

Integration of SHERPA/RoMEO in institutional digital repositories to address the uncertainty of copyright

A. Repanovici* and I. M. Barsan

Faculty of Product Design and Environment

Transilvania University Brasov, ROMANIA

e-mail: *arepanovici@gmail.com (corresponding author)

ABSTRACT

The article describes the research conducted to identify the barriers to dissemination and communication of scientific articles made available by institutional digital repositories through self-archiving. Transilvania University of Brasov Romania provides a new service attached to its digital repository, in the form of an automatic query interface of the SHERPA/RoMEO publisher reviewing platform to support the academic community. The query principles of the platform SHERPA/RoMEO's database of publisher policies is integrated with DSpace, a digital repository software. While defining a new submission for a collection on the DSpace platform, a query is sent to the SHERPA/RoMEO server, using the application programming interface it provides, which responds by an XML document containing the information required. The XML document is further processed through the XSLT transformation in order to generate the content displayed on the HTML page. Authors find out via a simple query, whether they are allowed to self-archive an article, whether the publisher allows it, and under what conditions. Developing a software application that integrates the copyright conditions imposed by publishers emerges as the solution for resolving authors' scepticism towards self-archiving published articles. The software application, which has been developed with low cost, would be very useful to other universities that face obstacles in populating institutional digital repositories due to copyright concerns.

Keywords: Digital repositories; Copyrights concerns; Self-archiving; Open Access; XML; XSL; DSpace; SHERPA/RoMEO.

INTRODUCTION

The movement for open access to scientific information has been actively promoted in recent years, being supported by more and more members of the scholarly community. There are significant economic, social and educational benefits of making available the scholarly research output, of providing accessibility to these resources without financial, legal and technical barriers. Houghton et al. (2009) discussed the following costs and benefits associated with open access as an alternative publishing model, among others: the costs for open access to scientific literature are much lower, therefore universities and research institutions can save financial resources; open access may have substantial net benefits in the longer term in comparison with traditional models; the impact of open access articles is higher than that of articles in journals by subscription; and, open access electronic journals and self-archiving are more cost-effective from an economic

perspective. The costs and benefits of open access publishing through digital institutional repositories are being debated by governments, financing bodies, publishers and universities worldwide. The proliferation of these repositories worldwide indicates that the digital services are attractive and compelling to institutions (Foster and Gibbons 2005) and for the repositories to be successful, previous studies have shown that they must be populated with scholarly works of enduring value (Foster and Gibbons 2005; Swan 2005; Kim 2011; Singeh, Abrizah and Karim 2013).

One of the barriers to open access publishing and populating institutional repositories is the uncertainty of knowing the conditions of copyright. The policies issued by various research institutions, as part of the grant award process, are offered for consultation by a specialised register maintained by the University of Nottingham in the United Kingdom i.e. SHERPA/JULIET (<http://www.sherpa.ac.uk/juliet/>), the research funders' open access policies. These policies are structured based on three groups: (a) Open Access Archiving, that requires open and free access to the author's published article or the reviewed version of the article (post-print), although the temporary embargo of the publishing houses minimises the efficiency of access to such articles; (b) Open Access Publishing, that requires publishing in open access journals or in hybrid journals in order to render more efficient process of disseminating the scientific research results; and (c) Data Archiving Policy, that requires archiving primary data in a certain period of time. In mid-2013 (as of September 12th, 2013), SHERPA/JULIET recorded 126 policies of funding agencies from the United Kingdom (63), Canada (13), USA (9), Denmark (5), Ireland (5), Sweden (5), and other countries (26). The Registry of Open Access Repositories Mandatory Archiving Policies (ROARMAP, <http://roarmap.eprints.org/>) recorded 81 approved policies and 12 planned policies of funding agencies.

Within the 2005 *JISC/Key Perspectives Survey*, a total of 1296 researchers from various countries and areas of science answered the question of whether they would fulfil the obligation of employers and funders with respect to self-archiving of scientific publications (Swan 2005). The research found that 15% of the authors of scientific publications self-archive the publications; 95% of researchers confirmed that they would self-archive the publications thereof if they were required by funders or employers; 81% would self-archive spontaneously; 13% would do so with reluctance; and 5% would not fulfil such request. These findings were confirmed by the objective tests of organisations that approved the self-archiving policies, where the self-archiving pace tends towards 100% (Harnad 2006).

Within the RoMEO project (Rights Metadata for Open Archiving, <http://www.sherpa.ac.uk/romeoinfo.html>), maintained by SHERPA at the University of Nottingham, four publishers archiving policies have been proposed (each of these policies is appointed by a colour): (a) the green policy allows self-archiving of pre-print and post-print or publisher's version/PDF; (b) the blue policy allows self-archiving of post-prints or publisher's version/PDF; (c) the yellow policy allows self-archiving of pre-print; and (d) the white policy indicates that archiving is not fully supported.

This paper describes the integration of a database of publisher policies on open access publishing and archiving of scholarly research output. One barrier to self-archiving of published papers is authors' lack of knowledge of publisher policies and copyright issues regarding the eligibility for self-archiving of articles. Developing a software application that integrates the copyright conditions imposed by publishers emerges as the solution for resolving authors' scepticism towards self-archiving published articles. The system resolves this issue in the context of a university's institutional repository in Romania.

LITERATURE REVIEW

The movement of open access in scholarly communication has developed new models of communication and dissemination of scientific information. Universities and research institutions worldwide are providing the scholarly community with instruments to promote research output through the implementation of institutional digital repositories. Repositories have existed in most institutions and universities since the onset of digital libraries. Krishnamurthy (2008) defines open access archives or repositories as digital collections of research articles that have been placed there by their authors. This self-archiving can be done before or after publication. Krishnamurthy identifies the following eight types of open access within the context of academic institutions (p.50):

- e-print (authors self-archive);
- unqualified (immediate and full open access publication of a journal);
- dual mode (both print subscription and open access version of a journal are offered);
- delayed open access (open access is available after a certain period of time);
- author fee (authors pay a fee to support open access);
- partial open access (some articles from a journal are available through open access);
- abstract (open access limited to table of contents and/or abstracts); and
- co-operative approach (institutional members support open access journals).

