RLOCETL) [20], as well as from [16][21][22], we define LOs as digital entities, with instructional value, that can be used, reused or referenced during technology supported learning. Moreover, they are tagged by some metadata to be easily found by searches.

LOs are generally archived with their metadata in learning object repositories (LOR) [16]. LORs typically allow to collect LOs in different formats [23], [24], [25], however a number of reasons may prevent content creators from following such standards, for instance the lack of support from tools and their familiarity with the tools [26]. Despite the potential issues due to the different formats [27], their online availability allows teachers to easily compose LOs into courses using specific, although sometimes complex, software packages. Most of them (e.g., Moodle [28], Edmodo [29]) come bundled together with a LMS, therefore they require the teacher to be fairly skilled in exploiting the features of the specific LMS.

The turning point in the diffusion of LORs usage is probably when the Open Educational Resource [30] has been formally defined. In fact, if no permissions are granted or they are just unclear about a LO, this results in a scarce usage and diffusion. Such clear definition helped to remove any ambiguities in terms of right of use, remix, modify, redistribute [31] of the contents, with great benefit for the diffusion of freely available resources and LOs in particular, though in practice bootstrapping a culture of sharing to facilitate the creation of open educational resources is not easy [32].

A huge number of open access repositories exists (from 2013 at least 635 repositories of this kind have been created [33]), however the different nature and peculiarities of all these repositories make it very difficult to integrate all the work spent in setting up repositories and the corresponding learning materials in few, large-spectrum repositories that can be accessed in a more uniform way from a single point. This is a critical aspect of any LMS as the vast majority of the institutions supporting the use of LMS (e.g., schools) would like to reuse content already available in other LORs or at least allow an easy integration of such content in their system [34].

Table 1. Comparison of the most widespread systems in terms of the FAIR objectives established in Section 1. Only shortcomings are reported.

Name	Open Source	Content License	Simple web user interface	Transparent handling of heterogeneous resources	Content review	Optimized performance
GLOBE/ ARIADNE [9], [35]	Source code partially available	Uploaded content handled as public domain.	ARIADNE finder can be used online. ARIADNE Moodle Search Widget can be integrated inside the Moodle CMS.	Federated access to multiple repositories but each type of repository needs to be handled by specific adaptation layers. URLs for federated access are not transparent. Adding new repositories may require to add new adaptation layers.	Each repository applies its own policies for review. A global policy does not apply.	Metadata of all repositories need to be initially harvested in a central location to run queries.
Merlot [4]	Not open. Fees required to join as a partner institution	Content must indicate copyright, usage costs, etc.	MERLOT web page has a non-responsive design. Different modules exist for several LMSs. Also a set of APIs is available not for free.	The federated access is not transparent: URLs are specific for each repository. Support multiple repositories if they have compatible metadata definition and protocol for queries.	A referee policy is needed for each attached repository.	Metadata of all repositories need to be initially copied in a central location to run queries.
LON-CAPA [14]	Free and Open Source.	Content has no associated license. Remixing is allowed.	The web interface makes it possible to browse for contents and create new assessments. A dedicated metacourse is necessary to make use of the platform.	Resources have a unique and persistent path (on a single virtual file system). Users can see resources as simple files on a network disk or by means of a web application.	The adoption of a resource inside a course represents the approval step of a review process.	Non-trivial client-server architecture. Home servers, Library Servers and Access Servers have to communicate and replicate resources.
POOL/ SPLASH [36]	Free and Open Source.	Specific licenses can be used for each content	Need to download a desktop application. Splash creates a folder on the local computer which is the shared repository.	Can search material in other nodes as in a P2P network. Nodes must use the POOL protocol.	Being P2P content cannot be easily reviewed before publication.	P2P may not provide very efficient searching.
LionShare [37] (no longer active)		May be covered by copyright.	Need to download a desktop application. Shared repository in a local folder. Upload to institutional servers available.	Use a P2P approach to search material in other nodes.	Being P2P content cannot be easily reviewed before publication.	P2P may not provide very efficient searching
Orange Grove [38]	Based on commercial software Equella.	Contents are free for educational purposes, may contain copyright statements	Possibility to integrate with well-known LMS through a module in CANVAS, Blackboard, etc. It is based on Java.	Content upload is only on a central repository (but it can run federated searching). Only searches are federated. LOs cannot be combined together to create other objects.	Not clear if FHE faculty content is reviewed or not.	
Wisc-Online [39]	No.	Educational content is provided for free under a CC BY-NC license.	Through the web application it is possible to browse for contents but not to upload new contents. Access to APIs exists.	Web-based centralized repository architecture. LOs cannot be combined together to create other objects.	LOs developed by faculty, for faculty, with assistance from the development team.	Centralized repository, it is not possible to draw from a distributed set of repositories.
LORSE [40]	Seems not available		It is not standalone: it is a module to be integrated in other LMS, such as Moodle.	Support federated queries only over web-service compatible repositories such as Merlot, Connexions, Fedora; for each one of them a specific agent is necessary to interpret results.		

To overcome such difficulties, different efforts have been done for creating standard ways to harvest metadata and resources. The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) provides a framework for harvesting metadata from different compatible repositories, including content URLs if available [41]. The MACE initiative uses OAI-PMH to harvest metadata but queries are performed by means of a centralized web interface [42]. Also the distributed querying issue has been investigated in the past. Results converged into the definition of the Simple Query Interface (SQI) as a universal interoperability layer for educational networks [12]. However, the integration of this framework still requires a high load on the repository administration side, since all the requests and responses have to be translated from the SQI language to the proprietary one of the learning repository. Moreover, such a technology does not consider pulling the content but only metadata, whereas the Simple Publishing Interface (SPI) aims at overcoming this issue by specifying yet another interface. In a distributed environment, such as, e.g., the MACE, MELT [43] or ASPECT [44] projects, OAI-PMH is used to harvest metadata in a central location, called harvester, and then, using SPI, the collected information is pushed on a metadata store. Finally, SQI is used to query this metadata repository.

One of the latest trends in integration is to use Unified E-learning Repositories (UER) that attempt to simplify LOs sharing between different institutions using a single access point and solution [8]. However, several integration issues need to be taken into account, for instance secure low-layer communication between repositories [11] and federated querying of the repositories in the network [10]. It is clear that the easiest scenario includes a number of homogeneous repositories which can, in the best case, seamlessly interconnect forming a robust network. For instance, the CampusConnect [13] project offers both the students and the teachers the possibility to use educational contents of other universities in the network using their centralized e-learning community server that is accessed transparently from each LMS interface of each single university.

