Library Collections, Acquisitions, & Technical Services xxx (2009) xxx–xxx

Contents lists available at ScienceDirect

Library Collections, Acquisitions, & Technical Services

journal homepage: www.elsevier.com/locate/lcats

An attempt of data exchange between the institutional repository and the information environment for the management of scientific research—ARP

Jianxia Ma *, Yuanming Wang, Zhongming Zhu, Runhuan Tang

Lanzhou Library of National Scientific Library, CAS 730000, China

ARTICLE INFO

Available online xxxx

Keywords:
Institutional repository
DSpace
Scientific information environment
ARP
Data exchange
EXCEL

ABSTRACT

In the present paper an attempt of data exchange between the institutional repository based on DSpace and the Academy Resource Planning System (ARP) of Chinese Academy of Sciences (CAS), which is the information environment for management of scientific research in CAS, was described. It includes the development of a tool based on browser which imports data from EXCEL into DSpace and a tool which exports data from DSpace into EXCEL. Consequently, data can be exchanged between ARP and DSpace through EXCEL. In addition, the way to customize the input form and the display page according to different types of documents was introduced.

1. Background

An Institutional Repository (IR) is an effective way to preserve the knowledge assets of a scientific institution. The National Scientific Library of Chinese Academy of Science (CAS) is planning to develop and implement institutional repository for use by scientific institutions in CAS. Lanzhou Branch of National Scientific Library of CAS is undertaking the task. We are developing and implementing the repository based on DSpace. The Institute of Mechanics of CAS (IMECH) will be the first institution to deploy the IR in CAS. During the implementation, the librarians of IMECH in the project put their needs forward and suggested that there should be interoperation between DSpace and Academy Resource Planning System (ARP) of Chinese Academy of Sciences [1], i.e., the administrator of IMECH IR can import/export data between ARP and DSpace. Then the librarians in IMECH will audit the supplied metadata and submit the full-text of metadata, so as to relieve the scientists of that burden.

In view of the need of IMECH, we analyzed the data in ARP. ARP is the information management platform of scientific research in CAS. The application system in ARP includes the research programs and administration system, the human resources management system, the financial management system, the scientific facility management system, the infrastructure management system, the e-government system, the education resources management system, and the evaluation system. ARP is based on ORACALE9i and Redflag Linux operating system. For the sake of security, staffs in the institution have to visit the system through VPN and browser, and it is difficult for them to connect with the Oracle9i directly. But the data in ARP can be exported/imported into EXCEL, so that EXCEL could act as a bridge to interchange data between DSpace and ARP. There are 10 types of data related to the knowledge output of the institution, including monographs, journal articles, conference reports, awards and achievement identification, standards, patents, software, medical, and consult reports. The common fields of these types are the title and the author fields, other more fields are different in terms of type. Since ARP is facing the administrative demand of institution, information in ARP related to knowledge output are only the metadata of the items without full-text.

After analysis, five points of the demand were recognized:

- Exporting data related to knowledge output from ARP to EXCEL.
- Importing data from EXCEL into DSpace.

* Corresponding author.
E-mail address: majx@lzb.ac.cn (J. Ma).

1464-9055/$ – see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.lcats.2009.01.003

- Customization of the input form and display page according to different types of documents.
- Exporting data from DSpace into EXCEL.
- And importing data from EXCEL into ARP.

According to the demand, some improvement of the IR has been made.

2. The attempt of data exchange between DSpace and ARP

We think that the demand of IMECH for the implementation of IR would be popular among the institutions of CAS. Since data in EXCEL is highly accessible and processable, it will be of significant use to import/export data between DSpace and EXCEL in the future. And EXCEL can be used as a bridge between DSpace and ARP or other types of applications. Therefore, some further development and configuration were carried out based on DSpace.

2.1. Development of a tool to export/import data between ARP and EXCEL

For the sake of the security of ARP and the convenience of data-maintenance staffs, a data-output/input tool based on Microsoft Windows.net and ACCESS running in Windows Operating Systems was developed.