The scientific community acknowledges the importance of scientific research visibility and impact by means of scientific production archived and made available in open access digital repositories. The importance of digital repositories is underlined by Ferro and Silvello (2013) who suggested a formal model for the digital archives, as well as by Kim (2011) who remarked that university professors contribute to institutional repositories to make their materials widely accessible in keeping with the benefits of open access. However, universities' commitment to institutional repositories depends on building trust with faculty and solving copyright concerns. Digital preservation and copyright management in institutional repositories should be strengthened to increase faculty participation (Abrizah 2009). However, most authors are afraid of breaching copyright and do not want to devote time to review the publishing conditions imposed by publishers (Bîrsan et al. 2014).

A few surveys have been conducted on the academic community's behavior concerning the open access movement and the motivations and impediments in self-archiving their articles. Palmer, Dill and Christie (2009) examined the attitude of academic librarians to open access in a national survey in the USA. The study found that librarians supported open access and were of the opinion that their main contribution was to educate academic staff and advocate for open access. However, Palmer, Dill and Christie (2009) found that "agreement with various open access-related concepts does not constitute actual action" (p.328) as research have shown that the academic community, especially the authors, encounter barriers that impede their participation in self-archiving practices. Copyright remains the biggest obstacle in self-archiving articles in institutional digital repositories. (Gadd, Oppenheim and Proberts 2003a, 2003b; Chan 2004; Allen 2005; Foster and Gibbons 2005; Abrizah 2009; Stanton and Liew 2012). Issues relating to copyright and intellectual property are also generated by not knowing the conditions within the publishing agreements. There is misconception that self-archiving breaches copyright agreements

(Harnad 2006). Authors fear that, by the publishing agreements entered into, they are not allowed to upload their papers in institutional digital repositories. This ignorance leads to a tendency for authors to be over-cautious (Sale 2006; Abrizah 2009). Among other barriers frequently cited are additional time and effort (Van House 2003), and mistrust (Crow 2002). Singeh, Abrizah and Karim (2013) concluded that the major barrier inhibiting Malaysian authors to self-archive in open access repositories is fear of plagiarism.

However studies have shown that the academic community with technical skills and younger faculty are more involved in self-archiving articles. Kim (2010) who studied the motivations of faculty self-archiving in institutional repositories wrote that providing logistical and technical support will also foster participation of those who are less computer adepts. Repositories have to develop a more productive Web 2.0 outlook in order to converge with an interactive learning model, which will help to achieve “a decolonized transcultural learning zone in higher education” as remarked by Eijkman (2009, p. 246). Repositories with Web 2.0 features will complement the existing interactive environment of repositories that will help in bringing the interactive innovations in the scholarly world. Institutionalizing the Web 2.0 tools in repositories will also help in the microscopic analysis of user behavior. It will surely enable a new method of communication between colleagues, encouraging both personal and professional sharing (DiMicco et al. 2008). Shaffi, Gul and Shah (2013) who explore the occurrence of Web 2.0 tools used in the open repositories emphasized that the development of digital repositories be based on Web 2.0 interactive applications. Given that open access to information is in continuing development, given that scientific research should be practically made visible according to European SHERPA (Securing a Hybrid Environment for Research Preservation and Access) guidelines, the academic community is being granted this tool, which enables the archiving of scientific outcome.

In 2009, at Transilvania University of Brasov, Romania, a quantitative marketing research was conducted to identify the attitudes and expectations of the academic community of Transilvania University on the development of an institutional digital repository with the scientific production of the university (Repanovici 2011). It was found that the academic community’s lack of awareness with respect to copyright conditions hindered the self-archiving flow of articles by the authors. The authors avoid archiving their articles due to copyright concerns, while editors impose their conditions, which are not known on the date of signing the publishing agreement. A solution to this problem is to set up an integrated platform, which incorporates the conditions of the publishers registered with SHERPA/RoMEO into the institutional repository.

OBJECTIVES AND METHOD

The main objective of the study carried out within the academic community of Transilvania University of Brasov was to outline the expectations with respect to the self-archiving facilities, correlated with the copyright conditions. Once the ideal model approved by the academic community was established, a comprehensive digital institutional repository platform was developed, which supports the community. The ultimate aim of this study is to promote the platform among the academic staff of the university. Table 1 details the fundamental issue at hand, and the research questions and research objectives that follow.

To address the fundamental issues at hand, data collection was conducted from February to March 2009, via an online questionnaire, accessed from the research platform of the

Faculty of Economic Sciences, Transilvania University of Brasov. The survey population was 880 members of the university teaching staff. A total of 278 survey responses were obtained. The assumptions made in this study were:

- a) The information that authors have on copyright and self-archiving are insufficient.
- b) The extent to which authors are interested in archiving an article in the university's digital repository, while automatically checking the copyrights imposed by the editor, is high.
- c) The author's position with respect to the use of the integrated platform with automatic access of the copyright terms is favourable.

Table 1: The Issues at Hand, the Research Questions and the Research Objectives

Fundamental issues	Research Questions	Research Objectives
1. The degree of awareness of the academic community with respect to copyright and archiving in digital repositories	a) What is the extent to which academic staff and postgraduate students are informed on the current practices of copyright?	a) To identify the degree of awareness with respect to the teachers' interest in the current practices on copyright and archiving in digital repositories.
2. The position of academic community (staff, masters students, doctoral students) concerning self-archiving, as well as copyright transfer	<ul style="list-style-type: none"> a) Do academic staff possess the competencies necessary for article self-archiving in the digital repository? b) What are the academic staff attitudes and opinions about the automatic verification of the copyright conditions imposed by the editor if an article is archived in the digital repository? 	<ul style="list-style-type: none"> a) To identify the reasons driving the academic community to store, under open access terms, their research results in an institutional repository. b) To ascertain the academic staff attitude with respect to the automatic verification of the copyright conditions imposed by the editor if an article is archived in the digital repository. c) To ascertain preferences on copyright transfer.