To better visualize the current situation, Table 1 compares the most widespread repository systems in terms of the FAIR objectives stated in Section 1. The table shows that no existing solution is able to completely fulfill our requirements for a variety of reasons.

For instance, the GLOBE [9][35] initiative allows to express a query by connecting to a centralized point which then propagates it to a federated set of appropriately configured repositories included in the network [10]. The relatively old project Merlot [4] is an example in which several academic institutions collaborated to form a vast homogeneous repository of ready-to-use contents for faculty and students. However, it requires compatible metadata definitions among repositories as well as the use of the same protocol as Merlot for querying. An alternative approach is to use a virtual shared file system as in the case of LON-CAPA [14], which however does not easily allow to structure resources into courses.

Peer-to-peer (P2P) systems have also been proposed to address repository federation issues. POOL/SPLASH [36] and LionShare [37] require to download a P2P application which then acts as a part of the repository network, holding some LOs on its local file system. However, this approach would pose several problems for

the FAIR objectives, such as requiring program installation, and careful management of the local LO storage. Moreover, preventing content publication before review is extremely difficult in a P2P scenario due to data integrity issues, although approaches have recently been proposed to avoid tampering [45]. Other systems focus on receiving fragments of contents in parallel from different sources, so to minimize the downloading time [46]. Other software include Orange Grove [38], based on the commercial software Equella, which offers the possibility to run federated searches. However, it imposes several limitations, e.g., LOs cannot be combined together to create another LO. Moreover, new content needs to be uploaded to a central repository. Wisc-Online [39] is a web-based centralized repository architecture aimed at giving the possibility to faculty to develop LOs for other faculties with the support of a development team. This relatively mature project, however, does not have a procedure to integrate existing repositories. Moreover, the source code running the platform seems to be not available. Finally, LORSE [40] is a module to be installed and integrated in other learning management systems which supports federated searches. However for each of the repository in the network a dedicated agent is needed to interpret the results.

In the latest years, with the increase in the availability of cloud infrastructures, some novel approaches tried to embrace this new set of technologies. The so called cloud-based e-learning model introduces scale efficient mechanisms [47] enabling fast deployments for the system administrators and better experiences for the users taking advantage of the distributed nature of the infrastructure (e.g. using CDNs) [48]. However, the use of cloud infrastructure as a provisioning method exposes to different issues in terms of privacy and sensitive data handling. Furthermore, the use of cloud infrastructures can help in distributing the nodes among a broad geographical area but nevertheless all the aforementioned issues regarding how to collect and treat LOs coming from different sources have to be tackled.

From the previous description it is clear that, for the case of many heterogeneous repositories, easy integration procedures are difficult to achieve and no currently available solution is able to completely fulfill our objectives stated in Section 1. The recently proposed CMIS specification [15] aims to overcome integration difficulties by providing a standardized API that is now supported by several content management systems including Alfresco and Nuxeo. This API is particularly useful for the design of the seamless cross-repository access and management feature of FAIR since it provides an all-in-one solution for tackling all FAIR needs of pulling, pushing and querying metadata and contents.

3 The FAIR Architecture

The aim of this project is to define and create a set of interoperable tools giving the users the possibility to define their own personal learning path leveraging the great variety of materials available on the interconnected repositories network. In this work, similarly to [49], the term *learning path* defines a chaining of LOs that guide the learner into its learning experience.

3.1 Overview

The main entities involved in the architecture are shown in Fig. 1 and are summarized in the following with specific reference to the objectives highlighted in Section 1.

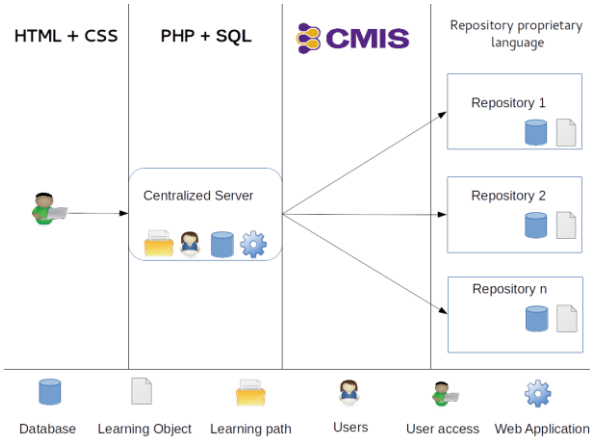

Fig. 1. Architecture of the proposed system.

With respect to achieving the first objective highlighted in the introduction (use of open source software only), the tools are composed of software developed by the authors of this work and licensed under the GNU General Public License (GPL). Any dependency is only on other open source software. In more details, the infrastructure supporting the web application is a standard LAMP environment: Linux, Apache, MySQL and PHP. The web application itself is based on the Drupal 7 and Bootstrap frameworks which are, again, open source software.

The second objective, i.e., content under a permissive license, is achieved by imposing that uploaded LOs have to be licensed under a Creative Commons CC-BY-SA 3.0 IT [17] clause.

Every functionality is available both to the teachers and the learners by means of a web interface. Through the same interface it is possible to access, manipulate and share the objects without relying on specific, locally installed, software which is often difficult to handle for less expert users. The system can handle a variety of LOs, ranging from simple ones (i.e., text files and images) to more elaborate ones such as SCORM [23] files which seems to represent the foundation for building interoperable e-learning repositories [50][51]. Each LO is enriched by proper metadata whose values are stored in the repository where the LO resides. Moreover, LOs can be combined together in a so called complex LO (CLO) in which the single elements can be explored in a predefined order suggested by the composer (i.e., the teacher). According to [52], a CLO is defined as an LO whose instructional material is an aggregation of learning objects. Being an LO, a CLO can be treated exactly as any

other LO. However, for maximum flexibility, it is always possible to access each single LO which forms the CLO in an individual manner.

Due to web-based nature of the platform, access to the LOs is completely transparent and independent of the actual physical location of the LO itself. From the technical point of view, this is achieved through the use of the CMIS standard. For testing purposes Alfresco Community Edition (CE) and Nuxeo have been selected but other CMIS-compatible solutions can be easily included. Both Alfresco CE and Nuxeo are open source software.

The web application allows to define different access permissions to the LOs, ranging from guest access that does not need authentication to privileged users that can ensure that the published content meets certain quality standards by means of a review and approval process, therefore allowing to control the quality of the LOs as required by the objectives in Section 1.

Finally, great attention has been paid to avoid performance issues in the use of the web application. Such issues can typically arise when dealing with objects residing in several different repositories and when serving the users heavy content such as video. In the former case, parallelization strategies have been implemented to minimize latency times during both querying and transfer of objects from the repositories. For the latter case, objects are converted and made available in different formats in order to match the needs and capabilities of each user device. This particularly applies to video which typically has restrictions in terms of accepted format and resolution depending on the type of device (smartphone, tablet and laptop).