The installation and running of the tool need dot net framework, Oracle9 client, VPN client. Since DSpace does not provide the function of checking and deleting repeated items automatically, some other measures were taken. We exported the document data and staff data from ARP into a temporary database in ACCESS, maintained a history database in ACCESS which stored the data exported from ARP recently, and recorded the time of export. During the exporting, we compared the data planned to be exported from ARP with the data in the history database in terms of timestamp and title, deleted the duplicating data and added a timestamp of export to guarantee that the data in the history database is the latest data input into ARP, and finally transferred the data in the history database into EXCEL. And similarly, we also imported the data in EXCEL into ARP.

2.2. Modification of metadata-schema in DSpace

Since there are 10 types of documents in ARP, some fields were added to accept data from ARP and to present the information clearly. We logged in the administrator's interface of DSpace, entered the page of meta-data registry and there appeared the name-space of Dublin-core. We kept Dublin core namespace and extended it by clicking the link http://dublincore.org/documents/dcmi-terms/ to add or modify metadata elements. For example, if we want to add a new field named call number, we could add a new element identifier with the descriptor of callnum and add the note as "callnum of the thesis", thus we get a new field. After new fields in the GUI were added, a new element dc-type in the source code file "/config/registries/Dublin-core-types.xml" was created.

2.3. Data import/export between EXCEL and DSpace

As the import/export tools of DSpace must be run by command line, and the data must be compatible with the DSpace [2] (i.e. every item is in fact a folder including several files such as the content file, Dublin_core.xml, handle, license.txt and full-text or jpg

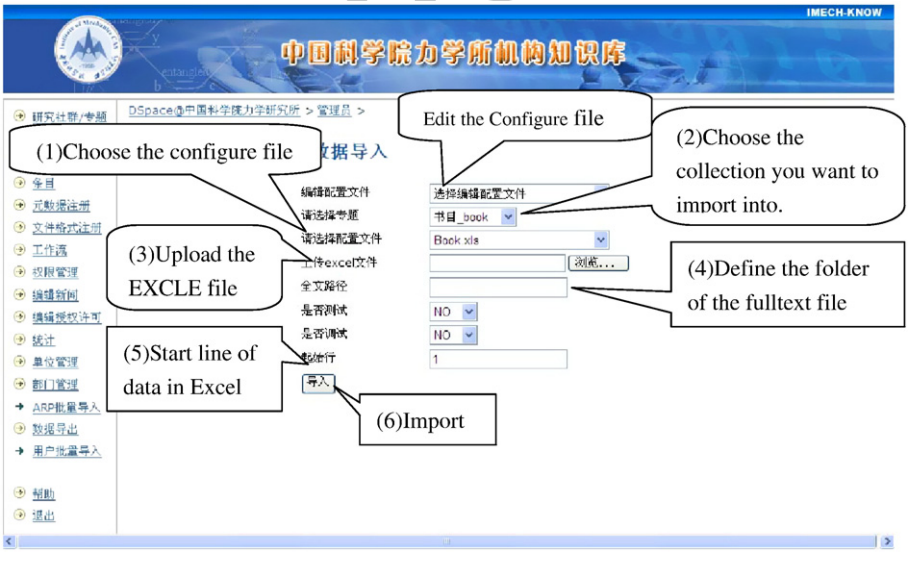

Fig. 1. Workflow of the tool importing data from EXCEL into DSpace.

Please cite this article as: Ma, J., et al., An attempt of data exchange between the institutional repository and the information environment for the management... *Library Collections, Acquisitions, & Technical Services* (2009), doi:10.1016/j.lcats.2009.01.003

file), we proposed that calling the item import tools through IE browser, according to the need of IMECH, would be much convenient for the data maintenance.

2.3.1. The tool to import data from EXCEL into DSpace

The workflow of the tool is shown as follows. Firstly, we chose a destination collection. Secondly, we should map the field of EXCEL with metadata of DSpace. The map was done through a configuration XML file. This is an important procedure for importing data correctly from EXCEL into DSpace. We could choose to edit the configure file through a form if necessary. And then we chose an EXCEL file to be uploaded to server temp directory and identified the start line of the EXCEL file to export. In the end, we imported data from EXCEL into DSpace. Fig. 1 shows the workflow of the tool importing data from EXCEL into DSpace.

