Integrating the FAIR infrastructure with EUDAT

Authors: Christine Staiger (SURFsara, NL), Mark Thompson (Leiden Universitair Medisch Centrum, NL), Shamanou van Leeuwen, Luiz Bonino (Dutch Tech Centre for Lifesciences, NL)

Summary

The data pilot set out to investigate how the FAIR infrastructure and EUDAT could profit from each other:

- Which technology can be employed by the FAIR infrastructure building blocks called FAIR Data Points?
- Which functionalities and technology from the FAIR Data Points can EUDAT use and profit from?

The overall goal was to give general advice for both infrastructures.

During the course of the data pilot EUDAT and FAIR gained more insight into each others infrastructure. In this document we present some plans how both infrastructures can profit from integration and how integration can be reached.

Introduction

EUDAT is a European initiative that devises technology and services to support the management of scientific data throughout the whole research process and beyond. It provides services for replicating (B2SAFE), publishing (B2SHARE), sharing and versioning (B2DROP) and searching (B2FIND) data. Furthermore, there are services for ad-hoc data transfers (B2STAGE), authentication and authorisation (B2ACCESS) and for labeling data with persistent identifiers (B2HANDLE). The infrastructure in which these services are embedded is by its nature distributed across several data, HPC centres and community centres. Together, they develop, deploy and maintain the services and infrastructure.

The FAIR Principles[1] provide an implementation-independent, precise and measurable set of qualities for the publication of scientific research data. Following the FAIR Principles, data is easily discovered, easily evaluated, and maximally reusable for both human agents and automatic (software) agents. DTL is currently leading a development effort to implement a FAIR Data infrastructure. The development of the FAIR Data infrastructure is supported by DTL and a number of national and European projects, including Elixir, BBMRI, FAIRdict and RDConnect. We note that the DTL FAIR Data infrastructure is aligned with the European Open Science Cloud, and the DTL FAIR Data infrastructure is the first general-purpose implementation of the FAIR principles. The infrastructure is composed of a number of applications that later, combined, will for the Data FAIRport. Currently, these applications are being developed as stand-alone tools that will later be integrated to form the Data FAIRport.

[1] The FAIR Guiding Principles for scientific data management and stewardship, Mark D. Wilkinson, *Nature Scientific Data,* http://www.nature.com/articles/sdata201618

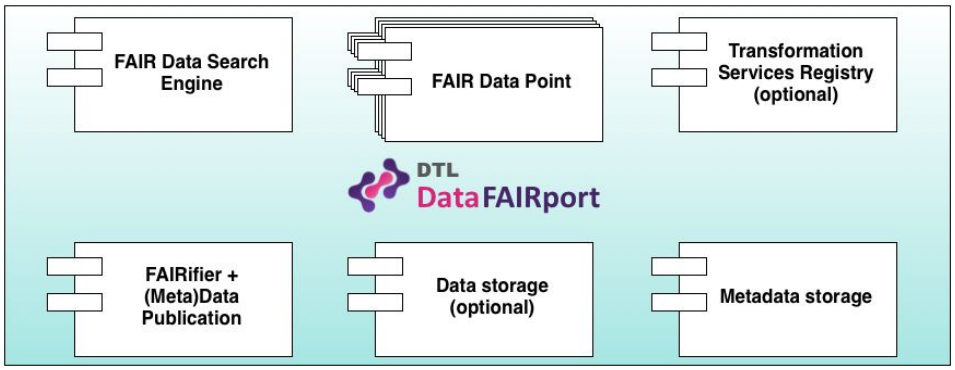

Figure 1: Overview of the Data FAIRport architecture

FAIR data point (FDP)