The findings led the researchers to the conclusion that the academic community of Transilvania University of Brasov, although had little information about open access journals and the benefits of publishing using this model, but were receptive of having an institutional repository in place. Regarding the assertion that the institutional digital repository is the basic condition for international scientific research, a high majority 97.1%, (representing 270 respondents), chose the statement “Yes, I believe that providing an institutional digital repository service is a prerequisite for international scientific research”. With respect to the need to create such repository, the most frequent answer is “Yes, I consider it necessary to create an institutional digital repository” – 92.1% (256). A total of 59.7% (166) of the respondents considered open access as the most important in creating a digital repository and 41.0% (114) did not consider open access to be important. When asked about their perception regarding the extent of free access to information that affects scientific research, 31.3% (87) responded to “a great extent”, and 24.8% (69) responded “neither to a great nor to a small extent”. A total of 260 (93.5%) respondents considered that providing a digital repository is a prerequisite for aligning the university with international scientific research.

The institutional digital repository, named Aspectt (<http://aspectt.unitbv.ro>), the first institutional digital repository in Romania, was then implemented, using the DSpace

software platform. A campaign to promote its services was further carried out. As the university implemented Aspekt, the researchers found that the majority of academic staff were not contributing, and they were often reluctant to contribute. The number of submissions were not as expected. Most teaching staff claimed ignorance of intellectual property issues.

Following the issue on copyright which the researchers suspected to be the primary reason for low submissions of articles to the institutional repository, another web-based survey was conducted in 2011 on the knowledge of copyright and intellectual property issues among the academic community at Transilvania University of Brasov. A total of 265 responses were obtained. Issues of copyright confusion surfaced. A total of 73.2% (194) respondents considered that for articles written by authors, authors hold the copyright. A total of 20.0% (53) deemed the publisher as the copyright holder. Nobody considered the university as a copyright holding institution. Similarly, 73.2% (194) of the respondents wanted to archive articles in the digital repository for fear of copyright infringement, while 26.8% (71) were interested but do not have time to review the rules imposed by publishers. Regarding the concern on copyright transfer/assignment as an author-owner, 53.2% (141) indicated assigning no copyright for free, while 33.2% (88) indicated assigning it for free. Another 13.2% (35) indicated author retain a non-exclusive right to his or her work.

Following such findings, the next section describes the results of implementing a new service attached to the digital repository, i.e. an automatic query interface of the SHERPA/RoMEO publisher reviewing platform.

RESULTS

Implementing and Using the DSpace Platform

DSpace (DSpace 2013) is a software application (platform) designed for academic, non-profit and also commercial organizations with a view to developing and managing digital repositories. The digital repository is a collection of digital documents, organized in a well-defined hierarchical structure. DSpace software is free of charge and easy to install (*out of the box*), and fully customizable, in order to suit any organization's needs. DSpace preserves and enables easy and open access to all types of digital content including text, images, moving images (video) and data sets. With a growing community of developers committed to continue the software's expansion and improvement, each application installation benefits from the experience of the previous users and developers.

Hence, DSpace is the software support of a digital repository of documents. In turn, the digital repository is the environment (software) where an institutional digital repository may be created. There are several definitions of the "institutional repository". Lynch (2003) defines the institutional repository as: "a set of services that a university provides to its community members, for the management and dissemination of digital materials created by the institution or by the members of such community" (p.333). Ware (2004) also includes the participation of the open archives initiative (OAI - Open Archives Initiative) in its definition as "a web-based database (repository) of scholarly material which is institutionally defined (as opposed to a subject-based repository); cumulative and perpetual (a collection of record); open and interoperable (using OAI-compliant software); collecting, disseminating and storing (is part of the process of scholarly communication) (p.118).

For DSpace to become an institutional repository, equal attention should be given to the configuration and management. DSpace can be of support to any other type of digital repository, the “institutional” or “subject-focused” or any other character of the repository being brought about by the way the software application is further configured and managed. Almost 1,500 organizations worldwide, mainly in the academic environment, have installed the DSpace software platform as their institutional repository (DSpace 2013). Out of these number, nine are in Romania, of which six are in the academic environment (one of them being the Aspect platform of Transilvania University of Brasov).

SHERPA/RoMEO Programming Interface

RoMEO is a database of publisher copyright policies on self-archiving, based on the publisher’s copyright transfer agreement. It is maintained by SHERPA with support by Joint Information Systems Committee (JISC) and the Wellcome Trust. Individual journal titles, ISSNs or publishers can be searched, and each title is identified as green (can archive pre-print and post-print), blue (can archive post-print (i.e. final draft post-refereeing), yellow (can archive pre-print, i.e. pre-refereeing), or white (archiving not formally supported).

API (Application Programming Interface) is an acronym used to generally refer to collections of predefined software functions that allow writing custom applications running in a predefined environment. The application programming interface for SHERPA/RoMEO is a machine/machine interface allowing programmers’ access to SHERPA/RoMEO data within their own developed applications. For example, API can be used to embed automated searches of journals or publishing houses during a record (submission) process of a paper in a repository. Like most APIs used in the web environment, the SHERPA/RoMEO interface does not involve downloading a library of functions on the user’s computer, but calling functions from a web application server, by HTTP queries.

a) Using principle

Basically, the process is conducted as follows: applications send requests to SHERPA/RoMEO as HTTP queries, using the URL of the API interface as query-attached parameters (SHERPA/RoMEO 2013) for example:

`http://www.sherpa.ac.uk/romeo/api24.php?pub=thomson&qtype=all`

where:

`http://www.sherpa.ac.uk/romeo/api24.php` represents the URL of the application programming interface. The php extension also indicates the language used by the API, namely PHP;

`?pub=thomson` is the first parameter and, in this example, “pub” is the name of the parameter indicating a search by the publisher name and “thomson” is the value (name, or part of the publisher name);

`&qtype=all` is the second parameter, “qtype” is the name of the parameter representing the query type, and “all” indicates the fact that all the sequences specified in the first parameter must be in the publisher name (namely in the value of the first parameter).