In view of the tedious work of adding information into DSpace for each staff of the institution, we also exported the staff information, including staff name, department, email, phone, title and so on from ARP into EXCEL, and imported the information into DSpace.

2.3.2. The tool to export data from DSpace into EXCEL

Once the knowledge output data have been submitted into DSpace, these data should be exported from DSpace into ARP to avoid repetitive work. Therefore, a tool to export data from DSpace into EXCEL was developed.

Firstly, we chose a destination collection and an EXCEL template into which we would import data. Then, we got a page showing the column name of EXCEL and all the names of metadata that have value, and we mapped the field of EXCEL with the appropriate metadata of DSpace. After that, we defined the relative directory to save full-text file in case it exists, and finally, we exported data from DSpace into EXCEL. Fig. 2 shows the workflow of the tool to export data from DSpace into EXCEL.

The quality of data export/import is highly dependent on the mapping between the field in EXCEL and the metadata in DSpace. From Fig. 3, we can see the column name of EXCEL in the left, and a series of dropdown list in the left. We can choose the metadata in DSpace from the dropdown list. The mapping procedure is very clear, and also convenient for librarians.

2.3.3. Description of the class

The main class in the tool of data import from EXCEL into DSpace includes ConfigReader, ExcelReader, ExcelRow, Importer, and ImportServlet.

ConfigReader is in charge of reading the configure file based on the dom4j [3] as xml resolving tool.

ExcelReader takes charge of reading the xml file based on the open-source software POI [4] of Apache as EXCEL resolving tool.

ExcelRow is to deal with a line in EXCEL.

Importer is the data importer.

ImportServlet is the http response class of the data importer.

When the data were imported, the EXCEL file exported from ARP could be chosen by clicking the data importer tool button in DSpace administrator; consequently the EXCEL file could be uploaded by the program. And then the class ImportServlet and the class Importer were called. The EXCEL file was read by the class ExcelReader and ExcelRow, and the configuration file was read by the class ConfigReader for the sake of matching the field in terms of the configure file and inserting the item into the database of DSpace appropriately.

Fig. 4 shows the main classes in the tool importing EXCEL into DSpace and their relationship.

The main class in the export tool from DSpace into EXCEL includes ExcelReader, ExcelRow, ExcelWriter, Exporter and ExportServlet.

ExcelReader is in charge of reading the EXCEL file.

ExcelRow is in charge of getting the column name of the EXCEL.

Exporter takes charge of mapping between the column of EXCEL and metadata in DSpace.

ExcelWriter is used to write into EXCEL.

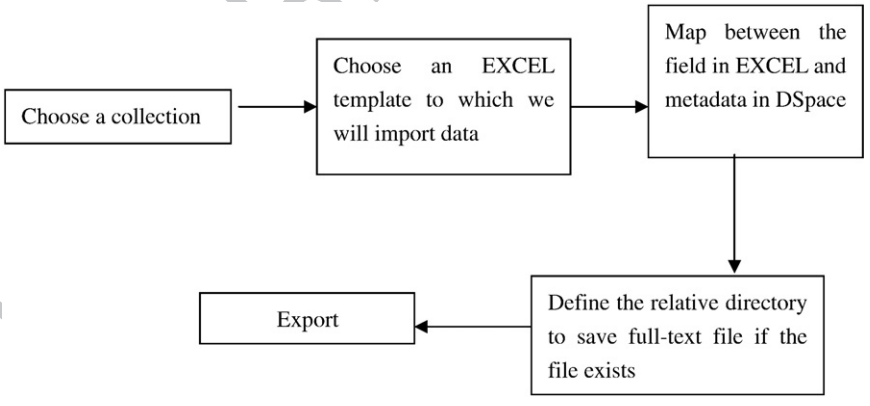

Fig. 2. Workflow of the tool exporting data from DSpace into EXCEL.