Within the Data FAIRport infrastructure, the FAIR Data Point (FDP) component is responsible for providing access to data and metadata. The FDP provides metadata at five complementary layers: FDP Metadata, Data Catalog Metadata, Dataset Metadata, Distribution Metadata and Data Record Metadata. The FDP implementation is centered around a RESTful HATEOAS API[2] and is optimized for machine discoverability and processing. (Meta)Data access starts with the FDP URL, which resolves to metadata that describes the FDP and supports further exploration of the FDP by iterating through the different levels of metadata. For example, the metadata includes a link to the next FDP level (catalogs), which in turn links to the Dataset-level descriptions. Metadata is represented using the ubiquitous RDF data model[3], which is a W3C recommended model for data interchange on the Web. Finally, the Dataset metadata contain links to one or multiple Distribution and Data Record metadata.
The FDP metadata specification[4] is currently being finalized and is based on existing standards and vocabularies such as OAI-PMH, Dublin Core, DCAT and LDP. Therefore, the FDP specification is not “yet another standard”, but rather aims to combine existing standards to meet the FAIR objectives. The FDP specification also gives a recommendation for the minimal metadata needed towards dataset findability (F) and it specifies (how) to declare a data license in order to be reusable (R).

FDPs are the part of the FAIR infrastructure that enables users to publish data and metadata. The metadata supports semantic data discovery using Linked Data (RDF) technology, but care is also taken to include “labels” and “descriptions” for metadata, which are useful towards the development of user-level applications. In EUDAT terminology a FDP may be similar to a “metadata catalogue”. Here we list in more detail some of the of metadata that is minimally available at each level of the FDP. There are five levels of metadata:

1) The **FDP** itself contains metadata about the organization or individuals offering the FDP and other “housekeeping” information: subject, title, description, FDP software version

[2] Hypermedia As The Engine Of Application State: https://en.wikipedia.org/wiki/HATEOAS
[3] https://www.w3.org/TR/rdf11-primer
[4] https://dtl-fair.atlassian.net/wiki/display/FDP/FAIR+Data+Point+Software+Specification

and (deployment, update) date/timestamps, UI landing page and finally: resolvable links to the catalogues contained in the FDP.

2) The **catalogues**. Catalogues are are a mechanism to group datasets, e.g. a FDP can contain data on “genomics” or “text mining” as two distinct categories. In addition to various housekeeping metadata, a “theme taxonomy” is specified to refer to a vocabulary or ontology that will be used to classify the “theme” of a Dataset (in the next level of metadata, see point 3). And finally, the metadata lists the Datasets belonging to the catalogue.
3) The **datasets** are annotated with a creator, description, version, identifier and publisher, etc. A description of the content of a Dataset is given both as keywords and with the “theme” property that refers to concepts in the theme taxonomy. Another metadata item may be specified to refer to a Datarecord specification in RML[5], which describes the semantic structure of the data, including the types for data entities and the relations between them. Finally, the metadata contains links to the dataset distributions.
4) The **datasets distributions** specify the different ways in which a dataset is made available: e.g. a dataset can have an html representation accessible through HTTP and/or an RDF distribution that is accessible through a SPARQL endpoint. An accessURL or downloadURL is specified for each distribution, as well as a data license and other necessary information to access (and use) the distribution.
5) And finally the **data record** metadata describes the actual content of the dataset at the record level such as the data types, their domain and range.

B2FIND

B2FIND is EUDAT’s metadata service. It provides the means to search in metadata via facets and retrieve the actual data from the original repository. The metadata is regularly gathered from EUDAT services such as the B2SHARE repository and from different community data providers and scientific repositories. The service, thus, consists of two components: a comprehensive metadata catalogue and a discovery portal to search and access research data.

The metadata is harvested from the data provider endpoints via OAI-PMH and other transfer protocols. After harvesting the metadata keys are mapped to facets in B2FIND and metadata entries are indexed in the catalogue, which is based on CKAN[6]. The CKAN software provides a well-defined restful API and a python API that allows scientists to query the data via a graphical user interface as well as command line interface.