The possibilities for querying the SHERPA/RoMEO database using API are as follows:

- by the publisher name (like in the aforementioned example);
- by the name of the journal, searching a single title, as in the following example:
`http://www.sherpa.ac.uk/romeo/api29.php?jtitle=Journal%20of%20Geology;`
- by the publication name, pursuant to the rules of the Medical Research Council:
`http://www.sherpa.ac.uk/romeo/api29.php?jtitle=oncology%20reviews&showfun`

der=mrc;

- by the publication name, searching several titles containing the specified expression: <http://www.sherpa.ac.uk/romeo/api29.php?jtitle=modern%20language&qtype=contains>;
- by the ISSN (*International Standard Serial Number*) code of the publication: <http://www.sherpa.ac.uk/romeo/api29.php?issn=2343-7006>;
- by the RoMEO color, for example for selecting all publishers with green color: <http://www.sherpa.ac.uk/romeo/api29.php?colour=green&showfunder=none>;
- by the date of updating RoMEO information (example for October 1st, 2011): <http://www.sherpa.ac.uk/romeo/api29.php?pdate=2011-10-01>;
- list of all publishers in the database (long response time of the server is expected): <http://www.sherpa.ac.uk/romeo/api29.php?all=yes&showfunder=none>;
- response in case of an invalid result (unsuccessful search), after searching a nonexistent journal: <http://www.sherpa.ac.uk/romeo/api29.php?jtitle=Recycling%20Journal>;
- error report when introducing invalid data (query) (e.g. missing search parameters): <http://www.sherpa.ac.uk/romeo/api29.php>.

b) Dynamic generation of HTML documents using XML and XSL

The possibility of easily transforming an XML document into another format such as HTML, WAP, text etc. has considerably contributed to the growing popularity of XML format (Jenkins 2008). There are two technologies which can be used for this purpose, namely: CSS (Cascading Style Sheet) and XSL (Extensible Stylesheet Language). Both belong to the family of page description languages (as is HTML), defining more precisely the style used in page description. Of these two technologies, XSL provides greater flexibility while being closer to XML concepts. It is basically an XML-based language, using stylesheets to transform the input document.

The way XSL operates is presented in Figure 1. XML is the document containing the data whereas XSL describes the data page layout and the final result can be an XML, HTML, or PDF document or other. The output format also depends on the particular technology implementing XSL, which is used to effectively apply the transformations described by XSL. For example, FOP is a particular technology (applications defined within the Apache open-source project) that uses XSL-FO as a page description language to produce PDF documents.

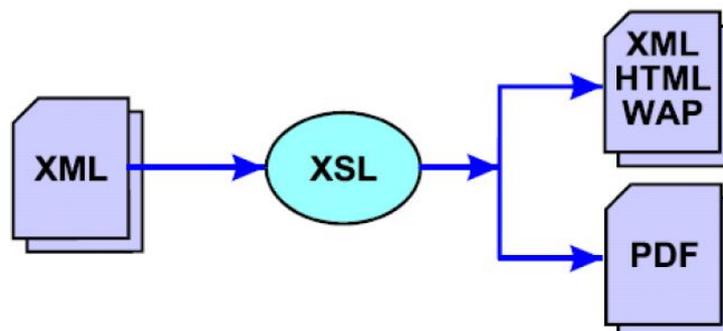


Figure 1: The Principle Diagram of using XSL

There are two versions of the XSL language used in practice: XSLT (SHERPA/RoMEO 2013), defining the transformations being applied to the XML tree; and XSL-FO (Formatting

Objects), used to transform XML documents into binary format documents such as PDF or even Microsoft Word. There are three ways that an XML document may be transformed into another type of document by applying an XSLT stylesheet:

- the XML document and the associated stylesheet are sent to the client application (browser) whose task is to effectively perform the transformation according to the information in the XSLT stylesheet. In such conditions, server load decreases but the browser should allow processing of XML documents;
- applying the XSLT stylesheet is carried out on the server itself, the resulting document (usually in HTML format) being sent to the client. Thus, processing may be carried out according to the nature of the client program;
- transformation of the XML document using an external application and placing the resulting document (HTML) on the server, being further sent to the client. This possibility is very rarely used.

The core element of the XSLT technology is the template: `<xsl:template>`. Two important elements may be found herein: the *match* attribute which specifies a path to the input tree; and the content which implements the way transformation is performed. The general form of a template is:

```
<xsl:template match="element_XPath">
...
<xsl:template>
```

The association of an XML document with an XSLT stylesheet is performed within the XML document by the processing instruction `<?xml-stylesheet>`:

```
<?xml-stylesheet href="stylesheet/Login" type="text/xsl" />
```

The *href* argument specifies the name of the XSLT stylesheet and, where appropriate, the path thereto.

Identifying the fields within the tree structure of the XML document is performed through *XPath* elements (which is sometimes described as a language, although it is not a language proper). The XPath convention is similar in functionality to navigation through the directory structure in the operating systems, such as MsDOS, Linux or Windows.

At the conceptual level, at the basis of the XML document's structure (but having no corresponding element within XML elements), is the root of the document, represented by the `"/"` character.

XPath expressions are interpreted from left to right, for example, for an XML tag which is at the first level of the tree structure (for example `<basis>`), the expression that reads the element value is `"/basis"`, and for the following levels, it could be, for example `"/basis/level1/level2"` and so on. The previous expression can be understood as: "starting from the document root, select the `<basis>` element, which is its child (the root's)." Failing to write the `"/"` character in the previous XPath expression radically changes the meaning of such expression, in which case "all the `<basis>` elements, which are children of the current node are selected". In the case of more complex XPath elements, the constituents are separated by the `"/"` character, which, as can be noticed, has a double meaning depending on the position in which it appears within the XPath element.

Moreover, there are certain situations where an XPath element must make a much more rigorous selection of the elements selected and treated within a template. It can be assumed, for example, that in a certain context only selecting the `<input>` elements is

wanted, whose “type” attribute has a value different from “hidden”. In order to achieve this, the element which will be filtered must be followed by the filter to be applied. It consists of a pair of brackets ([]), which usually frames a condition.

