Report on integration efforts easyDMP - DSW

Background

This work was performed by ELIXIR-Norway and Sigma2 in the context of BioMedData (RCN INFRA 295932), the ELIXIR2 (RCN INFRA 270068), the Centre for Digital Life Norway II (RCN 320911) and the EOSC-Nordic (EU H2020 857652) project, enabled by an advance user support agreement.

The original ambition was to enable two use cases:

1. Import of information provided by the user as answers to questions in different templates from one tool to another through an API to enable downstream processing
2. Import of templates/questionnaires from on tool to another

While use case 1 could be enabled through this exploration, we encountered major challenges on use case 2 which were not solvable within the frame of this work.

Observations on DSW

- See “Tree conversion issues”.
- The JSON export is a record of the changes made to a knowledge model and not the questionnaire itself. The other export possibilities represent mainly the answers given.

Observations on EasyDMP

- There’s a dedicated export-format that attempts to hide some complexity from the importer.
- Internally, everything in EasyDMP uses instance-specific integer IDs, these IDs are exposed in the export, this can lead to issues e.g. for IDs for newly imported questionnaires.

Exchange of DMPs using custom jinja2 templates

As an initial approach we tried to lever the DSW export templates to perform the import.

DSW provides customizable jinja2 templates for export to different formats. By manually mapping the information from a DSW knowledge model (KM) to an easyDMP template the DMP resulting from the DSW KM can be imported to easyDMP through the import API endpoint.

Upon changes on the DSW KM or import to different templates in easyDMP the mapping has to be adapted. This creates a large scalability and maintenance issue.

One factor of special consideration are the running number IDs for plans, questions and templates in easyDMP.

Exchange of DMPs using RDA DCS

This is working.

EasyDMP has API endpoints for exporting and importing to/from RDA DCS format as well as web ui support for the same. Users must be authenticated in order to import, and to export private plans. Public plans are accessible without authentication.

Through the submission function DSW can POST to the easyDMP import API endpoint. Using the RDA DCS export template the DMP is imported to easyDMP. This also works through manual export of a RDA DCS JSON and upload through the easyDMP UI.

DSW provides both a functional UI and API to import content from a RDA DSC JSON to an existing questionnaire in DSW. The mapping is handled by an import template.

Observations/Limitations on RDA DCS

Missing information

RDA DCS cannot store arbitrary text answering questions like **how** to do something.

The fields that are typed, like handles/persistent id systems for instance persons and metadata do have an “other” value but there is no process to add more standardized values, or select a scientific field specific subset (FIPs).

RDA DCS has no semantically backed field to express the types of data sets, beyond their publication status. This can be crucial for the evaluation of re-identification of individuals from the data.

Privacy, data sensitivity and ethical issues are represented only on a binary basis in a machine readable way, to allow the RDA DCS representation to be a basis for decision making. More detailed explicit statement on this issues beyond plain text is desirable, eg. on the need to carry out DPIAs, fulfillment of GDPR or other relevant laws.

Not all objects are safely editable

While many subobjects in RDA DCS have an id-field, some, like project and cost, do not. They are separated from each other solely by which order they are in the list of projects/cost. This means that two or more systems cannot update these subobjects at the same time as the order of the sub-objects may change. If a subobject is deleted, the sub-objects that follow get a new position in the list.

Implicit meaning of specific attributes

The most glaring example of a single field with implicit meaning is license's start_date. To quote: "If date is set in the future, it indicates embargo period." The meaning of this field then changes as time passes, and it is necessary to check for today's date in order to understand the meaning of this field. A separate field, or a renaming of the field, would have been better.

The meaning of subobjects depends on their placement in the tree. For instance, a dataset may have multiple distributions, and the distribution data are contained within the dataset object. This makes it harder to convert to and from a system with a more flat hierarchy, where the distribution has a pointer to a dataset.

The identity of some subobjects also depends on their chronological placement in the tree as they lack an explicit id-field.

Tree conversion issues on exchange of questionnaires

Incompatibility of the DSW and easyDMP tree structures

Three factors contribute to the incompatibility of the DSW and easyDMP questionnaire formats:

1. The DSW knowledge model JSON is designed to represent the performed edits and requires to be processed in order to allow import to easyDMP.
2. The tree structure of the DSW questionnaire follows extensively branching and disjointed trees which is in opposition to the unidirectional flow of the easyDMP format. This makes it difficult to automate the mapping of the DSW questionnaires to that of easyDMP.
3. Differences in type annotations for various fields within the questionnaire and their general ambiguity: EasyDMP uses a stricter format and requires type assignment of the various fields.

While manual conversion from one questionnaire to the other is possible, it is not a scalable solution as new iterations / version updates on either of the questionnaires could effectively break any mapping generated.

The JSON document export form the DSW lacks strict type annotations or any marker for repeatable questions in the DMP export. This ambiguity compounds with the large number of questions presented in the questionnaire. This forces a manual case-by-case analysis of the questions to determine the exact mapping to the easyDMP structure.

References:

easydmp - GitHub

Data Stewardship Wizard · GitHub

DSW Registry

RDA DMP Common Standard for machine-actionable Data Management Plans