Note, that B2FIND only offers to search through metadata, not the data itself. The user or machine are directed to the data via links in the metadata entry to the original repository. The content of the metadata is defined by the scientific communities and the mapping to the B2FIND facets is usually done in close collaboration with community representatives and the EUDAT

[5] http://rml.io/spec.html
[6] http://ckan.org/

developers. While the B2FIND schema is based on the DataCite schema, B2FIND uses self-defined keys, such as 'Discipline' or others that satisfy community specific needs, as e.g. spatial and temporal searches.

Comparing FDP and EUDAT B2FIND

Both the FDP and B2FIND provide and expose metadata to researchers. Both services do not own or contain the actual data but provide access to the data or a landing page of the data. FDPs and services in the EUDAT domain use different technologies to achieve this functionality. FDPs make use of the linked data approach. The Life Sciences, especially ELIXIR and BBMRI are basing their metadata services upon this technology, while also showing interest in employing and adopting EUDAT services.
Integrating (parts of) the FAIR data infrastructure into the EUDAT infrastructure will help propel the integration of such communities and in general help EUDAT to combine data hosted in its own domain with other data exposed as linked data. In this way EUDAT can leverage the FAIR data initiative and its proposed Data FAIRport architecture as a starting point to adopt Linked Data technology in its CDI. In contrast to many other Linked Data efforts (such as the “Linked Open Data Cloud” and community specific repositories as those offered by EBI), the FAIR Data Point provides a general purpose framework to couple metadata to datasets in both a practical and a semantically sound way. For example, the FAIR Data Point (FDP) prescribes a minimal set of (Linked) metadata to be associated with each dataset and dataset distribution using existing vocabularies and ontologies such as Dublin Core and DCAT. Other Linked Data approaches may mix data and metadata indiscriminately, do not clearly define their resource as a dataset (or other information artifact), or, conversely, over-specify using overly detailed ontologies that are not widely adopted. The fact that the FAIR data approach facilitates integration with EUDAT is illustrated by the quick progress made in our B2FIND/FDP metadata harvesting effort (see details below). In turn, EUDAT’s data ecosystem will be enriched by the growing number of datasets that are and will be made available through a the Data FAIRport/FDP.

Combined infrastructure

General overview

In this section we summarize the results of our evaluation of combining/integrating the FAIRport and EUDAT services. We also show preliminary results and list potential tasks to be addressed as a continuation of the data pilot.

In Figure 2 we show four major opportunities to integrate the FAIR Data Point with the EUDAT services ecosystem in an optimally mutual beneficial way. They are listed here in nonspecific

order, followed by more details in the following subsections. We offer a discussion of priorities in the final section of this document.

1) FDP to B2FIND: The FDP's metadata can be harvested by B2FIND and thus enable searches through a combined metadata catalogue comprising EUDAT metadata and FAIR metadata. Towards this goal, *we have realized an advanced prototype, which is ready for testing with B2FIND.*
2) B2HANDLE for FDPs: FDPs can employ the B2HANDLE service to mint PIDs -stable references- to their location such that users reference and access the source of their data.
   We have evaluated the B2HANDLE service and have made arrangements to adopt PIDs in the FDP.
3) B2SHARE/B2SAFE to FDP: Both the B2SAFE and B2SHARE services can make their metadata accessible as a FDP and and thus expose EUDAT data assets in a FAIR-compliant way.
   We have investigated the avenues for implementation and identified potential bottlenecks. Initial steps forward have been defined.
4) FDPs can employ B2SAFE to preserve and replicate their metadata.
   We have started to evaluate different possibilities. Although there seem to be possibilities that offer definite advantages, feasibility is not yet established and requires further investigation.