The web application presents a single interface for both teachers and learners. A standard login form allows to access a dashboard that offers several tools, personalized depending on the user role. In particular, it is possible to perform several actions: upload of LOs with metadata information, search by querying the database of all repositories at once, selection of the objects of interest to save the search or to perform actions such as export, sequential merge, cut and paste with page granularity. This set of operations is possible only on LOs which are in the form of text documents or presentations. If different types of objects are involved, sequential merge operations are possible in the form of handling the inclusion of their URLs in a text document containing a list, or embedding them into an HTML5 presentation or in a compressed archive. The final LO, resulting from the previous operations, can be downloaded, shared or exported in different formats (e.g., in an HTML5 presentation). Note that the merge and paste operations create a new LO that is considered a complex LO for which only the list of operations to apply to the original LOs is saved.

Finally, note that recently FAIR has been modified to support distance learning sessions through the use of BigBlueButton [53] that is an open source web conferencing system for on-line learning. Such a system integrates many functions typically used in education, e.g., slides presentation, drawing tools and a chat system, in addition to audio and video communication among a multitude of connected users. The system is fully integrated with FAIR: for instance, it can seamlessly use all the available LOs for presentation, and FAIR can make available all sessions recorded through BigBlueButton as independent LOs.

3.2 Detailed Description

Modules The architecture is composed of a series of modules that work in a multi-user environment. To develop the first prototype Drupal was selected as a framework because of its features. In particular, it provides 1) easy user management capabilities; 2) modularity; 3) scalability. The modules developed on the Drupal side are independent from each other and from the core. This design gives administrators maximum flexibility: enable a certain set of tools, disable the unused ones, upgrade the core and develop newer tools independently. The choice of using Drupal was also favored by the availability of a strong community that supports and maintains the modules.

Drupal is linked with a MySQL database which maintains the user information regarding the personal dashboard, the comments on existing files and the statistics on the usage of the platform. The repository side is separated from the rest of the tools, that is, that it can run on another machine, also in another physical location. This is fundamental to allow to connect numerous repositories to one single instance of the web application, increasing the amount of LOs available to the user.

Repositories The web application can support several different types of repositories, provided that they implement the CMIS specification as later explained. As an example, in the following we discuss the integration with Alfresco CE. In this particular case, the advantages of using Alfresco CE are: 1) nearly complete CMIS mapping (i.e., CMIS API were almost all already implemented and supported); 2) strong community driven development; 3) amount of documentation available; 4) selection by numerous institution as their repository system. Alfresco CE, as many other repository systems, includes both a file system that is in charge of storing the contents and a database server (PostgreSQL) used to store all the metadata related to the files. The latter is necessary to define an ontology to catalog contents in the repository. In the first design phase we intended to use a simple ontology containing at least the 6 terms indicated in Fig. 2.

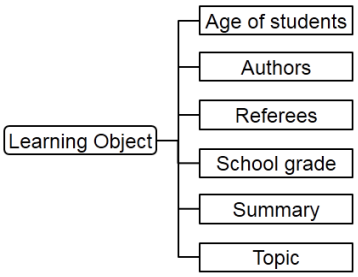

Fig. 2. Sample of the ontology needed for the LOs.

However, in a heterogeneous repository environment the creation of an ad-hoc, non-standard ontology leads to inconsistency issues each time a new repository is

connected. In order to avoid this situation we analyzed the international ontology standards and we eventually opted for the Dublin Core Metadata Initiative (DCMI) specification which is supported natively by a large set of CMIS compliant repository. Since the DCMI specification perfectly maps with our intended ontology as shown in Table 2, we decided to use it as default. In an educational environment, other ontologies may be considered more appropriate, e.g. the IEEE LOM [54]. However, general purpose repositories, such as the ones we are targeting, do not natively support these specification and the implementation efforts needed in order to be compliant are significant, since they should be performed on each of the repositories connected.

Table 2. FAIR ontology mapping onto DCMI.

FAIR Term	**DCMI Term**
Age of students	Audience
Author	Creator
Referee	Publisher
School grade	Education Level
Summary	Abstract
Topic	Subject

In practice, in Alfresco it is possible to create a new content type by creating XML files that define the metadata information related to each new content type that Alfresco should be able to manage. In our case, firstly we defined the new content type *fr:learningobject* which inherits all the properties from the *cm:document* parent. Secondly, we applied Alfresco's DCMI aspect to the newly created type to guarantee adherence to Dublin Core (DC). Alfresco *aspects* allow adding a specific functionality to the existing content types. Since the DC *aspect* has already been made available by the Alfresco team, no custom modifications are required to make it work with our new content type. A similar approach can be seen in Sharepoint by means of using *DC Columns*, which extend the metadata definitions of custom content types. Other repositories, such as Nuxeo, support natively the Dublin Core metadata, so they are ready to be connected with our platform without further efforts.

The meaning of each field is obvious from its name, except maybe for the school grade which expresses for which level of instruction the objects are suitable, i.e., primary schools, high schools, university, etc. This is specified at upload time by the teacher who uploads the content. For existing objects, this is set depending on the type of institution that runs the repository.

Querying When a user searches content using the web interface, the query is performed on all the databases connected with the repository. This is transparent for the user since the web application automatically formulates and translates the queries in the CMIS query language, then sends them to the CMIS interfaces of all repositories. Only when a LO is requested for download its CMIS identifier is resolved and the content is fetched from the repository system. When approaching the design of the distributed architecture is important to carefully choose a search model.

Two are the main possibilities [55]:

- Indexed search: This model implies the maintenance of a single centralized index which maps each entry to a corresponding position in the repository.
- Federated search: This model requires that each time a query is formulated, this has to be duplicated towards all databases of the repositories. No centralized index is needed but the response time is slower than in the previous case.

We chose the federated model because it fits better with a heterogeneous environment and the performance drawbacks were not so significant. In our specific case queries are duplicated towards the CMIS interface of all repositories through the authenticated connection established by the web application. Handling different distributed repositories in this way introduces a possible issue related to response latency. In our implementation, the handling of the parallel queries is done using a pool of threads. As soon as the threads are created a timer is set and, if a thread has not been joined before a given threshold time, is then terminated. In this way a user is guaranteed to receive a reply by the working repositories in a predetermined amount of time. The threshold value has to be tuned by considering the existing latencies between each of the repositories involved and the server where the web application resides. When all the results are collected from all the repositories, their ordering is performed locally by a module of the web application, then it is possible to pass the data to the visualization module. Since the results are collected asynchronously, the reordering process is somehow difficult. In our first prototype we decided to wait until all the threads have been joined and then perform the reordering. If the user expresses no particular filtering preferences, we apply an alphabetical reordering based on the learning object name. Another possibility could be to use the content ranking as an ordering key. This rank could be a measure of quality decided by the reviewers during the peer-review phase.

