DS Wizard Meets DAISY: A Romance Solving Data Protection Requirements in Data Management Planning

Marek Suchánek¹, Pinar Alper², Jan Slifka¹, Vilém Děd², Nene Djenaba Barry², Paulette Lieby³, Marko Vidak⁴, and Nadja Zlender⁴

¹ Faculty of Information Technology, Czech Technical University in Prague ² Luxembourg Centre for Systems Biomedicine, University of Luxembourg ³ Institut Français de Bioinformatique (IFB) ⁴ Institute for Biostatistics and Medical Informatics, University of Ljubljana

BioHackathon series:

[BioHackathon Europe 2021](#)

Barcelona, Spain, 2021

[#18 DSW + DAISY](#)

Submitted: 14 Dec 2021

License:

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

Published by [BioHackrXiv.org](#)

Abstract

This report summarises our activities and achievements in integrating the Data Stewardship Wizard (DSW) and Data Information System (DAISY) tools during the ELIXIR BioHackathon Europe 2021. As a data information system for GDPR compliance, DAISY is focused on a single goal – gathering all information required for GDPR accountability of biomedical research projects. On the other hand, DSW is very flexible and can be used beyond data management planning. We worked on the integration between both tools on two fronts. Firstly, we created a new Knowledge Model in DSW together with a document output template to be able to generate a data protection impact assessment (DPIA). Secondly, we introduced a new integration type between projects in DSW and DAISY that allows the querying of DAISY data upon document generation in DSW. Both of these independent activities brought successful results that were refined and published after the actual BioHackathon. Finally, we provide the related materials as an on-demand training course in the ELIXIR eLearning Platform.

Introduction

The GDPR requires research projects with sensitive human data to perform a data protection impact assessment (DPIA) for documenting the project's data protection risks and corresponding safeguards. Data stewards across Europe are tasked to support researchers with DPIAs, which occur commonly in tandem with data management planning. Two ELIXIR tools fall in the data protection realm. [Data Stewardship Wizard \(DSW\)](#) (Pergl et al., 2019), which for now is used mainly to help in data management planning, raises awareness for data protection requirements such as the DPIA. However, it is not specialised in DPIA reporting. The [Data Information System \(DAISY\)](#) (Becker et al., 2019), which allows institutions to keep a register of their projects using sensitive data, stores structured information on the project's GDPR-relevant aspects – the crucial input to a DPIA. Meanwhile, DAISY lacks the means to combine the project facts into the narrative response needed in a DPIA.

As the DSW and DAISY are highly complementary, we decided to integrate the two to support DPIAs in DSW. The integration has been designed in two independent tasks:

1. **DPIA in DSW (Content-Based Integration):** Allows the creation of a DPIA directly in DSW by introducing a new Knowledge Model (questionnaire structure) and a corresponding document output template. Thus, users may create a DPIA project in DSW, fill in the information, and generate a DPIA document. Also, working with DSW, users will benefit from all its intrinsic features like version history, collaboration, comments, and others.
2. **Querying DAISY Data in DSW (Technical Integration):** Enables linking to a project in DAISY from within a project in DSW and so querying the DAISY project details using an API when necessary. The goal is to benefit from DAISY GDPR-specific features and

allow users to use those when generating documents in DSW. For example, one can then add to the data management plan (DMP) an appendix that exclusively deals with GDPR compliance.

To support and disseminate the use of our integration work, we also planned to compose a brief on-demand training course in the [ELIXIR eLearning Platform](#). Training on DPIAs has already been identified as a gap by the ELIXIR Training Platform.

DPIA in DSW

Writing a data protection impact assessment for a scientific project is a challenging task. Input from multiple key groups is required to obtain complete and precise information. In addition to data stewards, which are commonly in charge of the information collection since their expertise overlaps all other concerned domains, the process involves researchers themselves bringing their own knowledge about the data under consideration and the related scientific processes. Data managers and IT staff who are knowledgeable about the local IT infrastructure and external collaborators may also report on the shared data processing tasks. Finally, legal representatives (e.g. data protection officers) validate the resulting document using existing contracts and institutional policies.