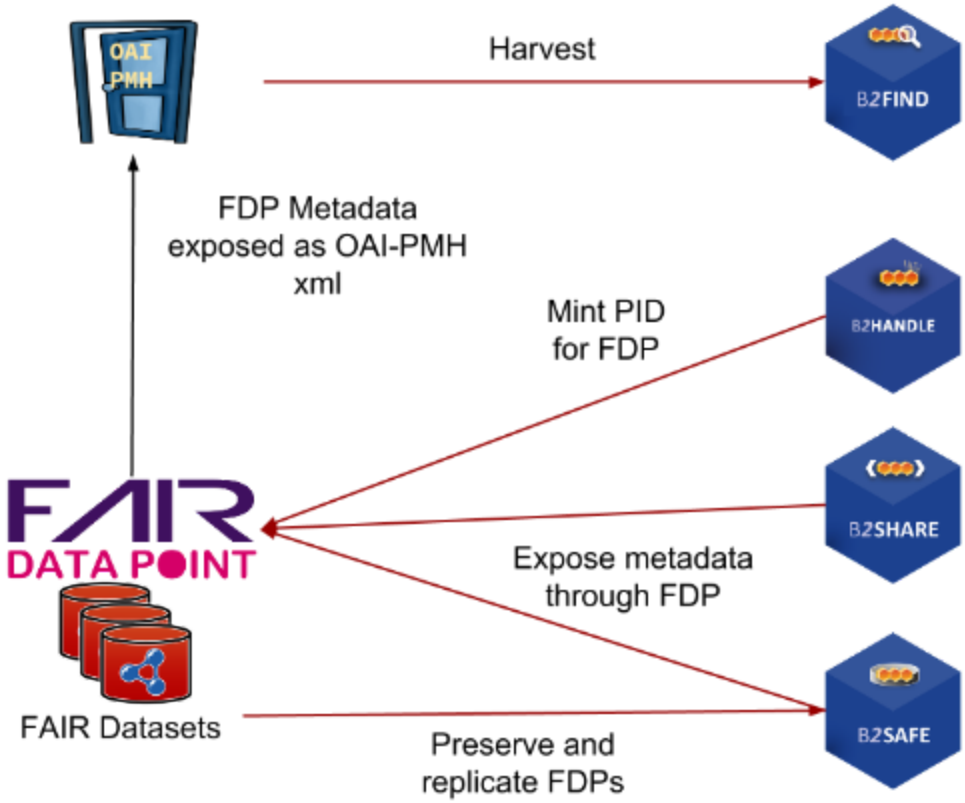

Figure 2. A combined infrastructure. The Figure shows how the FDP and EUDAT's services can be combined. The black arrow indicates work that has already been carried out, red arrows indicate future work.

FDP to B2FIND

The search service shown in the data FAIRport architecture (Figure 1) is designed to provide a way to find datasets across FDPs. The envisioned implementation will index the FDP metadata and expose both keyword and semantic query functionality with both programmatic and user interfaces. In our data pilot we have investigated if and how B2FIND could serve as a search engine for FDPs. Below we will discuss the prototype implementation that was developed in the first phase of the data pilot. Using B2FIND as a FDP search service has the notable advantage that we expose the datasets contained in the FDPs to the EUDAT/B2FIND user community in a way that is already familiar to them, thus increasing the visibility of the growing number of datasets available through FDPs. Moreover, our investigation shows that B2FIND can satisfy most requirements for keyword-based search functionality. However, the FAIRport architecture has the additional aim to make datasets findable to semantically-enabled software agents. Such agents will use the rich, semantic metadata from the FDPs to, for example, automatically evaluate dataset relevance and perform automatic data integration (when possible). We understand that it is possible to extend B2FIND with additional search facets for specific communities, but this will require additional effort per facet to map FDP metadata facets to

newly defined B2FIND facets. Therefore, this approach is not scalable for our purpose: FDPs are not restricted in the number and type of metadata that may be specified by the data owner. Over the course of the pilot we have found that the work from Yann LeFranc towards a "Semantic Resource Lookup Service" in the context of B2NOTE bears close resemblance to our requirements for the FAIRport search engine. We have had several meetings and teleconferences with Yann and a future collaboration in this area between the DTL developers and Yann's team looks promising. For a more detailed description of the considerations, see Section.