Fig. 3. A snapshot of the tool exporting data from DSpace into EXCEL — mapping.

ExportServlet is the http response class of data exporter.

And the related classes in DSpace were called to get the metadata of the items in the target collection. Fig. 5 shows the main class in the tool exporting data from DSpace into EXCEL and their relationship.

As shown above, EXCEL serves as a bridge in the process of data exchange between DSpace and ARP. We think it can also act as a bridge to exchange data between DSpace and other applications.

2.4. Customization of the submission GUI according to the different types of document

Since we customized the metadata schema in DSpace in order to accept different types of data from ARP, we should customize the submission GUI in terms of the different types of data.

In DSpace, input-form.xml controls the submission GUI [5]. The file was changed to customize the pages, the meta-data displayed in each page, the sequence of displayed meta-data, tabs of the fields, notation information, and the choosing element of menu driven fields.

The structure of the input-forms.xml is described as follows. The xml file includes three top elements including the form-map, the form-definitions and the form-value-pairs.

In the tag <**form-map**>, you should define the relation between collection and the form.

In the tag<**form-definitions**>, you should define the form mentioned in <form-map>, and define the detail of each element,

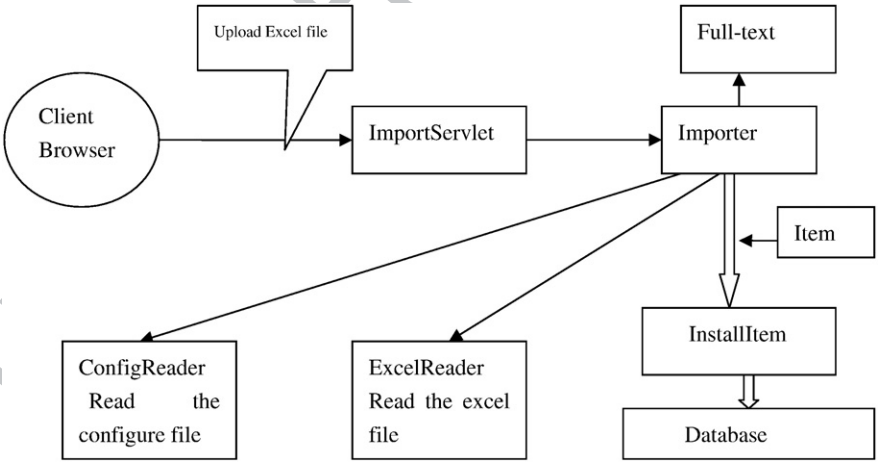

Fig. 4. Main classes in the tool importing EXCEL into DSpace.

Please cite this article as: Ma, J., et al., An attempt of data exchange between the institutional repository and the information environment for the management... *Library Collections, Acquisitions, & Technical Services* (2009), doi:10.1016/j.lcats.2009.01.003

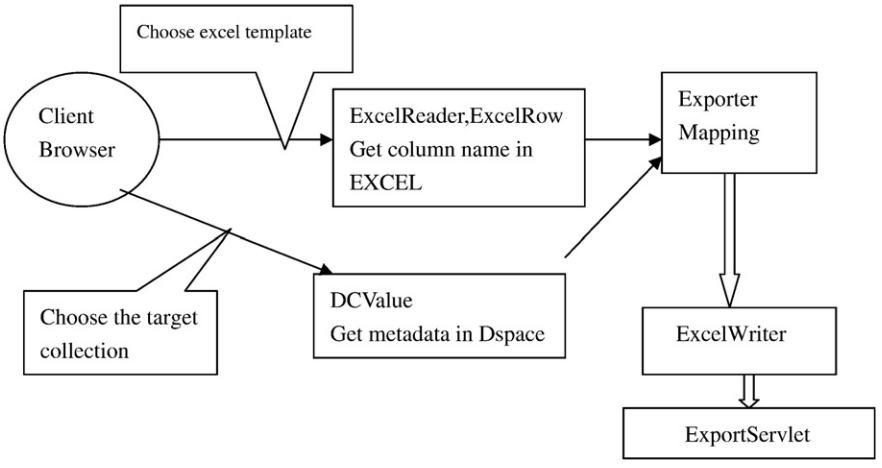

Fig. 5. Main classes in the tool exporting data from DSpace into EXCEL.

such as various attributes of a field, including its name, what is the data type, whether it is repeatable, what should be its heading in the input form, whether any help message should be displayed, whether it is mandatory or optional, etc.

