Progress on Data Stewardship Wizard during BioHackathon Europe 2020

Marek Suchánek¹, Rob Hooft², and Konogan Bourhy³

¹ Faculty of Information Technology, Czech Technical University in Prague, Thákurova 9, Prague, Czech Republic ² Dutch Techcentre for Life Sciences, Jaarbeursplein 6, Utrecht, The Netherlands ³ Centre National de la Recherche Scientifique, IFB-CORE team, 1 Avenue de la Terrasse, Gif-sur-Yvette, France

BioHackathon series:
[BioHackathon Europe 2020](#)
Virtual conference 2020

Submitted: 01 Dec 2020

License
Authors retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Published by [BioHackrXiv.org](#)

Abstract

We used the Virtual BioHackathon Europe 2020 to work on a number of projects for improvement of the data stewardship wizard: (a) We made first steps to analysis of what is needed to make all questions and answers machine actionable (b) We worked on supporting the Horizon 2020 Data Management Plan Template (c) Several new integrations were made, e.g. to ROR and Wikidata (d) we made a draft plan for supporting multiple languages and (e) we implemented many suggestions for improvement of the knowledge model that had been suggested to us over the past time. Quickly after the biohackathon, the adapted knowledge model, new integrations and the H2020 template have been made available to all users of the wizard.

Introduction

Data Stewardship Wizard (DSW, [ds-wizard.org](#)) is a flexible tool that helps data stewards and researchers in building data management plans together (Pergl, Hooft, Suchánek, Knaisl, & Slifka, 2019). DSW is being actively developed and new features are regularly included according to the current needs of users. In terms of the content, direct collaboration with researchers and data stewards is necessary. Currently, it is possible to build DMP templates to any output format, as well as to create or to customize questionnaires according to the needs of any field of science or institutional regulations. Moreover, integrations with other services using APIs to help researchers filling answers is just a question of configuration. As we summarize in this report, during the BioHackathon Europe 2020, we strived to outline how to increase machine-actionability beyond RDA DMP Common Standard (Miksa, Walk, & Neish, 2020) (Cardoso, Proença, & Borbinha, 2020), to incorporate new integrations for answer suggestions, and to enhance our data stewardship knowledge model to further help researchers in the future. As part of the work we also implemented the Horizon 2020 DMP template and updated maDMP template according to the latest release of the standard.

The Data Stewardship Wizard (DSW) supports data stewardship planning with a focus on benefits for research projects. As data management planning tools usually target composing a data management plan (DMP) primarily for funders, DSW considers producing a DMP as a secondary goal. Guidance in DSW is provided in several forms: tree-like (conditional) questionnaires, references, suggested answers (closed or from integrated registry such as [FAIRsharing.org](#) (Sansone et al., 2019)), or advice. A DMP can be then exported from DSW as a document that follows a funder template, so that the researcher can also use DSW to fulfill the obligation of making that document.

Finally, we would like to make DSW friendly to users that do not use English as their primary language.

All of those topics were addressed during the BioHackathon Europe 2020 and are described in this report.

Machine-Actionable DMPs

Many people in data support roles want to use information that is collected in DMPs to support their own tasks. To make it possible to process many DMPs efficiently, there is a movement to make these machine-actionable. For some core information that is done by the RDA DCS, but we would like to proceed even further by making DMPs fully RDF-encoded using existing ontologies and vocabularies. A related issue is also to incorporate new integrations for answers to following the ideas of Linked Data that are essential when working with RDF.