Here we give a summary of our efforts to expose FDP metadata through B2FIND. The EUDAT hackathon in Barcelona provided us with the opportunity to interact with the B2FIND developers, which facilitated the creation prototype implementation. Although this implementation is beyond the original scope of our data pilot, we are currently close to a solution that enables FDP metadata harvesting with B2FIND.

The implementation consists of an additional end-point on the FDP that allows a harvester to access the FDP with an OAI-PMH compatible API to retrieve the FDP metadata. Our code is based on an existing OAI-PMH implementation in Java (https://github.com/Dspace/xoai). This code was first remodeled to work with the Java Spring Framework and then merged with the FDP code base.

The overall workflow is as follows:

- accept API call with OAI-PMH verb
- formulate an oai_dc XML formatted response
- retrieve (on-the-fly) the required metadata fields from the FDP and insert in oai_dc XML
- Wrap oai_dc XML in required response OAI-PMH response XML
- send response to the caller

This approach is different from the typical "data provider" approach such as for example offered by jOAI (http://www.dlese.org/oai). In that case a directory of XML files containing the metadata are exposed to the jOAI Data Provider, which then attempts a conversion to the required oai_dc format. In our approach we have no explicit conversion step: the required oai_dc format XML is generated on-the-fly whenever a harvester accesses the FDP. Although there are pros and cons to either approach, we think this approach fits well with the scale at which FDPs are intended to be deployed. Finally we note that this approach gained support from some of the other user community participants at the hackathon. Therefore we could consider generalizing our code to support back-ends other than the FDP and maintain the resulting code base as a collaborative open source project.

On the OAI-PMH enabled FDP, the OAI-PMH “verbs” are available on the end point located at <FDP base URI>/fdp/aoi. Although our initial aim was to just show minimal compatibility with existing AOI-PMH harvesters, we realized that a basic implementation all of the AOI-PMH verbs was required:

- Identify
- ListSets
- ListMetadataFormats
- GetRecord
- ListIdentifiers
- ListRecords

We have checked the correct implementation of each of the verbs in a test setup where two different OAI-PMH compatible harvesters were directed to our FDP. In both cases the harvester was able to cycle through its iteration of verbs and no protocol or oai_dc format syntax errors occurred. The final task, to let B2FIND harvest the FDP metadata, has been postponed to the second phase of the data pilot.

We had some discussion about whether the hierarchical metadata structure of the FDP should be reflected in the "Set" hierarchy of OAI-PMH. This way we could expose for example the FDP notion of a Dataset Catalog which is a container for datasets. However, we did not pursue this further for two reasons: 1) the set information would not be used by our primary harvester (B2FIND) and 2) OAI-PMH does not support any further specification of the semantics of a set. The latter means that the interpretation of the set-relations (why some metadata items are in a set), needs to be "hardcoded" in every application that is consuming the OAI-PMH metadata. In contrast, the hierarchical structure of the FDP is clearly and unambiguously semantically defined through the use of standard ontologies such as DCAT.

At the hackathon, we agreed with Hannes to setup a “FDP community” on B2FIND to serve as an entry-point searching FDPs. This would also allow experimentation with a number of FDP-specific search facets. In case the FDP structure is adopted to expose metadata from other EUDAT services (such as B2SHARE, see Section B2SHARE to FDP), then the separate classification of “FDP community” may become superfluous.

B2SHARE/B2SAFE to FDP

B2SHARE is EUDAT’s data publishing service and provides the functionality to archive so-called long-tail data. Data is uploaded in deposits, which can be mapped to the concept of FDP Dataset. A deposit is always annotated with some metadata which is collected from the user at the time the deposit is uploaded to B2SHARE. The standard EUDAT metadata template and one of its extension template is shown in the Figure 3. We note that the metadata template also forms the basis of the search facets of B2FIND.