However, this approach implies the definition of a new term in the ontology which is not part of the Dublin Core ones. Other algorithms for reordering are also under investigation. By means of using some Javascript libraries, the rendered tables are sortable by the users using any of the table headers as key. A textual filter can also be applied in order to further refine the research. A sample web search result is shown in Fig. 3.

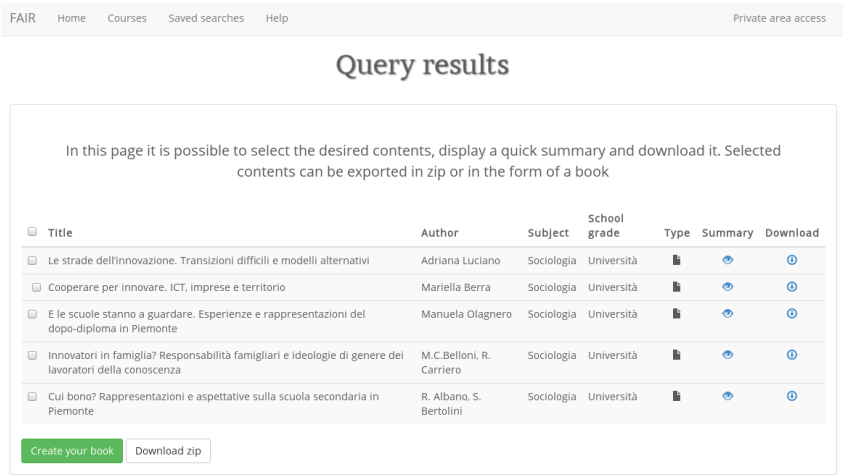

Fig. 3. Screenshot of the results of a sample query.

User Management Since the system must allow the creation and administration of a potentially large number of users, the definition of user roles is fundamental.

Apart from the administrator, who has complete control over the configuration of all tools, the other main roles are:

- staff member;
- referee;
- registered user;
- anonymous user.

A different set of permissions is granted to each category of users, according to Table 3. Apart from the anonymous user, all other roles require a registration to the web application. For example, an anonymous user can only search for materials and perform basic operations. Registered users, once logged in, can access a personal dashboard that allows to review the past taken actions, the saved files, the comments and the sharing state of their CLOs. In addition, they can insert new materials in the repository and the corresponding metadata information.

Table 3. User permissions.

User	Query	Save/Upload	Review	Delete
Anonymous	✓			
Registered	✓	✓		
Referee	✓	✓	✓	
Staff	✓	✓	✓	✓

The role of the referee is to approve any of the uploaded material, which is by default not visible to other users, for publication. Referees are typically voluntary teachers (i.e., a set of registered users) that offer their expertise to check the uploaded

LOs by other users. Typical evaluation criteria include the appropriateness of the content for the declared school grade, proper attribution of sources and absence of mistakes. Moreover, they are also checked to ensure they can be played without any visualization issue. This should ensure that the quality of the LOs in the repositories stays at an acceptable level. Staff users have complete control over the materials, including the possibility to delete objects from the repository.

Repositories Integration using CMIS CMIS is a standard for interoperability between content management systems. It allows to connect together repositories of different nature without using a custom-written proxy to translate each call in different proprietary languages. Using its API it is possible to achieve a high level of flexibility, writing code just once for all the CMIS compliant repositories.

The standard handles the contents by means of two abstract entities: folders and documents. This is perfectly aligned with the design of our FAIR architecture since we are building a hierarchical structure of folders and objects. Using the API, through the CMIS service it is possible to interact with the connected repository and perform the main actions required by FAIR.

The CMIS standard is supported by a growing number of repository vendors. Alfresco and Nuxeo are probably the best supported solutions on the market but also the latest versions of Sharepoint are, for instance, compatible with the standard. Since Drupal is mainly written using PHP, in the development of the connector module the CMIS PHP API was used. In comparison to other more complete connectors such as the Java one, this API does not fully map the Alfresco features into the CMIS API yet but the community is working towards their completion.

CMIS requires to perform an initial authentication as a user of the repository to be able to perform actions. In our prototype configuration all the users of the web application are users for the Drupal system. All the users of the platform are mapped to a single Alfresco user by our Drupal system, i.e., from the repository side a single user is performing all the actions. This is also the most convenient choice for the repository administrators since they just need to create a user and grant access to it. However, this solution also hides the actual users performing the actions from the point of view of the repository administrators. For this reason, it may not be optimal for network administrators and security managers which potentially need to review the Drupal logs to determine the actual users responsible for given actions. To provide more choices about this aspect, a direct mapping between Drupal users and the users of each repository is under investigation for the next version of the prototype.

3.3 Example Usage

This subsection provides some examples of the activities that each of the main actors involved (teachers and learners) can perform using the web application. They are summarized in Fig. 4.

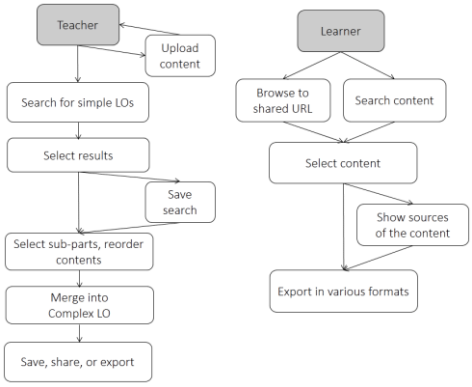

Fig. 4. Example of tasks performed by teachers and learners.

Teachers An activity that a teacher typically does, apart from uploading, is searching. This can be conducted on the metadata inserted by other users during the upload phase, and advanced querying and filtering is possible. Results are presented in the form of tables, with short previews of the content itself. Using the same visualization technique as the major search engines it is easy to spot and select the most appropriate results.

Selecting some content allows to export them in a single archive (e.g., compressed with ZIP), or some editing activities can be carried out. In the first case, the user will find the selected material as well as other supporting files to allow easy usage and presentation of the material. In particular, two HTML5 slideshows are included in the archive. One simply contains the metainformation about the content itself. The other allows to display the content easily on an interactive whiteboard environment. In particular, few, big icons are displayed to ease the selection also with imprecise pointing devices (e.g., interactive whiteboards). The advantage of using HTML5 is that the whole presentation can be played inside the browser, with clear advantages in terms of portability.