Our primary goal for this BioHackathon was to enhance machine-actionability of DMPs produced from DSW. The idea was to use existing ontologies and vocabularies related to the domain of data management and write an export template that would be creating RDF documents. We started by gathering the ontologies and vocabularies with participants from other projects of the BioHackathon (including colleagues from EDAM ontology (Ison et al., 2013)):

- [RDA DMP Common Standard](#) (already implemented in DSW)
- [DUO Data Use Ontology](#) (data use requirements and limitations)
- [ADA-M](#) (not an ontology nor vocabulary)
- [Linked Data Glossary](#) (turned not usable)
- [DCAT](#) (related to RDA DCS)
- [PROV-O](#) (provenance)
- [Data Management Terminology / Glossary](#)
- [Data Collection Ontology](#)
- [HROntology](#) (expertise and contributors)

In the end, we encountered a problem that there are many ontologies and vocabularies that tackle specific parts of data management, but we are not able to use it to produce a sensible RDF from DSW. After a discussion with participants from the EDAM and Data Stewardship Curriculum project, we decided to pursue developing a suitable data management ontology/vocabulary in the future as part of the ELIXIR CONVERGE project, where others (like people working on data management training) will also be able to use it.

Currently, we support machine-actionable DMPs according to RDA DMP Common Standard (and related DCS Ontology) in JSON and various RDF formats. During the BioHackathon, a new version of this standard was released, and so we updated our export template according to the changes.

Horizon 2020 DMP Template

A template for a DMP is made in the Wizard using the well-known Jinja2 templating language (Pallets, 2020) which allows significant flexibility. One can create a template for practically any text-based format, e.g., HTML, RDF, Markdown, etc. Furthermore, DSW allows automatic transformations between formats using [Pandoc](#) or [rdflib](#). Prior to the BioHackathon Europe 2020, we supported export in RDA DMP Common Standard (v1.0) and according to the Science Europe DMP template; and we had users request the support for Horizon 2020.

The [Horizon 2020 DMP template](#) is used for EU H2020 projects, but other project calls accept DMPs according to this template as well. Moreover, it is highly related to FAIR data principles which we also stress in DSW.

In order to build the template we first had to specify the mapping between our knowledge model and questions that are in the H2020 form. Experiences from the previous BioHackathon

where we implemented the Science Europe template were very helpful. During the mapping process, several new questions were added to the knowledge model in the Wizard, and others were slightly improved or moved to different places in the knowledge model.

With the mapping prepared, we developed the export template to produce an HTML document (in DSW convertible to PDF, docx, LaTeX, etc.) using Jinja2 templating language. The development was highly simplified, thanks to two things:

1. DSW now provides a [Template Development Kit](#) (TDK) to manipulate templates so you can quickly try it on-the-fly as you develop without any need to package and upload the template to a DSW instance manually.
2. A lot of questions were (naturally) overlapping with the Science Europe template so we could efficiently re-use fragments from our previous work.

The template is now released using the DSW Registry together with updated knowledge models and other templates. Users can easily migrate their projects to this newer version and then export DMP with Horizon 2020 template as well as Science Europe template or maDMP template just by clicking a button using the very same questionnaire. We also made this template available in our own hosted DSW instances.

New Integrations for Suggestions

During the changes in the knowledge model based on needs for Horizon 2020 as well as user feedback, we also identified the need for new integrations to suggest answers where we previously asked for free-format text. In addition to FAIRsharing.org that we already use for querying curated lists of standards, policies, and databases, and CrossRef for funders, we wanted to suggest both licenses and scientific organizations (for contributor affiliation).

To support *organizations*, we added an integration of Research Organizations Registry ([ROR.org](#), (Lammey, 2020)) that has a public API suitable for integration. Therefore, the implementation was very straightforward, just setting the request and response details in the knowledge model editor.

Surprisingly, we had a hard time to find a suitable service that provides a JSON list of *licenses* with the possibility of filtering. There is a good list <https://opendefinition.org/licenses/api/>, but it is a static JSON file rather than API. As a good source of suggestions turned to be well-known Wikidata (Vrandečić, 2012). It has a SPARQL query interface that can be used as an API and returns results as a JSON list. Thanks to the capabilities of SPARQL and the versatility of DSW, we managed to create an integration that queries instances of ([wdt:P31](#)) specific class passed as a parameter. In this case, we used [wd:Q79719](#) for licenses. The case-insensitive filtering is done by SPARQL FILTER with a standard label and description.