Integrating SHERPA/RoMEO with DSpace

Integrating information contained in the SHERPA/RoMEO database into DSpace platform can be made by combining all the information presented earlier. A general diagram of the process is shown in Figure 2. While defining a new submission for a collection on the DSpace platform, a query is sent to the SHERPA/RoMEO server, using the application programming interface it provides, which responds by an XML document containing the information required. The XML document is then processed through the XSLT transformation so as to generate the content displayed on the HTML page.

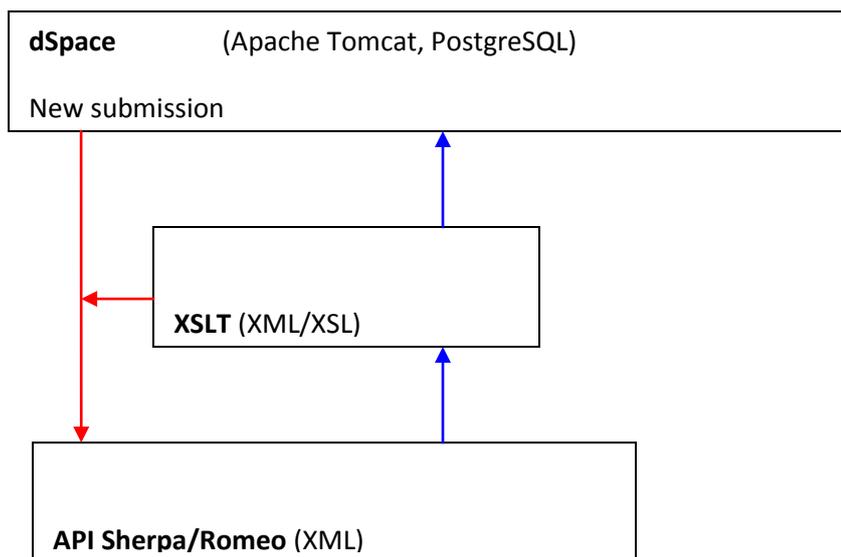


Figure 2: Diagram of integrating Sherpa/Romeo with DSpace

These actions involve taking control from the original application of the DSpace platform and inserting a page in the chain of submitting a new item on the platform. As the DSpace platform is developed by using the JSP (Java Server Pages) technology, the HTML pages are generated using functions written in Java language. The sequences newly introduced in the record chain should be preferably written in the same language, using the same JSP technology.

a) Submitting the request

The first step in recording a new article consists of the interactive setting of several variables influencing the way in which the following pages are displayed: the number of versions of the article title, the number of files to be uploaded and whether the article was published before (in this case, specifying the publisher will be requested). In this first page, intervention may be made on the JSP code to create a new button, as shown in Figure 3. The newly-created button is “*Check Publisher*”, circled in red (Figure 3).

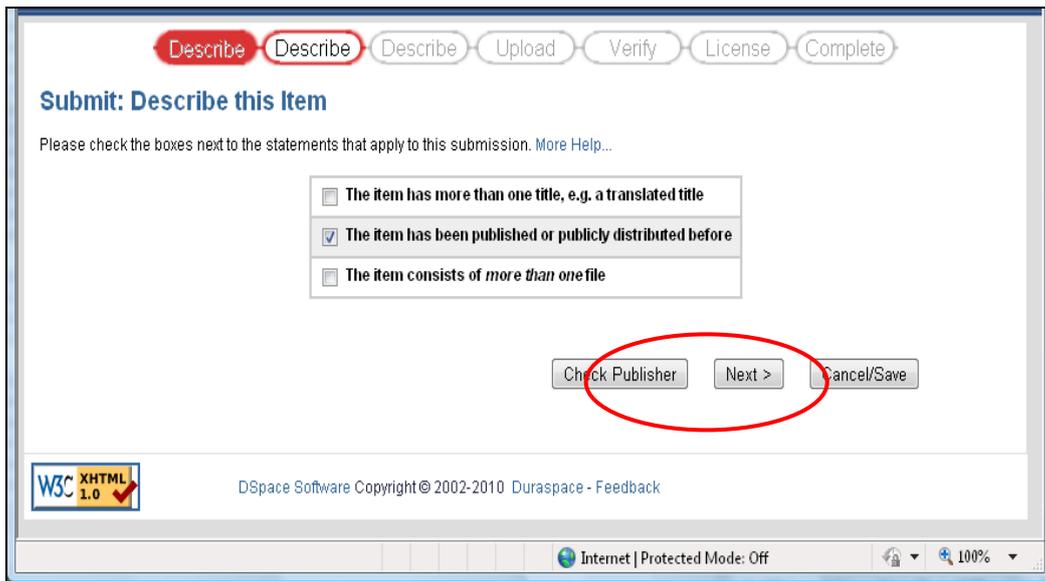


Figure 3: The First Page in the Chain of Submitting a New Item, Modified

The modified JSP code is presented in Figure 4:

```

177         <table border="0" width="80%">
178             <tr>
179                 <td width="100%">&nbsp;&nbsp;&nbsp;</td>
180                 <% //if not first step, show "Previous" button
181                 //numai in primul pas al lantului de incarcare a unui articol, afiseaza si butonul Previous
182                 if(!SubmissionController.isFirstStep(request, subInfo))
183                 { %>
184                 <td>
185                     <input type="submit" name="<%=AbstractProcessingStep.PREVIOUS_BUTTON%>" value="<fmt:message
186                     key="jsp.submit.general.previous"/>" />
187                 </td>
188                 <% } %>
189                 <!-- ***** -->
190                 <!-- Secventa urmatoare trebuie afisata numai in primul pas, pentru interfatare cu Sherpa/Romeo -->
191                 <% if (SubmissionController.isFirstStep(request, subInfo))
192                 { %>
193                 <td onclick="window.open('www.arbeit.ro!')">
194                     <input type="submit" name="romeo" value="Check Publisher"/>
195                 </td>
196                 <td>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td>
197                 <% } %>
198                 <!-- ***** -->
199                 <td>
200                     <input type="submit" name="<%=AbstractProcessingStep.NEXT_BUTTON%>" value="<fmt:message
201                     key="jsp.submit.general.next"/>" />
202                 </td>
203                 <td>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td>
204                 <td align="right">
205                     <input type="submit" name="<%=AbstractProcessingStep.CANCEL_BUTTON%>" value="<fmt:message
206                     key="jsp.submit.general.cancel-or-save.button"/>" />
207                 </td>
208             </tr>
209         </table>
    
```

