

Nested metadata in DSpace items

This functionality has been introduced in DSpace-CRIS 5.7.0 (and DSpace-CRIS 6 RC2)

In some case you need to store group of related information such as the author and their affiliation as appear in the publication "header", the project title, grantno, funding program related to the publication or the Journal Title, the ISSN, the E-ISSN, etc.

In the first two examples the group of metadata is repeatable, this mean that we need to couple the first value in the "author names" with the first value in the "affiliations" etc.

Generally, in DSpace-CRIS you want to **solve these scenario using additional entities to link with the item**. This is of course the proper way to manage the use case but supporting in the system additional entities is not always an option or possible at all the extend. In addition, you could want to allow the submitter to input proper rich information during the item submission to be saved / consolidated after in a separate entity (see also [How to collect data about CRIS entities](#)).

In other case, also if the information are related to an external entities as in the example of the author affiliation, these information are also needed to be stored / freeze on the item level. For instance you could be interested to know which institutions are listed on a publication as affiliations regardless of the institution that is linked "now" to each authors. Also if you assume to know exactly the affiliation of each author at any time (complete career track) you will be unable to know which institution he has listed in the publication signature (this because the date to use is partially uncertain and also because the author could have made a choice between several institutions depending on the one that have actually supported his publication work).

To meet such requirement, DSpace-CRIS allows you to specify that a metadata is a children to another metadata. To do that in the input form you need to include a `<parent>` tag in the metadata field definition referring to the parent metadata. For example the following definition will configure the contributor department as a child of dc.contributor.author

```
<field>
  <dc-element>contributor</dc-element>
  <dc-qualifier>department</dc-qualifier>
  <label>Author Department</label>
  <input-type>onebox</input-type>
  <repeatable>true</repeatable>
  <parent>dc_contributor_author</parent>
  <required></required>
  <hint>Enter an affiliation for each author.</hint>
</field>
```

Here how the UI looks like with such configuration

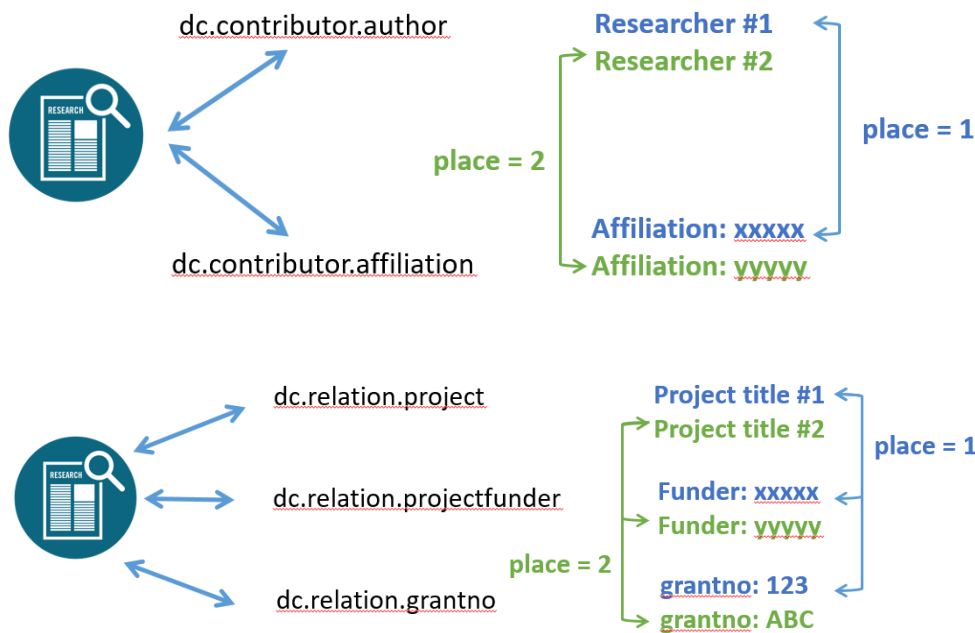
Enter the names of the authors of this item in the form Surname, Firstname [i.e. Smith, Josh or Smith, J].

Authors	Bollini, Andrea	Q	Remove Entry
	Author Department		
	4Science		
Authors	Researcher, Sample	Q	Remove Entry
	Author Department		
Authors	Pascarelli, Andrea	Q	Remove Entry
	Author Department		
	4Science		
Authors		Q	+ Add More
	Author Department		

you can see such configuration in the "Other materials" collection of the [DSpace-CRIS demo](#) and [testdrive](#).

the child field **must be in the same page than the parent metadata and listed in the input-forms.xml before than the parent metadata**. All the child fields must be repeatable or less as the parent child. It is not possible to configure a child metadata as repeatable itself, i.e. to allow the user to input several values for a single metadata all associated to the same value of the parent metadata. A possible workaround to this limitation is to configure multiple children metadata such as contributor.department, contributor.department2, etc.

A the database level the values of the children and parent metadata are stored independently to each other. A placeholder (`#PLACEHOLDER_PARENT_METADATA_VALUE#`) is introduced to guarantee that children metadata are always in the same order and count than the parent metadata also when some "parent value" doesn't have a value for a children metadata (author without affiliation). Such placeholder is skipped in indexing and visualization.



Pre-fill information from the linked authority

This functionality has been introduced in DSpace-CRIS 5.10.0 (March 2019)

As discussed above, in some case it is needed to store nested information or additional information that are loosely related to a linked object. It is clear for instance that the author affiliation that appear on the paper is often the current affiliation of the researcher or at least one of his previous affiliation. In such case it is convenient for the user to have such information pre-filled when the researcher is selected so to only change it when needed.

To do that DSpace-CRIS uses special extra information on the ChoiceAuthority, everything that is prefixed with `data-` will be copied over the corresponding metadata if present in the same page than the metadata authority controlled. For instance if the authority report an extra property named `data-dc_contributor_department` and in the page where the authority is enabled (ie typically where is included the `dc.contributor.author`) is presente a metadata `dc.contributor.department` (please note the different syntax used here `_` and `.`) the extra property will be used to fill this metadata.

Out of box DSpace-CRIS provides configurable implementations of the CRIS Authority that support exposure of CRIS metadata (and linked CRIS object metadata) as extra properties, they are

```
org.dspace.app.cris.integration.RPMultiAuthority
org.dspace.app.cris.integration.ORCIDMultiAuthority
```

the extra properties to expose are configured in the [cris-authority-metadatagenerator.xml](#)

In the `cris-authority-metadatagenerator.xml` you will see

```
<property name="relatedInputformMetadata" value="dc_contributor_department"/>
```

defines where the additional information will be saved, it uses the `editmetadata.jsp` syntax so `_` instead than `.`

```
<property name="schema" value="affiliation"/>
```

defines the property to use in the linked RP to fill the extra information, in such case the affiliation "nested"

```
<property name="element" value="affiliationorgunit"/>
```

defines the property inside the previous nested to use, so the organisation

```
<property name="qualifier" value="name"/>
```

finally, define which attribute of the linked orgunit use as display value