Furthermore, we investigated other APIs for future use. For [identifiers.org](#) as a registry of identifiers and their providers, we prototyped integration for both resources and institutions. Then we prototyped also [BioPortal](#) integration for querying various ontologies and vocabularies that are used mainly in life sciences, but this API requires the use of API key. And finally, we also contacted [RAiD](#) (Resource and Activity Persistent identifier) to obtain the API key for testing, but they currently do not provide a way to query public projects through an API. We've learned that it is on their roadmap so we will implement it when it is released. We wanted to also integrate with [ORCID](#), but their API is too hard to use (even for public records), and it cannot be currently integrated with DSW. Also, OpenID is implemented differently with ORCID, and the conditions do not allow us to use it in DSW.

In order to support full machine-actionability in the future, we would like to have as many integrations as possible. Open “text” answers are very difficult to process automatically. A question that is answered using an integration associates the text with a hopefully persistent URL, which can be used to make the DMP available as machine actionable RDF. The best

example is the Wikidata integration, where the URIs are persistent and already provide machine-readable content in RDF as well as human-readable web pages.

Multi-level Internationalization

As the Data Stewardship Wizard is starting to be used in many countries across but also beyond Europe (e.g. in Canada and several African countries), internationalization is becoming more and more desired. Although people usually want to produce a DMP in English for a funder, questions and guidance could be more comfortable in their native language (or another preferred one). There are standard methods and tools for translating software, including generically i18n (*internationalization*, abbreviated by replacing the central 18 letters by the digits “18”), l10n (*localization*), and an implementation in GNU *gettext* (Drepper, Meyering, Pinard, & Haible, 2020). For several programming languages, there are packages to support those standards and offer means for translations and localizations. However, in DSW, it is quite complicated to do a translation because there are multiple levels where text is stored and needs to be translated. Also, it must be solved consistently and without unnecessary burden for users that do not want to use this feature. We had a brainstorming session where we designed a possible architecture and planned how to internationalize DSW (see Figure 1).

Translation of the DSW client (its user interface) is very straightforward as there is already the concept of localizations as a key-value dictionary. Simply said, all messages that users see from the UI are already separated from the code, and the file can be provided to the client from the server. This has already been used to, for example, rename “Knowledge Model” to “CRF Templates” for *VODAN-in-a-Box* setup where DSW serves as a form tool for filing case report forms. In the future, when a user visits the DSW, based on HTTP headers, the client application will be able to request translated messages from the server. To support the translation, the server can store standard PO-files (“portable object” files) used to store translation information in its database (GridFS) and use them with *gettext* to translate the messages. This could either be done every time text is rendered, or once when the PO file is loaded. PO files could be manageable from the administration settings in the client so that data stewards can manage the user interface translations available to their users.

Figure 1: Schema of localizing content with DSW using GNU gettext.

Similar PO files management would need to be done not only for the *user interface*, but also for each *knowledge model*. This requires that we store information about the base language for the knowledge model (not necessarily English). Then each PO file associated with the knowledge model can provide a translation. When a user opens a questionnaire, the client application requests from the server the translated KM just as for client messages. The same functionality could also be used for the preview mode in the knowledge model editor with the possibility to check a specific translation.

Finally, a *document template* also has a specified language but likely this does not need to be translated, we expect document templates are always distributed in one language and if a version in another language is required, it can be provided as another template (e.g. with the same name but different language-tag). This architecture does not incorporate translations

for user-entered content in questionnaires and in some instance settings including welcome or login messages. For this, the UI could select language per input and provide other languages directly (for example as shown in Figure 2).

2.a.1 Project name

Můj první experiment

☒ Desirable: Before Submitting the Proposal

Figure 2: Localized input field in questionnaire with Czech (cs-CZ) text.