Figure 4: Modified JSP Code Sequence

A simplified solution is chosen here, using JSP instructions combined with DHTML (Dynamic HTML). Pressing the new button will open a new window (pop-up) of the browser, where the actual query of the SHERPA/RoMEO server will be performed then, upon further closing the window, the user will automatically be switched to the next page – the second step of submitting the new item (where account is taken of the options being checked in the first page, even if the newly created button was pressed instead of the “Next” button).

The code lines (DHTML) achieving this are:

```
<td onclick="window.open('www.arbeit.ro')">  
  <input type="submit" name="romeo" value="Check Publisher"/>  
</td>
```

Where opening the window is triggered on the *onclick* action (mouse click on the button) (DHTML), and the transition to the next page is made by the *input* tag of *submit* type. The page opened upon pressing the button will include the query options: ISSN, Publisher, and Journal Title. A simple search page by the three criteria is presented in Figure 5. It is basically a form with three input fields and three different *submit*-type buttons, one for each search criterion. Pressing each button sends the corresponding query to the SHERPA/RoMEO server, the parameter being the one specified in the input field concerned.

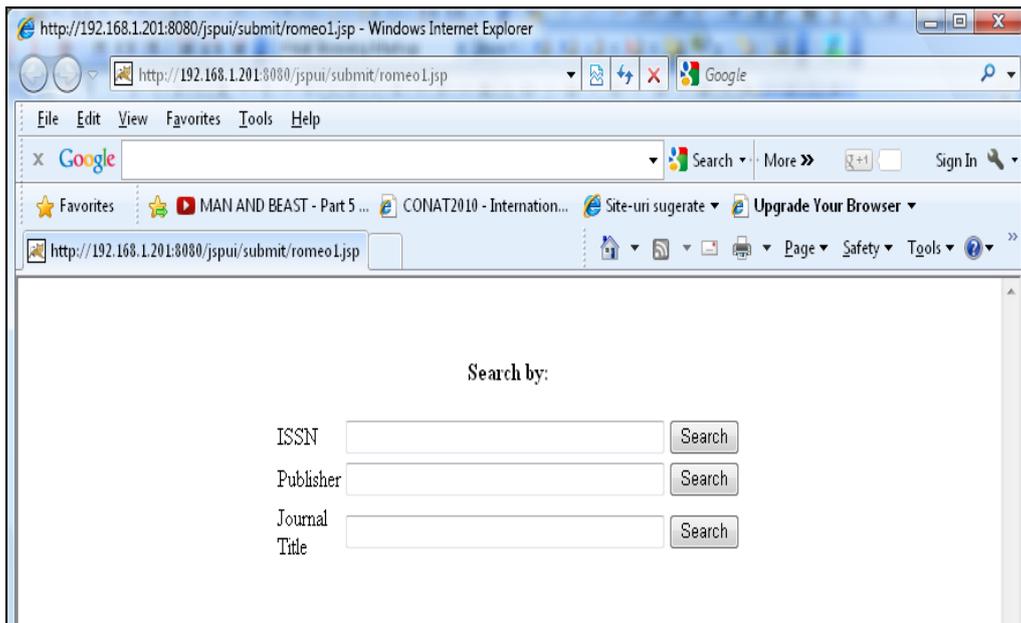


Figure 5: A Simplified Search Page Based on ISSN, Publisher and Journal Title

Identification of Publishers and their Archiving Policies in DSpace

Depending on the query sent, the SHERPA/RoMEO server responds with an XML document whose content needs to be processed. Processing consists of applying an XSLT transformation. The transformation is applied by resorting to the Java function whose source code is given in Appendix A. The XML document is received from SHERPA, so the XSL "mask" is to be made. A search by the publisher name can return multiple results, such as: <http://www.sherpa.ac.uk/romeo/api24.php?pub=university%20press&qtype=all> will return, as a result, a quite large XML document, with many subsections of *publisher* type. The XSL code processing this result is shown in Appendix B. The transformation results in an HTML file will look like in Figure 6. It is a simple list of identified publishers and the "romeo" color associated with each.

With a view to obtaining all the information about the publisher, it should be searched either the exact name of the publisher, but it is rather likely to be introduced differently from the SHERPA submission, or the ISSN code. The search by title equally displays a list as

a result, as if searching by the publisher name. In both cases the processing may be extended by the user selecting an item from the list.