Once the content is selected, the teacher can edit it in several ways. The editing page offers the possibility to reorder the contents, to select just some parts of it (i.e., only some pages of a text document) and to export the whole selection as a single PDF file. This is very convenient for teachers preparing content by splicing together several text documents since it allows to share a CLO instead of a collection of links to sparse contents. The final CLO can be automatically created on-the-fly and be exported.

During the upload phase, an authenticated user with upload permissions (which is the typical role of the teacher) can populate the repository with new objects. Metadata are entered by means of forms, one of which is specifically aimed at inserting tags. Those can be inserted by the authors or the referees after reviewing the uploaded content. When a teacher creates a complex LO comments can also be added, so that other teachers who would like to use the content itself can have additional side

information about potential usages. This again leads to a new set of metadata which can be useful in the search process for other educators.

Finally, note that some teachers can also act as referees. Fig. 5 shows a screenshot of the advanced actions available to them. In addition to all the possibilities already described, referees have the option to approve the content, uploaded by other registered user, for publication.

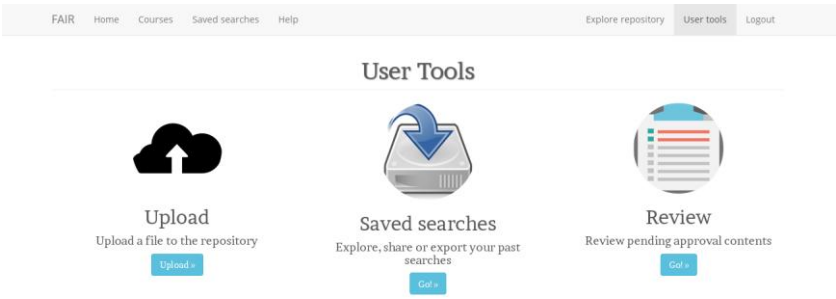

Fig. 5. Screenshot of the advanced actions available to a user in the referee category.

Learners They typically use the web application for two reasons: they received a URL from a teacher pointing to some LOs prepared in advance for them or they want to look for new educational material. In the first case the URL shows not only the final CLO as edited by the author but also it provides pointers to all the sources used in the CLO. In this way, the learners can decide whether to download the CLO, revise all the sources, or download only parts of the CLO. This possibility increases the amount of flexibility of the web platform, i.e., the content can be tailored to the specific needs of each learner, e.g., depending on the device used for that specific access. The learner can also search for new material exactly as the teacher does, without the need of logging into the web application.

4 Experimental Evaluation

This section aims to evaluate the proposed architecture from the point of view of the user experience by means of quantitative and qualitative results. First, objective performance parameters will be presented, in particular the response time of the CMIS queries done through the web interface. This is to ensure that the designed system does not incur in performance issues due to the need to handle LOs that are inherently distributed. Then, in order to assess the perceived user experience, e.g., the simplicity of the user interface as mentioned in the objectives, we asked the users' opinions through a questionnaire. The evaluation relies on a prototype which has been implemented and is currently tested at our technical university in a live environment.

We are currently using a live installation of the prototype available to all users involved in the experimentation.

Table 4 reports summary information about such an installation. Two repositories are involved, i.e., the Alfresco repository that the university set up for this experimentation and a Nuxeo repository used for tests that will soon be replaced by the Nuxeo repository of the other university in our city. Note that the total values for the FAIR platform may be lower than the sum of the individual repositories due to some overlaps. There are 36 registered users, some of them performed and tested several actions, including upload of LOs. Learning objects represent different types of contents, i.e., text documents, images, video lectures, covering 18 subjects. The cut and merge feature has been tested by the users producing 225 CLOs that are currently stored in the web application, drawing from the existing LOs in the two repositories.

Table 4. Summary information about the current status of the project.

	Repository-specific information		**Platform information**
	Our University (Alfresco)	**Other University (Nuxeo)**	**FAIR**
Learning Objects	1127	221	1348
School grades	6	3	6
Subjects	15	6	18
Authors	77	25	100
Uploaders	-		31
CLOs	-		225

4.1 Quantitative Evaluation

Setup of the Distributed Test System In order to evaluate the performance of the platform, a set of tests have been run on a distributed system of repositories to simulate a real use case in heavy load conditions. All the tests measure the response time of FAIR when performing both typical and demanding operations that users might do while interacting with the web application. We focus on latency since it is known to be an important parameter that can heavily influence the user experience [56]. Ten repositories have been included in our test system: 5 instances of Alfresco CE and 5 of Nuxeo.

To quickly create a large and geographically distributed test system we relied on the Amazon AWS infrastructure which offers the possibility of selecting different data centers distributed worldwide. Since the systems in use require a minimum of 4 GB of RAM, we selected the *m3.medium* type of virtual machine (VM). Five data centers have been used for the test. In each of them, we instantiated both an Alfresco and a Nuxeo VM, for a total of 10. Two instances of *m3.large* VM have also been used to check if the availability of a higher quantity of RAM (i.e., 8 GB) could somehow improve the final performance. Since no particular improvements could be observed in the querying performance, *m3.medium* machines appear an appropriate choice for the sake of the tests. Each repository has been loaded in advance with a

different amount of Learning Objects, ranging from 1,000 to 10,000 to emulate a heterogeneous scenario in terms of repository sizes. Having large repositories in the test is important since their size might influence the response time. Table 5 summarizes the characteristics of the repositories in the setup.

Table 5. Characteristics of the repositories included in the test system. Virtual machines (VM) have been provisioned using the Amazon AWS Virtual Servers in the Cloud infrastructure.

Software (version)	Location	Server type	RAM	LOs
Alfresco CE (2016)	Ireland	m3.large	8GB	10,000
Alfresco CE (2016)	Frankfurt	m3.medium	4GB	1,000
AlfrescoCE (2016)	N. Virginia	m3.medium	4GB	1,000
Alfresco CE (2016)	California	m3.medium	4GB	1,000
Alfresco CE (2016)	Sydney	m3.medium	4GB	2,000
Nuxeo(8.2)	Ireland	m3.large	8GB	10,000
Nuxeo(8.2)	Frankfurt	m3.medium	4GB	1,000
Nuxeo(8.2)	N. Virginia	m3.medium	4GB	1,000
Nuxeo(8.2)	California	m3.medium	4GB	1,000
Nuxeo(8.2)	Sydney	m3.medium	4GB	2,000

Web Application Performance The first set of tests had been designed to evaluate the response time of a query selecting a specific field, i.e., the title associated with the LO. This simulates a real use case scenario where users specify which terms to search by means of input forms within the web application. In order to assess the scalability of the system, we first run a set of queries connecting just one repository and then we repeated the same ones connecting all the 10 available repositories. To avoid being overloaded by results when repositories have many objects we limited the maximum number of results to 50 items for each repository. This is consistent with the typical experience of a web page which only shows, on the first page, a given maximum number of results. Note that the CMIS API allows to easily specify such limit. Results are shown in Table 6. Times are measured using the development tools available in the Mozilla Firefox browser. Values show the total time required by the web application to query the database, handle the returned objects and finally process and display a table similar to the one shown in Fig. 3. For convenience, the last column shows the time difference between one repository chosen as reference and the ten-repository cases. As expected, the ten-repository case exhibits higher response times but the maximum response time increase is about 2.5 seconds. Therefore, it appears that the system can scale well, which is a desirable feature as highlighted in Section 1.