A DPIA is commonly written using a spreadsheet or text-based templates with questions and predefined answers. This approach suffers from the absence of branching logic, overwhelming and poorly structured content, and a lack of support for collaborative editing. Furthermore, subsequent updates of the template make the whole process laborious and unnecessarily complex. Dedicated tools (e.g. [Processing Impact Assessment](#) from the Commission Nationale de l'Informatique et des Libertés – CNIL) are also available, and they do bring substantial help, but in general, they lack the ability of full customisation of the template.

DSW offers a variety of needed functionalities for DPIA document generation. It provides user access and project visibility control, which are necessary features since the DPIA content is often confidential. DSW supports the creation of a versioned structured questionnaire with complex branching logic. Questions can be interlinked, accompanied by extensive explanations, hints, predefined answers and guidance; they may be enriched by links to external sources (e.g. specific GDPR content) or integrations to local or external vocabularies. A smooth collaboration is possible thanks to the ability to add comments on each question.

Finally, as both data management plan and data protection impact assessment are usually composed by the same group of people at the same phase of the research project, using DSW for DPIA generation is very convenient as users can work in only one familiar environment.

Creating a DPIA Knowledge Model

The DPIA Knowledge Model is mainly inspired by several existing documents, namely the [CNIL DPIA document template](#), the DPIA document template used by the Luxembourg Centre for Systems Biomedicine and the [ICO guidance](#) on data protection.

The first chapter collects administrative information, and description of the method(s) followed to establish the project's DPIA. The following chapters contain questions related to:

- The identification of the need for a DPIA
- The nature, scope, context and purposes of the processing of data, as well as the list of related responsibilities
- The data processes and supporting assets
- The proportionality and necessity of the processing
- The measures taken to protect the rights of the subjects
- The identification, analysis, and evaluation of the data risks factors and the measures and controls to mitigate those risks

Although a risk assessment is a very important part of the DPIA, our Knowledge Model is addressing it only very briefly by questions on general controls which are or will be in place. As risks generally depend on the infrastructure used and the institutional units involved, they would apply globally to all the processing under their authority. We propose a process where the risk assessment matrix can be generated by in-house or external professionals using dedicated tools (e.g. [Monarc](#)) and re-used for each DPIA as a resource. For this purpose, our Knowledge Model includes a question to provide a link to such a document.

Lastly, our Knowledge Model was designed with user experience in mind. Complex branching logic, question description and guidance allow users to fill the questionnaire without additional help.

The final version of the DPIA Knowledge Model will be released through the [DSW Registry](#).

DPIA Document Template

We used the [Template Development Kit \(TDK\)](#) provided by DSW for the creation and the customisation of the template; recommendations and hints can be found [here](#).

As a proof of concept, we have implemented [DSW template](#), which produces a document matching the [CNIL Template](#). This template was chosen for its completeness and wide user base. The complexity of the Knowledge Model ensures the information required by other existing institutional DPIA templates is collected, and new DSW templates can be implemented by the community.

Querying DAISY Data in DSW

The other independent approach that we decided to implement uses linking a DAISY project from DSW and then querying data using DAISY API when needed. The overall idea is shown in Figure 1. A user of DSW works on its project (with a goal to get a DMP but not limited to it) and in the questionnaire selects using an integration question one of the projects in DAISY. Now, the project from DAISY is linked via its link stored as a reply in the DSW project. When a user wants to produce a document, e.g., a DMP, DPIA or another report, current DAISY project data are queried using DAISY REST API and transformed into the desired format. The user gets a document with structured data from the DAISY project appropriately included within the other data originated directly from replies in DSW.

Figure 1: Design of DSW-DAISY Integration

With such an approach, data in DAISY can be independently changed without modifying the DSW project: only at the time of document creation is the updated DAISY data queried and included in the document. Thus, document creation is a snapshot of both DSW and DAISY projects at the time of the creation. To implement this feature, we first needed to investigate how the DSW API call functionality, also called 'integration question', can be put into practice and extended to allow project selection in DAISY. We then had to solve how to format the DAISY data with a DSW document template that uses [Jinja2 language](#).

