

Citation Generation

This is out-dated but our most recently migration is on DSpace 6.x. I may update this document later with recently update.

IMPORTANT NOTE: If you have static citation stored in dc.identifier.citation field, they are going to be wiped out with this implementation. DON'T use this method if you want to keep those static citations. Or you may change the code and have the citation generated in another field.

Outline

This page will introduce you how to generate the citation for a DSpace item from other metadata fields. The format of citation can be defined in an XML configuration file. The code was implemented on DSpace 1.5.0 and tested on XMLUI and OAI. Should work on JSPUI too. The features include -

1. Generate the citation in a format defined in an XML configuration file. Basically, you can choose any format your want, e.g. IEEE, Chicago, or even your own format.
2. Define citation format in different level - community, collection and item's document-type (based on dc.type).
3. Sensitive field - citation won't be generated if provided sensitive field(s) are empty or null.
4. Flexibility to format a metadata field before it goes to the citation.

Configuration Files

citation-config.xml is used to config the citation format. Here is an example of [Citation-config.xml](#) and it's DTD file [Missing File: Citation-config.dtd](#). Please refer to citation-config.xml for detail instructions.

To activate the citation generation, place the line below in [DSPACE]/config/dspace.cfg file -

```
citation-config = ${dspace.dir}/config/citation-config.xml
```

Comments this line out will turn off the citation generation.

DSpace API

1. CitationManager.java

This is a new java class which is used to read and parse the citation configures and assemble the citation for the given item. Please download [CitationManager.java](#) and place it under [dspace]/dpace-api/src/main/java/org/dspace/app/util directory.

2. FormatIt.java

This new java class is used to customize the metadata fields for the citation. Please see citation-config.xml for details. [FormatIt.java](#) is under [dspace]/dpace-api/src/main/java/org/dspace/app/util directory too.

3. Item.java

Item.java is where the citation gets generated. When you access an item, the citation will be generated (if configured) and stored into field dc.identifier.citation. It is a in-memory copy and won't be written back to the database. It gives you the flexibility to change the citation format whenever you want. Download [Item.java](#) (for DSpace 1.5.0) and replace the one under [dspace]/dpace-api/src/main/java/org/dspace/content directory. Please backup your old copy of Item.java.

4. DSpaceCocoonServlet.java and LoadDSpaceOAConfig.java

Finally, you need to load your configurations. For XMLUI, this can be done in DSpaceCocoonServlet.java. Download [DSpaceCocoonServlet.java](#) and replace the one under [dspace]/dpace-xmlui-api/src/main/java/org/dspace/. If you would like to expose the citation in OAI harvesting, please refer to the changes made in DSpaceCocoonServlet.java and do the same in LoadDSpaceOAConfig.java. Also, as mentioned earlier, this should work for JSPUI too. You just need to find the appropriate file to load the configuration.

Manakin Theme

You can access the generated citation in dc.identifier.citation just like you access title or author in a Manakin Theme. However, to correctly render the HTML tags in your citation format, you need this xslt script [Html-tag-parser.xsl](#) to parse them. Here is an example of how to use it -

```
<xsl:copy>
<xsl:call-template name="parse">
<xsl:with-param name="str" select="dim:field[atelement='identifier'][atqualifier='citation'][1]/node()"/>
</xsl:call-template>
</xsl:copy>
```

Here you go

If you have everything ready in place, you should be able to compile the code, update the dspace and deploy the changes to the application server now.