Report on META-SHARE & CLARIN metadata interoperability

The full META-SHARE (hereafter "MS") metadata schema v3.0[1] has been uploaded in the CLARIN Component Registry (http://catalog.clarin.eu/ds/ComponentRegistry/#); however, the CLARIN implementation of the MS schema is not exactly the same with the one described in the XSD (http://metashare.ilsp.gr/META-XMLSchema/v3.0/) and supported by the MS platform (i.e. editor and browser) due to technical constraints.

To convert your XML files between the two schemas, you can use the following XSL converters:

- from MS to CMDI: metashareToCmdi
- from CMDI to MS: cmdiToMetashare

Given that some technical issues cannot be resolved – see below for a list thereof - you are advised to validate your XML files against the relevant XSD to make sure that the files can be uploaded to the relevant repo.

Main issues/differences resolved:

- The MS schema includes metadata for all resource and media types in the same *resourceInfo* profile; in the CLARIN component registry, this is split into four profiles corresponding to the four resource types: corpus (http://catalog.clarin.eu/ds/ComponentRegistry?item=clarin.eu:cr1:p_13618...), language description (http://catalog.clarin.eu/ds/ComponentRegistry?item=clarin.eu:cr1:p_13618...), lexical/conceptual resource (http://catalog.clarin.eu/ds/ComponentRegistry?item=clarin.eu:cr1:p_13551...) and tool/service (http://catalog.clarin.eu/ds/ComponentRegistry?item=clarin.eu:cr1:p_13609...).
- The MS schema includes an *actorInfo* component which is used as a typing component for entities such as annotators, validators etc., where there can be a choice between *person* and *organization*; in the Component Registry, these are split into two components, e.g. *annotatorPerson* and *annotatorOrganization*, both of which are optional so as to cater for the choice between the two.
- The same solution has been adopted for the choice between *structured* (bibtex-like bibliographic references) and *unstructured documents*.
- In the MS schema, certain components are used as "types", i.e. the same component is used with different names: e.g. *sizeInfo* is used for *sizePerDomain, sizePerLanguage* etc. In this case, the CMDI-MS implementation includes a new component for each of these with the addition of an element *role* which takes as value the name of the desired component: e.g. the *validator* component includes the element *role* with the value *validator*.

---

1 Another version of the MS schema, namely the "minimal" v3.0 (i.e. mainly mandatory components and elements) has also been uploaded by the Centre for Language Research Infrastructure (resp.: Josef Misutka).

- In the CMDI, all elements must appear before components while the MS schema has a mixed ordering of elements and components to reflect the order used also in the MS platform; the converters take care of the proper ordering for each version.

Main issues/differences that cannot be resolved and will appear as errors at the validation stage:

- The *validationReport* in the MS schema is optional but not repeatable; in the CMDI implementation, it can be repeated. The validation against the XSD will spot the error.
- For multilingual elements, different attributes are used, namely "xs:language" in MS vs. "xml:lang" in CMDI.
- For some of the multilingual elements (e.g. *resourceName*) the MS schema includes a further uniqueness constraint, allowing their repeatability only if the element is used for different language text; this constraint could not be reproduced in the Component Registry.
- In the MS schema, the length of the free text elements is controlled; no such constraint has been used in the CMDI-MS version.
- The *characterEncoding* element in the original MS-version includes a long list of values which has not been reproduced in the CMDI version.
- Some XML types are not allowed in the CMDI, and have thus been replaced as follows:
    - xs:double used for the element *perplexity* has been replaced by xs:string
    - xs:integer used for various elements (e.g. *samplingRate*, *numberOfTracks* etc.) has been replaced by xs:int.