Table 6. Time (ms) taken to perform operations involving repository querying. The considered Alfresco repository is located in N. Virginia.

Query term	1 Alfresco		10 Repositories		Time Difference
	Results	Time	Results	Time	
“Introduction”	30	2,758	62	3,082	324

“Jpeg”	4	1,962	122	4,467	2,505
“PDF”	50	3,078	389	3,991	913
“Python”	50	3,015	250	4,755	1,740
“Matematica”	23	2,207	115	3,962	1,755
“doc”	9	1,810	338	3,844	2,034
“Informatica”	5	1,554	55	2,864	1,310
“Elettronica”	16	2,028	166	4,308	2,280
“Biologia”	13	1,780	199	3,817	2,037
“Internet”	12	1,642	138	3,010	1,368
Average	21	2,183	183	3,810	1,627

The previous values can be improved by showing fewer elements in the table and splitting the results in different pages to benefit user friendliness.

Another possibility could be to use a frontend framework capable of dynamically rendering the elements inside the page as soon as they are available. In this case, the fastest replies could be shown in advance so the user waiting times could be lowered. Furthermore, it is possible to slightly reduce the response time by means of some interface optimization tricks, such as aggregation and compression of CSS and Javascript resources, but the improvement is negligible (about 10 ms) with respect to the non-optimized web application.

Other experiments are devoted to test the more demanding operations, i.e., export of LOs after mixing and merging. This requires processing the LOs themselves in addition to querying and retrieving them. Note that some repositories already offer a few of the features provided by the web application. For instance, both Alfresco and Nuxeo, which are well integrated with the Libreoffice suite, offer the possibility to export LOs in other formats, such as PDF. However, such interfaces are different for each system and they cannot be accessed through a standardized channel such as the CMIS API. Therefore, the web application does the processing itself, by first retrieving the LOs from the remote repositories, then elaborating them locally to perform the requested operation. Table 7 shows results for a set of typical merge operations. Since users typically expect that the merge operation takes some time especially when it produces a large file, the values in Table 7 appears more than adequate for a good quality of experience. Moreover, note that even with a very heavy task such as merging 20 different resources to produce the equivalent of a large size electronic book of 569 pages, the time is still limited to less than one minute.

Table 7. Time required to perform the given merge operation.

Number of files	Resulting pages	Time (ms)
5 (5 random repos)	21	9,544
10 (1 for each repo)	97	11,753
20 (2 for each repo)	569	54,572

The last set of results addresses the case in which the user requests the creation of a single archive (e.g., a ZIP file) of a number of selected LOs so that they can be easily

exported. They also include the time needed for the creation of an HTML5 file included in the ZIP to simplify the presentation of the LOs. This was a suggestion by the early users of the system that wanted to avoid navigating the folder tree in the ZIP file especially when the content was meant to be immediately presented on screen. The HTML5 file included in the ZIP can, in fact, run on any HTML5 compliant browser. Results are shown in Table 8.

Table 8. Time required to perform the test.

Number of files	Total size (MB)	Time (ms)
10 (1 for each repo)	17	15,430
20 (2 for each repo)	38.5	26,588
40 (4 for each repo)	104	62,341

From a technological perspective, a background thread starts as soon as the user selects the execute button. The user can continue the navigation and, as soon as the file will be ready for download, a notification appears. This helps improving user friendliness and avoiding long waiting times. In order to evaluate these results, we rely on [56], which shows that the quality of experience for a download task is considered acceptable, i.e., graded from 3 to 4 (out of 5), if the download takes between about 20 and 60 seconds for a 10 MB file.

Therefore, in all cases the quality of experience is good. Hence, we conclude that the optimized performance objective stated in Section 1 has been achieved.

Actual Usage Cases A logging tool has been used in the prototype to investigate the activity of the users through the platform, in order to get an overview of users' behavior. The analysis shows that users, after signing in to the platform, search for content, and most of the time download single LOs directly. Some users create a CLO from the content found by the searches. The CLO is, in a few cases, saved in the personal dashboard.

Many users then take advantage of the possibility of creating live streaming sessions in which the content just searched or created is typically imported. Note that about 80% of the times the user also activates session recording. Such a high percentage seems to indicate that many of the registered users of the platform are educators that need to search for material to use in their online lectures. Some of them are recoding the session for later usage, probably because it can be easily exported to MOOC platforms or be played, either online or offline, after download. Even in such a rather complex scenario which requires the integration between the streaming platform and ours, we did not experience any particular technical issue apart from the need, in the initial phase, for minor bug fixes.

Unregistered users, which are typically learners (since educators often use a login, needed to save work and initiate streaming sessions), almost always land directly on the home page. Such a behavior suggests that the URL creation tool, available in the platform, to directly point to content is not used much. We also noted that unregistered users typically start searching and exploring the LOs, and in some cases, they download entire sets of materials which have been pre-packaged by staff

members. This fact highlights that platforms offering a significant set of blocks of already prepared and revised contents seems to be very appealing for users.

4.2 Qualitative Evaluation by Users

In the context of platform and content evaluation for future developments we asked a set of users about their opinion through a questionnaire about their experience with FAIR. The set of respondents included both registered and unregistered users for a total of 19 persons, balanced in gender and age (ranging from 26 to 65). The large majority of them has a master in some technical field and are reasonably familiar with technology. Only three have a social sciences background.

The questions proposed to the users have been prepared in relation to the objectives of this work stated in Section 1. Questions included the evaluation (on a 1 to 5 scale) of the easiness of navigating the website, the satisfaction in the organization of the web interface functionalities, the perceived response time of the web interface and easiness of using the interface in general. Open questions included the possibility to describe any issues experienced during navigation, the most interesting functionalities, their experience with different devices and free comments. Moreover, we asked all the users if they expect to reuse the website in the future.