We envision that the translations could also be distributed using the DSW Registry just like knowledge models and templates are at this moment. This plan contains quite a large number of changes that need to be made synchronously, we will carefully consider the implementation process hopefully in the upcoming year.

Conclusions and Future Steps

We abandoned one of our primary goals for the BioHackathon and let it be part of ELIXIR CONVERGE, but we accomplished other important achievements. First, we were the first one who updated our implementation of the new version of the RDA DMP Common Standard (a few hours after its release). Then, we investigated various existing vocabularies and established collaboration with other colleagues in making more concise vocabulary for data management in the future. In terms of integrations, we incorporated ROR.org and Wikidata in the new version of the knowledge model and also prototyped identifiers.org, RAiD, and BioPortal for future use. We implemented and released a document export publicly according to the Horizon 2020 DMP template for our knowledge models that have also been updated with new relevant questions. Last but not least, we outlined how DSW can be internationalized consistently on all its levels. In the future, we will use the outcomes of this BioHackathon to enhance Data Stewardship Wizard further so researchers and their projects can benefit from it even more.

Acknowledgements

This work was done during the BioHackathon Europe 2020 organized by ELIXIR in November 2020. We thank the organizers and fellow participants. DSW development is supported by ELIXIR CZ research infrastructure project (MEYS Grant No: LM2015047) and ELIXIR CONVERGE (Horizon 2020).

References

- Cardoso, J., Proença, D., & Borbinha, J. (2020). Machine-Actionable Data Management Plans: A Knowledge Retrieval Approach to Automate the Assessment of Funders' Requirements. In *European conference on information retrieval* (pp. 118–125). Springer. doi:[10.1007/978-3-030-45442-5_15](https://doi.org/10.1007/978-3-030-45442-5_15)
- Drepper, U., Meyering, J., Pinard, F., & Haible, B. (2020). GNU gettext tools, version 0.21. Retrieved from <https://www.gnu.org/software/gettext/manual/gettext.pdf>
- Ison, J., Kalaš, M., Jonassen, I., Bolser, D., Uludag, M., McWilliam, H., Malone, J., et al. (2013). EDAM: An ontology of bioinformatics operations, types of data and identifiers, topics and formats. *Bioinformatics*, 29(10), 1325–1332. doi:[10.1093/bioinformatics/btt113](https://doi.org/10.1093/bioinformatics/btt113)

- Lamney, R. (2020). Solutions for Identification Problems: A Look at the Research Organization Registry. *Science Editing*, 7(1), 65–69. doi:[10.6087/kcse.192](https://doi.org/10.6087/kcse.192)
- Miksa, T., Walk, P., & Neish, P. (2020). RDA DMP Common Standard for Machine-actionable Data Management Plans. doi:[10.15497/rda00039](https://doi.org/10.15497/rda00039)
- Pallets. (2020). Jinja2 documentation (2.11.x). Retrieved from <https://jinja.palletsprojects.com/en/2.11.x/>
- Pergl, R., Hooft, R., Suchánek, M., Knaisl, V., & Slifka, J. (2019). “Data Stewardship Wizard”: A Tool Bringing Together Researchers, Data Stewards, and Data Experts around Data Management Planning. *Data Science Journal*, 18(1). doi:[10.5334/dsj-2019-059](https://doi.org/10.5334/dsj-2019-059)
- Sansone, S.-A., McQuilton, P., Rocca-Serra, P., Gonzalez-Beltran, A., Izzo, M., Lister, A. L., & Thurston, M. (2019). FAIRsharing as a Community Approach to Standards, Repositories and Policies. *Nature biotechnology*, 37(4), 358–367. doi:[10.1038/s41587-019-0080-8](https://doi.org/10.1038/s41587-019-0080-8)
- Vrandečić, D. (2012). Wikidata: A New Platform for Collaborative Data Collection. In *Proceedings of the 21st International Conference on World Wide Web* (pp. 1063–1064). doi:[10.1145/2187980.2188242](https://doi.org/10.1145/2187980.2188242)