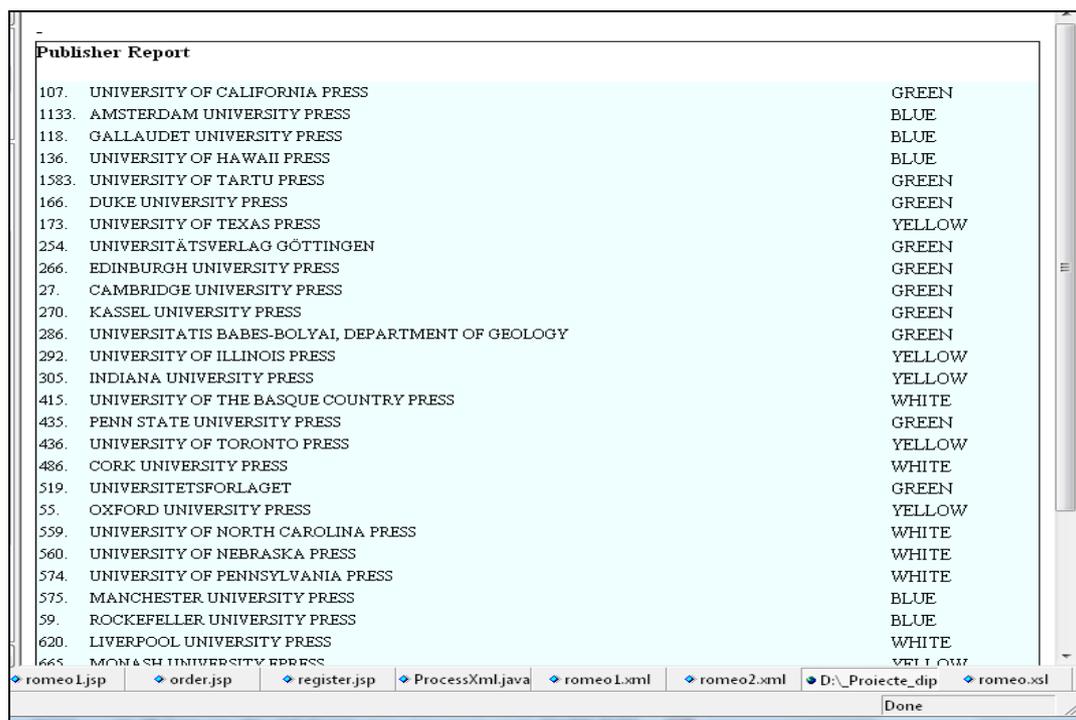


Figure 6: Search Result by the Publisher Name

DISCUSSION AND CONCLUSION

Following the analysis of opinions of the academic community within Transilvania University from Brasov, it has been determined that there are similar barriers and fears - the main fear being copyright issues - just like in the studies of Gadd, Oppenheim and Proberts (2003a, 2003b), Chan (2004), Allen (2005), Foster and Gibbons (2005), Abrizah (2009) and Stanton and Liew (2012). From the perspective of the academic community in this study, a digital repository is a very important tool within the scientific activity for disseminating results, increase visibility and scientific communication. The hypotheses of Kim (2011) are corroborated.

Developing a software application that integrates the copyright conditions imposed by publishers emerges as the solution for resolving the academic community's scepticism towards self-archiving of published articles. The integrated application is based upon interactive applications, as suggested by Shaffi, Gul and Shah (2013).

The software application has practical implications and represents an original solution to the needs of the academic community. The application is created with a low cost, and both the SHERPA/RoMEO and DSpace platforms are free to use, being the results of research projects. This approach can be very useful to other universities that experience the same problems and obstacles in populating and implementing institutional digital repositories. A practical implication concerns the easy, one-button, access to two applications simultaneously: archiving, in a digital platform, a published article and accessing the list of publishers enrolled in the platform SHERPA/RoMEO's database of publisher policies on

open sharing. Another practical implication of this approach is to reduce the time of promoting the digital repository's services and the archiving time. Regarding the originality and contribution of this approach, it is ascertained that a need identified in the self-archiving process is solved, a barrier to the use of digital repository through an original software application.

ACKNOWLEDGEMENT

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

- Abrizah, A. 2009. The cautious faculty: their awareness and attitudes towards institutional repositories. *Malaysian Journal of Library & Information Science*, Vol. 14 no. 2: 17-37.
- Allen, J. 2005. Interdisciplinary differences in attitudes towards deposit in institutional repositories. Unpublished master's thesis, Manchester Metropolitan University, UK.
- ASPECKT DSpace. 2013. Transilvania University Institutional Repository Web Page. Available at: <http://aspekt.unitbv.ro/jspui/>
- Birsan, I., Drugus, D., Stoianovici, M., Repanovici, A. 2014. Integration of open source systems for visibility of scientific production of universities. *Advances in Information Science and Applications - Volume I*. Available at: <http://www.europment.org/library/2014/santorini/bypaper/COMPUTERS/COMPUTERS1-24.pdf>.
- Chan, L. 2004. Supporting and enhancing scholarship in the digital age: The role of open-access institutional repositories. *Canadian Journal of Communication*, Vol. 29, no. 3: 277–300.
- Crow, R. 2002. The case for institutional repositories: A SPARC position paper. ARL Bimonthly Report, 223. Available at: http://www.sparc.arl.org/sites/default/files/media_files/instrepo.pdf.
- DSpace. 2013. Project webpage. Available at www.dspace.org.
- Eijkman, H. 2009. Using Web 2.0 to decolonize transcultural learning zones in higher. Education. *Campus-Wide Information Systems*, Vol. 26, no. 3: 240-255.
- Ferro N. and Silvello G. 2013. Empowering archives through annotations. *Digital Libraries and Archives, Communications in Computer and Information Science*, Vol. 354: 57-68.
- Foster, N.F. and Gibbons, S. 2005. Understanding faculty to improve content recruitment for institutional repositories. *D-Lib Magazine*, Vol. 11, no. 1. Available at: <http://www.dlib.org/dlib/january05/foster/01foster.html>.
- Gadd, E., Oppenheim, C. and Proberts, S. 2003a. RoMEO studies 1: The impact of copyright ownership on academic author self-archiving. *Journal of Documentation*, Vol. 59, no. 3: 243–277.
- Gadd, E., Oppenheim, C. and Proberts, S. 2003b. RoMEO studies 2: How academics want to protect their open-access research papers. *Journal of Information Science*, Vol. 29, no. 3: 333–356.
- Harnad, S. 2006. Maximizing research impact through institutional and national Open-Access self-archiving mandate. Invited Keynote. Open Access Institutional Repositories. *CRIS2006 Current Research Information Systems*. Bergen, Norway, 11-13 May. Available at: <http://cogprints.org/4787/2/harnad-crisrev.pdf>.