In the tag <**input-type**>, you should be clear about what type of input is required for a particular field. In DSpace, input type can be any one of the following: name, textarea, onebox, twobox, date, dropdown, or qualdrop_value.

The tag <**form-value-pairs**> is defined outside the 'form-definitions', which is mainly used for pull-down and qualdrop_value data types.

Three steps were taken to customize the submit GUI. (An example of adding an input form of callnum for a conference paper is given in the appendix.)

Step 1: add a form map

You should add the new forms with their 'form-name' along with the associated collection-handle in the element<form-map>. e.g.

```
<form-map>
<name-map collection-handle="default" form-name="traditional" />
<name-map collection-handle="311007/4" form-name="conf" />
</form-map>
```

The third line in the above code indicates that the form 'conf' will be used for the collections 311007/4, whereas the 'default' form will be used for the rest of the collections.

Step 2: add new form elements with extended Dublin-core fields in form-definition tag

As all the different forms are accommodated in the same file, they should be distinguished from each other with a name, which is called 'form-name'.

Each input 'form-name' should be associated with one or more collections and the form-name in form-definition tag should be the same with the corresponding value defined in the form-map tag. In the <form-definition> tag, the <form-name>, <page>, <field>, <dc-schema>, <dc-element>, <dc-qualifier>, <repeatable>,

<label>, <hint>and <input-type> should be defined. And the page of the form should also be defined. For example: <page number="1">.

Step 3: define the form-value-pairs

If you have pull down menus in your input form, you should define the form-value-pairs. The tags, such as <value pairs> and <pair>, included in this element are mainly used for pull-down and qualdrop_value data types.

2.5. Indexing

If one feels the necessity of indexing some of the newly added elements, one can modify dspace.cfg file so that the required elements can be indexed.

For example: search.index.6=contributor: contributor.org The first column search.index [number] simply indicates how many fields you are indexing. In the second column, you have to state which field should be indexed. There is a difference between the field names used by metadata schema like Dublin Core and the way Lucene search engine's naming pattern for the field name. DSpace uses qualified Dublin core where the notation is represented as 'fieldname.qualifier', the Lucene search engine uses only field names [6]. Once the difference is clear, it should not be difficult to index the required fields in a metadata schema.

Please cite this article as: Ma, J., et al., An attempt of data exchange between the institutional repository and the information environment for the management... *Library Collections, Acquisitions, & Technical Services* (2009), doi:10.1016/j.lcats.2009.01.003

2.6. Customization of display page according to different types of document

Since different types of document have different DC fields, the display page was customized in order to display the data according to the different types of documents.

To deal with the problem, the dspace.cfg was changed and the source code of DSpace which is in charge of item display was modified. In the file dspace.cfg, webui.itemdisplay.full was added and all the metadata elements and qualifiers will be displayed in the full item.

For example:

```
webui.itemdisplay.full = dc.title, \
dc.title.alternative, \
dc.contributor, \
dc.contributor.author, \
dc.contributor.advisor, \
dc.contributor.org \
.....
dc.identifier.callnum, \
... ... .....
```

Then the file src/org/dspace/app/webui/jsptag/ItemTag.java was changed. A function akoRenderFull was added to display newly added elements according to the dspace.cfg. Then based on the dspace.cfg and the ItemTag.java, we can get the customized display page now.

