

Digital Author Identifier

Background

At Leiden University, The Netherlands, we wanted to add a DAI to authors in Dublin Core metadata.

This implementation let's administrators add DAIs to existing authors, and provides an API for displaying / using the DAI.

How it works

The DAI of a author is defined by using the [Basic RDF implementation](#) . Because the author in Dublin Core metadata does not have a DSpace ID, a RDF resource is generated to identify the author and link it to the DAI (an RDF value).

Which DSpace version

The modifications were made to Dspace 1.3.2. This is also the version of DSpace where this functionality was tested extensively.

It should be possible to use this with another version of DSpace.

How to install

This installation provides a way for administrators to associate DAIs to authors, and provides an API for displaying / using the DAI.

Step 1: Install the Basic+RDF+implementation

See [Basic RDF implementation](#) for details on how to do this.

Step 2: Add java classes

- Download the file [RDFServlet.java](#) and place it in the directory src/nl/leidenuniv/dspace/app/webui/servlet/RDFServlet.java

Step 3: Change etc/dspace-web.xml

Add the following:

```
<servlet>
  <servlet-name>rdf</servlet-name>
  <servlet-class>nl.leidenuniv.dspace.app.webui.servlet.RDFServlet</servlet-class>
</servlet>
```

Step 4: Change JSP page jsp/tools/edit-item-form.jsp

- Make a local copy of this file in the directory jsp/local/tools/
- Add a button where you feel it is appropriate by inserting the following html:

```
<form method="post" action="<%= request.getContextPath() %>/rdf">
  <input type="hidden" name="item_id" value="<%= item.getID() %>" />
  <input type="submit" name="edit_rdf" value="Edit RDF" />
</form>
```

Step 5: Add JSP page jsp/local/rdf/rdf-item-edit.jsp

- Download the file [Missing File: Rdf-item-edit.jsp](#) and place it in jsp/local/rdf/

Step 6: build / install / restart

Do the stuff you normally do when deploying a new version of DSpace.

Notes

- To display / use the DAI, here is a code example:

```
RDFpredicate predicate = RDFpredicate.find(context, "hasDAI");
if (predicate != null) \{
    RDFtriple[] triples = RDFtriple.findAll(context,item,predicate,(Object)null);
    for (int i=0;i<triples.length;i++) \{
        Object dai = triples[i].getObject();
        // use the dai as a String
    }
}
```

- This implementation also works for Digital Object Identifiers (DOI)

Contact me

If you need help, or have any comments, or you just want to inform me that you (are going to) use this, please contact me at [schaik \(at\) library.leidenuniv.nl](mailto:schaik@library.leidenuniv.nl)