Integration Question for DAISY Projects

DSW supports integration questions since v1.6.0 (April 2019). It allows to query a REST API for type hints – as users type a reply, answers are suggested by the API response from which one specific answer can be chosen. The advantage is the configurability of the integrations and the stored ID of the selected item. On the other hand, it does not allow any form of user authentication towards the external API nor more complex item selection than just a regular input dropdown. In Figure 2, you can see how the integration works for selecting affiliation from [Research Organization Registry \(ROR\)](#).

Figure 2: Integration question for affiliation using ROR.org

We attempted to use this integration feature as is to query DAISY projects. However, it has two pitfalls:

1. To access the API, a secret token must be used for authorisation. Everyone wanting to have such a question (or whole Knowledge Model) in their DSW instance would need to acquire such a token and place it into a server configuration file.
2. As there is no way to distinguish specific users, the call response will consist of all

(published) projects from DAISY. The natural behaviour – displaying only projects related to the user filling the questionnaire – is not achievable.

This is obviously not practical, and one must thus overcome the need for authentication. To overcome these two issues, we designed a new type of integration question described in the following subsection.

DSW Widget Integrations

Although the integration question was a good start, we needed some improvements. The main issue was that the standard integration question in DSW connects to a public API of an external service to find the results. However, in DAISY, users need to log in first to access their projects. So the standard solution was not suitable unless all of the projects in DAISY were public, which would not be desirable.

Therefore, we came up with the idea of an integration widget. Instead of a search field in DSW, we open a popup window with a web page implemented on the connected service side, in this case, DAISY. The implementation of the widget itself is the responsibility of the connected service. This means that any necessary functionality, such as login, can be implemented.

We implemented the [DSW Integration Widget SDK](#), a JavaScript library that is a wrapper for sending the selected options back to DSW, and corresponding supporting code on the DSW side. The library and DSW uses JavaScript `Window.postMessage()` under the hood for the communication between DSW and the popup window with the widget. Besides sending the answers back to DSW, the library also checks whether the widget was invoked from the allowed DSW instance to avoid leaking information somewhere unwanted.

Once the question is answered using the widget, it works in the same way as the standard integration question regarding how it is saved. We save the name of the project retrieved from DAISY and the URL to the project so it can be used for further processing.

Figure 3: Integration widget question in DSW – empty and with a selected answer from DAISY

Widget for DAISY Projects

The widget implemented on DAISY's side allows users to select one of the existing projects and send them back as an answer to DSW. When the widget is opened in a popup window, users are asked to log into DAISY. As the initial page is the actual DAISY login form, the DAISY login credentials are required. Once logged in, users are redirected to the `/integrations/dsw/list-projects` DAISY endpoint presenting a list of projects they have access to. The list contains only records for which users are appointed as local custodians or personnel. The internal DAISY project ID, the project's acronym and the full title are presented. When a project of interest is clicked on, the popup window is closed, and the information is sent to DSW.

Figure 4: Integration widget in DAISY showing a list of user's projects

Retrieving Data for Document Generation

To allow for retrieving of project data from DAISY using its REST API, we had to allow making HTTP requests from Jinja2 templates. This is now a new feature of the [DSW Document Worker](#) component. As it is written in [Python](#) programming language, we decided to use the well-known [Requests library](#). Because any data steward can upload their own template and creating requests might be harmful, we also designed a configuration for the document worker that can enable this feature for specific templates (based on their IDs), set limit and timeout for the HTTP requests made from Jinja2.

To make limits and timeouts effective, we had to introduce a simple wrapper class around the requests library. An object of such a class is then passed as a variable into a Jinja2 template if the feature is enabled. On each request through the object, a counter is increased and checked against the limit. The configured timeout is passed to the call of Requests library functions. It does not support sessions as it is expected that just a single request (or a few consecutive requests) will be made to optimise document generation speed. An HTTP request can be made from a Jinja2 template in the way shown in Figure 5.

```
{%- set response = requests.get("https://example.api?param=" ~ myVar) -%}

{%- if response.ok -%}
    Some value from response: {{ response.json()["someField"] }}
{%- else -%}
    Error occured while contacting external API...
{%- endif -%}
```

Figure 5: Example HTTP request in Jinja2 template

DPIA Appendix Template using DAISY

Using the HTTP requests enabled in document templates, we implemented the [DPIA Appendix Template](#) that can be easily re-used in other DSW document templates. It is based on the example Knowledge Model where there is only a single question that asks for the DAISY project using the widget integration. However, the template is designed in a way that its central part is a re-usable Jinja2 macro. Anyone can take this part of the template, include it in their own one and call this macro using the DAISY project identifier as an argument.