Easiness of navigation scored about 4.1 out of 5, organization of the web interface was about 3.9, perceived response time was about 3.6, and easiness in using the interface was about 4.2. The open questions did not highlight any particular difficulty in navigation, whereas one of the most interesting functionalities indicated by the users is the possibility, after merge operations, to automatically generate a PDF file with all the content. In the free comments, some users particularly appreciated the possibility to download an HTML5-based presentation that can run entirely inside the browser. On the downside, we still need to work on the content adaptation functionality since it does not seem to work correctly for some mobile users due to issues about supported video formats. All but two users expect to reuse the site sometimes in the future. Therefore, we believe that the objectives stated in Section 1 have been achieved and the perceived quality of experience of the users is good.

5 Conclusion

This work presented FAIR, an architecture to integrate LO repositories by means of a unified and transparent access through a web-based interface. FAIR is licensed as open source software, and it relies only on other open source software for installation and operation. It is a web-based application with a particularly simple interface, yet it allows to perform even complex operations such as cut, paste, merge of different LOs entirely online without the need to download any LO. Newly created LOs (i.e., CLO) can then be shared or downloaded. In the latter case, they can be archived in a way which allows to easily play them simply through a recent HTML5 browser. Results show that the performance of the application is good when measured both objectively through response times and subjectively through user experience. Future work will be

devoted to test the application in more complex scenarios. In particular, we are in the process of concluding an agreement for a locally-funded project about improving high-school teaching in which nearly all high schools in our local area will participate. This would allow to significantly enlarge the user base and to perform more in-depth tests so that, in case the system will be selected by the Ministry of Education as one of the platforms at the national level for e-learning, the architecture will be very well tested.

6 References

1. P. Brusilovsky et al., "Increasing adoption of smart learning content for computer science education," in Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference. ACM, 2014, pp. 31–57.
2. "Alfresco project." (2017, Jan 24) [Online]. Available: https://www.alfresco.com
3. "Nuxeo platform." (2017, Jan 24) [Online]. Available: http://www.nuxeo.com
4. M. Burns and G. P. Schell, "Merlot: a repository of e-learning objects for higher education," E-service Journal, vol. 1, no. 2, pp. 53–64, 2002.
5. H. Abelson, "The creation of opencourseware at MIT," Journal of Science Education and Technology, vol. 17, no. 2, pp. 164–174, 2008.
6. E. N. Forte, M. H. Wentland Forte, and E. Duval, "The ARIADNE project (part 1): knowledge pools for computer-based and telematics-supported classical, open and distance education," European Journal of Engineering Education, vol. 22, no. 1, pp. 61–74, 1997.
7. M. Gaebel, V. Kupriyanova, R. Morais, and E. Colucci, "E-learning in European education institutions," European University Associations (EUA) Publications, 2014.
8. A. Bucciero, L. Mainetti, and R. Vergallo, "Structuring repositories of educational experiences: A case study," in IEEE Global Engineering Education Conference (EDUCON), April 2012, pp. 1–9.
9. "GLOBE: Global learning objects brokering exchange." (2017, Jan 24) [Online]. Available: http://globe-info.org
10. S. Ternier, E. Duval, D. Massart, A. Campi, S. Guinea, and S. Ceri, "Interoperability for searching learning object repositories: the ProLearn query language," D-Lib Magazine, vol. 14, no. 1, 2008.
11. M. Hatala, G. Richards, T. Eap, and A. Shah, "Secure communication layer for scalable networks of learning object repositories," in E-Learning Networked Environments and Architectures. Springer, 2007, pp. 276–305.
12. B. Simon, D. Massart, F. Van Assche, S. Ternier, E. Duval, S. Brantner, D. Olmedilla, and Z. Miklos, "A simple query interface for interoperable learning repositories," in Proceedings of the 1st Workshop on Interoperability of Web-based Educational Systems, 2005, pp. 11–18.
13. D. Boehringer and H. Bernlohr, "CampusConnect: An open-source initiative to connect learning management systems," in IEEE Global Engineering Education Conference (EDUCON), 2014, pp. 134– 141.
14. G. Kortemeyer et al., "The LearningOnline network with computer-assisted personalized approach (LONCAPA)." (2017, Jan 24) [Online]. Available: http://www.lon-capa.org/papers/cyprusloncapa.pdf
15. D. Choy, R. McVeigh, and F. M¨ uller, "Content management interoperability services CMIS version 1.0," OASIS Standard, 2010.

16. H. Xu, “Factors affecting faculty use of learning object repositories: an exploratory study of Orange Grove and Wisc-Online,” Ph.D. dissertation, University of North Texas, 2011.
17. “Creative Commons BY-SA 3.0 IT license,” (2017, Jan 24) [Online] http://www.creativecommons.org/licenses/by-sa/3.0/it/deed.it
18. W. Schweibenz and J. Sieglerschmidt, “Curating digital heritage by MusIS - the south-western German museum network,” in Proc. Annual Conference of CIDOC, Athens, Greece, 2008.
19. “IEEE learning technology standards committee.” (2017, Jan 24) [Online]. Available: https://www.computer.org/web/standards/learning-technology
20. “Centre for excellence for teaching and learning in reusable learning objects (RLO-CETL).” (2017, Jan 24) [Online]. Available: http://web.archive.org/web/20160304101716/http://www.rlo-cetl.ac.uk/
21. L. Campbell, “Engaging with the learning object economy (ch.4),” in Reusing online resources: A sustainable approach to e-learning. Routledge, 2003, pp. 35–45.
22. D. A. Wiley, “Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy,” in The instructional use of learning objects, D.A. Wiley, Ed. Agency for Instructional Technology, 2002.
23. O. Bohl, J. Scheuhase, R. Sengler, and U. Winand, “The sharable content object reference model (SCORM) — a critical review,” in Proc. Intl. Conf. on Computers in Education, Auckland, New Zealand, Dec. 2002, pp. 950–951.
24. “CMI012 — AICC Packaging Specification (Version 1.0),” Sep. 2006. [Online]. Available: https://github.com/ADL-AICC/AICC-Document-Archive/releases/tag/cmi012v1
25. “IMS Content Packaging Specification Primer, Version 1.2 Public Draft v2.0,” Mar. 2007. [Online]. Available: http://www.imsglobal.org/content/packaging/#v1p2
26. P. R. Polsani, “Use and abuse of reusable learning objects,” Journal of Digital Information, vol. 3, no. 4, 2003.
27. N. Friesen, “Interoperability and learning objects: An overview of e-learning standardization,” Interdisciplinary Journal of E-Learning and Learning Objects, vol. 1, no. 1, pp. 23–31, Jan. 2005.
28. “Moodle project.” (2017, Jan 24) [Online]. Available: https://moodle.org
29. “Edmodo project.” (2017, Jan 24) [Online]. Available: https://www.edmodo.com
30. J. Hylen, “Open educational resources: Opportunities and challenges,” Proceedings of Open Education, pp. 49–63, 2006.
31. B. Wenk, “Open educational resources (OER) inspire teaching and learning,” in IEEE Global Engineering Education Conference (EDUCON), Madrid, Spain, Apr. 2010, pp. 435–442.
32. H. C. Davis, L. Carr, J. Hey, Y. Howard, D. Millard, D. Morris, and S. White, “Bootstrapping a culture of sharing to facilitate open educational resources,” IEEE Transactions on Learning Technologies, vol. 3, no. 2, pp. 96–109, 2010.
33. “Registry of open access repositories.” (2017, Jan 24) [Online]. Available: http://roar.eprints.org/view/year/
34. R. De Santiago and A. L. A. Raabe, “Architecture for learning objects sharing among learning institutions — LOP2P,” IEEE Transactions on Learning Technologies, vol. 3, no. 2, pp. 91–95, 2010.
35. J. Klerkx, B. Vandeputte, G. Parra, J. L. Santos, F. Van Assche, and E. Duval, “How to share and reuse learning resources: the ARIADNE experience,” in Sustaining TEL: From innovation to learning and practice. Springer, 2010, pp. 183–196.