In the same way that we previously (FDP to B2FIND) mapped FDP metadata to the B2FIND model, we now would need to offer procedures and technology to enable the inverse in order to expose B2SHARE metadata in a FDP-compliant way: i.e. mapping metadata from the B2FIND template to the FDP metadata model. The B2SHARE metadata template is well-structured and contains a number of metadata fields that can be trivially mapped to the FDP metadata schema.

For the metadata fields like “Discipline”, or the community-specific metadata fields, a small amount of semantic data modeling may be necessary. However, this will not be problematic, since the “types” of the metadata fields are known and the number of fields in a template is finite.
We, however, see challenges in the deduplication of nodes. More specifically, authors do not carry a unique identifier and are merely a free-format string in B2SHARE. The same person could thus appear as different nodes in linked data. This might hinder the effective search for data in both the FDP and actually B2SHARE and thus B2FIND.

B2SAFE is EUDAT’s data storage and replication service. Files can be organised in so-called collections and sub collections in order to create a hierarchy similar to a file-folder hierarchy. Each file will receive a persistent identifier (PID) to identify files and link them to their respective replicas and the PID will resolve to the physical location of the file.
There are two ways of creating metadata in B2SAFE.

1) B2SAFE contains a simple sql-database in which one can store key-value-unit triples that annotate files and collections.
2) Metadata can be stored in separate files and ingested into B2SAFE. B2SAFE has no notion on which files are data files and which files are metadata files. The relationship between data files and metadata files can be established by providing a manifest-file, which is based on a simple METS scheme.

The most notable difference with respect to B2SAFE and B2SHARE with regard to metadata is that B2SHARE offers well-defined metadata templates, whereas metadata in B2SAFE can in principle have any kind of format.

Overall we have found that in the EUDAT infrastructure, metadata is distributed over the different services and offered in different formats and integrated with different technologies. Furthermore it seems that in EUDAT there are already some efforts ongoing to improve the handling of metadata. One of those efforts is utilizing a graph database to store the relation between EUDAT’s digital objects and their respective metadata. There is a striking convergence with the approach of the FDP: both are relying on a graph structure to link datasets to their metadata (albeit using different technology). This points to an interesting opportunity for EUDAT to homogenize their metadata infrastructure by adopting the FDP, because a FDP may be instantiated for all resources, regardless of their various back-end formats, data representations and data storage implementations.

A practical first step towards such a goal would be to perform the mapping of the B2SHARE metadata template to FDP metadata and expose it using a FDP-compatible web service built into B2SHARE. In a second step the same could be done for B2SAFE metadata, although we recognize the challenge given that the B2SAFE metadata is typically not as well-defined. However it is likely that both the “FAIRification” tools in development at DTL and the semantic annotation service from B2NOTE will be able to mitigate such problems. The result of such an effort would be that a large part of EUDAT resources will be uniformly findable and accessible through the simple, yet semantically-enabled interface of the FDP.