The macro itself makes three HTTP requests to the DAISY API: to retrieve project details, datasets for the project, and contracts for the project. If all of these three requests are successful, the DPIA appendix is rendered as a structured set of tables. Again, the rendering of separate tables is done using macros, so if someone needs to change the rendering, they can adjust some of the macros in the appendix template. In the end, one may use just the main macro for retrieving data from DAISY and write the whole rendering part on its own. All macros and parts of the appendix template are documented both in README and using Jinja2 comments.

Training Materials in EeLP

As a part of our work, we also created a new on-demand course in the [ELIXIR-SI eLearning Platform](#). The course contains basic information about the outcomes of the BioHackathon and how to use them. It also summarises some critical aspects of the related topics such as Data Stewardship, GDPR Compliance, Data Management Plans, or Data Protection Impact Assessment. We plan to extend the course in the future once the results (of tasks 1 and 2) are polished and recommended for wide use.

The EeLP is based on the well-known [Moodle](#) learning management system. As such, it allows to improve the content continuously and add different types of activities and resources. We also plan to include a quiz or feedback form for better user engagement in the future.

Conclusions and Future Steps

We completed the two tasks that we set initially. The DPIA Knowledge Model in DSW, together with the related [document template](#), can be used to create a project in DSW and answer the questions in order to produce the DPIA document. Both of these deliverables will need final polishing, e.g., styling the document template; they can then be published. We plan to publish them on GitHub (the template) as well as using [DSW Registry](#) service so anyone

can easily pull it to their own DSW instance. In this way, both can be easily updated and maintained over time, and others will quickly get the latest versions.

The second task brought new features to DSW. The integration question with a widget still needs some adjustments on the backend side of DSW (making it a new type of integration question); however, it is fully operational. The feature enabling HTTP requests from the document templates is finalised, and it may be enhanced in the future if needed. Both of these features were released in version 3.6.0 of DSW. The widget implementation on the DAISY side also became directly part of its codebase, and [DSW Integration Widget SDK](#) has been published on [npmjs.com](#). Finally, the DPIA appendix template is ready to be (re-)used and further developed as any other DSW document template.

The integration task achieved in this project allowed us to extend the DAISY API and expose more of the project information in the DAISY database. The resulting DAISY API is re-usable not just for DPIAs but also to for DMPs or any other project documentation that shall include up-to-date factual information on the project.

We observe that the resulting tool integration and knowledge model is a candidate best practice for addressing data protection and GDPR accountability in biomedical research projects. As the next step towards validation of our approach, we will be linking the DPIA Knowledge Model in DSW to the Data Protection guidance in [ELIXIR's RDMkit](#). We will continue to improve the knowledge model based on the feedback of the RDMkit users and the wider community.

Acknowledgements

This work was done during the BioHackathon Europe 2021 organised by ELIXIR in November 2021. We thank the organisers and fellow participants. The development and operation of DSW are supported by ELIXIR CZ research infrastructure (MEYS Grant No. LM2018131) and ELIXIR-CONVERGE (H2020 No. 871075). The contribution of ELIXIR LU is partially funded by the Luxembourg Ministry of Higher Education and Research.

References

- Becker, R., Alper, P., Grouès, V., Munoz, S., Jarosz, Y., Lebioda, J., Rege, K., Trefois, C., Satagopam, V., & Schneider, R. (2019). DAISY: A Data Information System for accountability under the General Data Protection Regulation. *GigaScience*, 8(12). <https://doi.org/10.1093/gigascience/giz140>
- Pergl, R., Hooft, R. W. W., 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, 59. <https://doi.org/10.5334/dsj-2019-059>