36. M. Hatala and G. Richards, “Pool, Pond and Splash: A Canadian infrastructure for learning object repositories,” in Proc. of the 5th IASTED Int. Conference on Computers and Advanced Technology in Education (CATE), 2002, pp. 54–59.
37. “Connecting and extending peer-to-peer networks - LionShare white paper,” Oct. 2004. [Online]. Available: https://web.archive.org/web/20060111005319/http://lionshare.its.psu.edu/main/info/docspresentation/LionShareWP.pdf
38. “The Orange Grove.” (2017, Jan 24) [Online]. Available: http://www.theorangegrove.org/
39. Fox Valley Technical College, “Wisc-online.” (2017, Jan 24) [Online]. Available: http://www.wisc-online.com/
40. S. M. Baldiris, J. L. Bacca, A. N. Rojas, J. C. Guevara, and R. Fabregat, “LORSE: Intelligent meta-searcher of learning objects over distributed educational repositories based on intelligent agents,” in Frontiers in Education Conference (FIE), Rapid City, SD, 2011, pp. 1–5.
41. H. V. D. Sompel, M. L. Nelson, C. Lagoze, and S. Warner, “Resource harvesting within the OAI-PMH framework,” D-Lib Magazine, vol. 10, n. 12, 2004.
42. A. Fuertes, M. Casals, N. Forcada, A. Giretti, M. de Grassi, S. Apelt, and M. Eisenhauer, “MACE eContentplus project: Metadata for architectural contents in Europe,” in Proceedings of the IV Simposio Pluridisciplinar sobre Diseño, Evaluacion y Desarrollo de Contenidos Reutilizables. Madrid, 2007.
43. MELT, “Metadata ecology for learning and teaching.” (2017, Jan 24) [Online]. Available: http://info.melt-project.eu
44. ASPECT, “Adopting standards and specifications for educational contents.” (2017, Jan 24) [Online]. Available: http://aspect-project.eun.org/
45. G. Rajendra and M. Bernard, “Quality Assurance for Reusable Learning Objects on a Peer-To-Peer Network,” International Journal of Emerging Technologies in Learning, vol. 11, no.10, pp. 4-10, 2016.
46. Higashino, Masayuki, et al., "Management of streaming multimedia content using mobile agent technology on pure P2P-based distributed e-learning system," International Journal of Grid and Utility Computing, vol. 5, no. 3, pp. 198-204, 2014.
47. H. Masud and X. Huang, "An e-learning system architecture based on cloud computing," International Journal of Computer, Electrical, Automation, Control and Information Engineering, vol. 6, no. 2, pp. 255-259, 2012.
48. S. Nasr and Y. Helmy, “An Enhanced E-Learning Ecosystem Based on an Integration between Cloud Computing and Web 2.0,” in Proc. of the IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Luxor, Egypt, Dec. 2010, pp. 48-55.
49. E. Pecheanu, C. Segal, and L. Dumitriu, “On modeling adaptive Web-based instructional systems,” in IEEE Global Engineering Education Conference (EDUCON), Amman, Jordan, Apr. 2011, pp. 1146–1151.
50. J. Marciniak, “Building E-Learning Content Repositories to Support Content Reusability,” International Journal of Emerging Technologies in Learning, vol. 9, no. 3, pp. 70-76, 2014.
51. J. Schreurs, R. Dalle, and G. Sammour. “Authoring systems delivering reusable learning objects,” International Journal of Emerging Technologies in Learning, vol. 4, no. 2, pp 27-31, 2009.
52. E. Di Nitto, L. Mainetti, M. Monga, L. Sbattella, and R. Tedesco, “Supporting interoperability and reusability of learning objects: The virtual campus approach,” Journal of Educational Technology & Society, vol. 9, no. 2, pp. 33–50, 2006.

53. “BigBlueButton Open Source Web Conferencing,” (2017, Jan 24) [Online]. Available: http://bigbluebutton.org
54. IEEE LOMWG12 - Working Group for Learning Object Metadata, “Standard for learning object metadata,” IEEE Std. 1484.12.1, 2002.
55. H. Stehouwer, M. Durco, E. Auer, and D. Broeder, “Federated search: Towards a common search infrastructure,” in Proc. of 8th International Conference on Language Resources and Evaluation (LREC), Istanbul, Turkey, May 2012, pp. 3255–3259.
56. S. Egger, T. Hossfeld, R. Schatz, and M. Fiedler, “Waiting times in quality of experience for web based services,” in 4th Intl. Workshop on Quality of Multimedia Experience (QoMEX), Yarra Valley, Australia, July 2012, pp. 86–96.

7 Authors

Leonardo Favario received the M.Sc. in Computer Engineering from Politecnico di Torino in 2014. He currently is a Ph.D. student in Information and System Engineering at Politecnico di Torino, Italy with the Internet Media Group. His primary research interests include e-learning systems and technologies, with particular attention to the use of multimedia in such contexts.

Enrico Masala received the Ph.D. degree in computer engineering from the Politecnico di Torino, Turin, Italy, in 2004. Since 2011 he is assistant professor in the Control and Computer Engineering Department at the Politecnico di Torino. His main research interests include new communication technologies and their optimization, with special attention to multimedia and wireless.