3. Conclusion

In this paper, we have presented our attempt to integrate IR with the information environment of the management of scientific research, i.e. to exchange data between the institutional repository based on DSpace and the Academy Resource Planning System (ARP) of the Chinese Academy of Sciences, which is the information environment for management of scientific research in CAS. The attempt includes the development of two tools to exchange data between DSpace and EXCEL. One tool imports data from EXCEL into DSpace and another tool exports data from DSpace into EXCEL. Consequently, data can be exchanged between ARP and DSpace through the bridge of EXCEL. The tools can be run through browser conveniently. This made it convenient for DSpace to reuse data in other application systems and hence relieve the burden of librarians. Since data in EXCEL is highly accessible and can be easily processed, we considered that EXCEL, as a bridge to exchange data between DSpace and other system, would be very useful.

However, we were also faced with some problems during our development.

First, it is somewhat difficult for us to extend metadata schema using DSpace. We had to extend metadata schema, customized input form and displayed page according to the extended metadata. Thus, we think the function of supporting the different metadata schema and convenient modification of input form and display page according to extended metadata will be important for DSpace and other IR software.

Second, there is no automatic mechanism to check and remove reduplicated items in DSpace. So we added timestamp to sign the export/import date when we were exporting/importing data. The timestamp, in combination with the title helps us to judge if the data is reduplicated or not, hence, avoiding import/export repeated items. Our experience suggests that there should be some automatic mechanism to check and remove reduplicated items in DSpace.

Our work on IR based on DSpace will continue: We are thinking about how to release the burden of submitting items into DSpace. We have planned to do further tests in extracting metadata from pdf file and doc file when submitting an item, and in integrating IR with personal knowledge management tools and other information systems to approach the goal of integrating IR with scientific information environment.

Acknowledgments

The financial support of this work by the National Planning Office of Philosophy and Social Science under 07BTQ019, and by the National Scientific Library of CAS under the project Study and Development of Federation Institutional Repository in Chinese Academy of Science is gratefully acknowledged.

Appendix A

An example of adding an input form of callnum for a conference paper:

```
<?xml version="1.0"?>
<!DOCTYPE input-forms >
<input-forms>
<form-map>
<name-map collection-handle="311007/4" form-name="conf" />
</form-map>
<form-definitions>
```


Please cite this article as: Ma, J., et al., An attempt of data exchange between the institutional repository and the information environment for the management... *Library Collections, Acquisitions, & Technical Services* (2009), doi:10.1016/j.lcats.2009.01.003

```
<form name="conf">
<page number="1">
.....
<field>
<dc-schema>dc</dc-schema>
<dc-element>identifier</dc-element>
<dc-qualifier>callnum</dc-qualifier>
<repeatable>false</repeatable>
<label>callnum of the conference proceeding</label>
<input-type>onebox</input-type>
<hint>Please input the callnum of the conference proceeding </hint>
<required></required>
</field>
.....
</page>
</form> .....
</form-definitions>
<form-value-pairs>
<value-pairs value-pairs-name="common_identifiers" dc-term="identifier">
<pair>
<displayed-value>call number</displayed-value>
<stored-value>callnum</stored-value>
</pair>
</value-pairs>
……
</form-value-pairs>
</input-forms>
```

References

[1] Introduction to ARP, Retrieved from the World Wide Web http://www.arp.cn/

[2] DSpace System Documentation, Retrieved from the World Wide Web http://www.dspace.org/index.php?option=com_content&task=view&id=151&Itemid=116

[3] dom4j:flexible XML framework for Java. Retrieved from the World Wide Web. http://sourceforge.net/projects/dom4j/

[4] The Apache POI Project Retrieved from the World Wide Web http://poi.apache.org/

[5] Custom Metadata-entry Pages for Submission, Retrieved from the World Wide http://www.dspace.org/index.php?option=com_content&task=view&id=155

[6] Prasad, ARD (2006). Implementing LOM Schema in DSpace. Retrieved from the World Wide Web. https://drtc.isibang.ac.in/handle/1849/221

Please cite this article as: Ma, J., et al., An attempt of data exchange between the institutional repository and the information environment for the management... *Library Collections, Acquisitions, & Technical Services* (2009), doi:10.1016/j.lcats.2009.01.003