- Houghton, J., Rasmussen, B. Sheehan, P., Oppenheim, C., Morris, A., Creaser, C., Greenwood, H., Summers, M and Gourlay, A. 2009. Economic implications of alternative scholarly publishing models: Exploring the costs and benefits. *A report to the Joint Information Systems Committee (JISC)*. Available at: <http://www.webarchive.org.uk/wayback/archive/20140614211536/http://www.jisc.ac.uk/media/documents/publications/rpateconomicoapublishing.pdf>.
- Jenkins, C., Oppenheim, C., Proberts, S. and Hubbard, B. 2008. RoMEO Studies 7: Creation of a controlled vocabulary to analyse copyright transfer agreements. *Journal of Information Science*, Vol. 34, no. 3: 290-307.
- Kim, J. 2011. Motivations of faculty self-archiving in institutional repositories. *Journal of Academic Librarianship*, Vol. 37, no.3: 246–254.
- Krishnamurthy, M. 2008. Open access, open source and digital libraries: a current trend in University libraries around the world. *Program: Electronic Library & Information Systems*, Vol. 42, no. 1: 48-55.
- Lynch, C.A. 2003. Institutional repositories: essential infrastructure for scholarship in the digital age. *Portal: Libraries and the Academy*, Vol. 3, no. 2: 327-336.
- DiMicco, J., Millen, D.R., Geyer, W., Dugan, C., Brownholtz, B. and Muller, M. 2008. Motivations for social networking at work. *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work (CSCW 08)*, New York, NY.
- Palmer, K.L., Dill, E. and Christie, C. 2009. Where there's a will there's a way? Survey of academic librarian attitudes about open access. *College & Research Libraries*, Vol. 70, no. 4: 315-335.
- Repanovici, A. 2011. Intellectual property and open access to information, *International Conference on intellectual property and information management (IPM '11)*. Repanovici, A. & Murzea, C (eds.). Faculty of Law, Transilvania University of Brasov, Romania, April 7-9, 2011, pg. 153-158.
- Shaffi, S.M., Gul, S. and Shah, T. 2013. Web 2.0 interactivity in open access repositories, *The Electronic Library*, Vol. 31, no. 6: 703-712
- Sale, A. 2006. The acquisition of open access research articles, *First Monday*, Vol. 11, no. 10. Available at: <http://firstmonday.org/ojs/index.php/fm/article/view/1409/1327>.
- SHERPA/RoMEO Application Programmers' Interface. 2013. Project webpage. Available at: <http://www.sherpa.ac.uk/romeo/api.html>.
- Singeh, F.W., Abrizah, A., and Karim, N.H.A. 2013. Malaysian authors' acceptance to self-archive in institutional repositories, *The Electronic Library*, Vol. 31, no. 2: 188-207.
- Stanton, K. V. and Liew, C. L. 2012. Open access theses in institutional repositories: An exploratory study of the perceptions of doctoral students. *Information Research*, Vol. 17, no. 1, paper 507. Available at <http://InformationR.net/ir/17-1/paper507.html>.
- Swan, A. 2005. Open access self-archiving: An author study. Available at: http://www.jisc.ac.uk/uploaded_documents/Open%20Access%20Self%20Archiving-an%20author%20study.pdf
- Van House, N.A. 2002. Digital libraries and practice of trust: Networked biodiversity information. *Social Epistemology*, Vol. 16, no. 1: 99–114.
- Ware, M. 2004. Institutional repositories and scholarly publishing. *Learned Publishing*, Vol. 17, no. 2: 115-124.

Appendix A : Source Code for XSLT Transformation

```
package test;

import java.io.File;

public class ProcessXml {
    public ProcessXml() {
    }
    /**
     * Accept two command line arguments: the name of an XML
     * file, and the name of an XSLT stylesheet. The result
     * of the transformation is written to stdout.
     */
    public static void main(String[] args)
        throws javax.xml.transform.TransformerException {
        if (args.length != 3) {
            System.err.println("Usage:");
            System.err.println(" java " + ProcessXml.class.getName( )
                + " xmlFileName xsltFileName outHtmlFile");
            System.exit(1);
        }

        File xmlFile = new File(args[0]);
        File xsltFile = new File(args[1]);
        File htmlFile = new File(args[2]);

        javax.xml.transform.Source xmlSource =
            new javax.xml.transform.stream.StreamSource(xmlFile);
        javax.xml.transform.Source xsltSource =
            new javax.xml.transform.stream.StreamSource(xsltFile);
        javax.xml.transform.Result result =
            new javax.xml.transform.stream.StreamResult(htmlFile);

        // create an instance of TransformerFactory
        javax.xml.transform.TransformerFactory transFact =
            javax.xml.transform.TransformerFactory.newInstance( );

        javax.xml.transform.Transformer trans =
            transFact.newTransformer(xsltSource);

        trans.transform(xmlSource, result);
    }
}
```

Appendix B : XSL Code Processing

```

<?xml version="1.0" encoding="ISO-8859-1" ?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xalan-nodeset="http://xml.apache.org/xalan">
  <xsl:output method="html" version="1.0" encoding="ISO-8859-1" indent="yes" />
  <xsl:template match="/">
  <xsl:apply-templates />
  </xsl:template>
  <xsl:template match="publishers">
  <table id="sortingTable" border="1" cellspacing="0" width="100%" style="border-collapse:
    collapse" bordercolor="#111111" cellpadding="0">
  <tr>
  <td width="100%">
  <p style="font-size: 15px;">
  <b>Publisher Report</b>
  </p>
  <TABLE width="100%" border="0" cellspacing="0" cellpadding="2">
  <tbody>
  <xsl:call-template name="publisher" />
  </tbody>
  </TABLE>
  </td>
  </tr>
  </table>
  </xsl:template>
  <xsl:template name="publisher">
  <xsl:for-each select="publisher">
  <xsl:sort select="@id" />
  <tr style="background-color:#efffff;font-size: 12px;">
  <xsl:call-template name="pub_td" />
  </tr>
  </xsl:for-each>
  </xsl:template>
  <xsl:template name="pub_td">
  <td width="5%">
  <xsl:value-of select="@id" /> . </td>
  <td width="65%" style="text-transform:uppercase">
  <xsl:value-of select="name" />
  </td>
  <td width="15%" style="font-size: 14px;" />
  <td width="15%" style="font-size: 14px;text-transform:uppercase;">
  <xsl:value-of select="romeocolour" />
  </td>
  </xsl:template>
  </xsl:stylesheet>

```