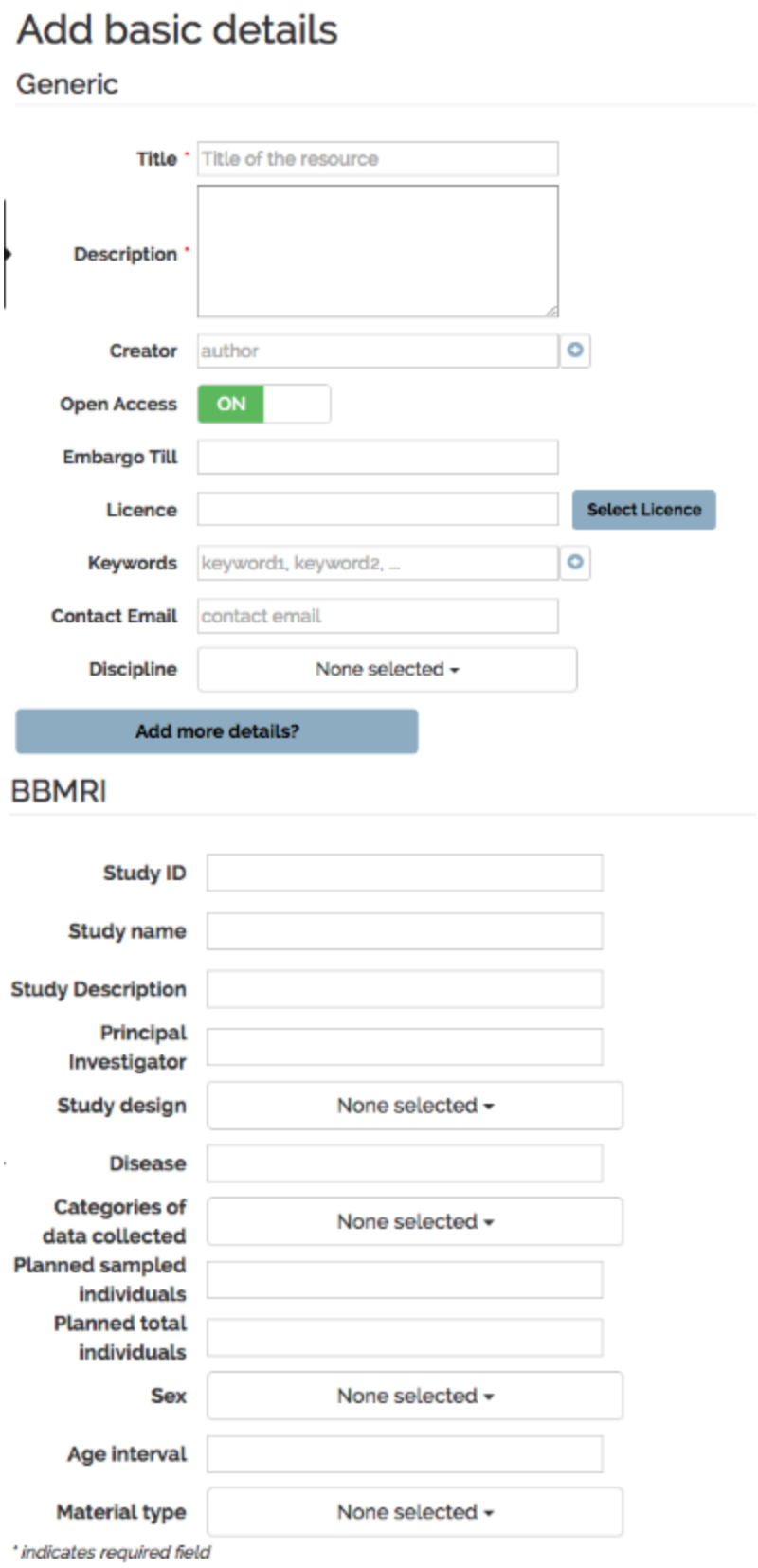

Figure 3. The B2SHARE standard metadata template. EUDAT standard template and BBMRI template.

B2HANDLE for FDPs

PIDs in conjunction with their respective resolver, can be seen as persistent URLs. The PID consists of a unique prefix and a suffix, which is unique for the respective prefix. PIDs can be resolved with help of the HTTP protocol via their resolver.

EUDAT employs EPIC PIDs, which are based on the handle technology. In addition to the points above they allow for structuring the PID entries towards the owners' needs, i.e. the owner of a prefix can decide which key-value pairs are stored along the PID in the handle system and thus the owner defines how to mint a PID.

FDPs can leverage EPIC PIDs to provide a stable link to the FDP even if the FDP will be moved and the FDP's URL changes. We see the following applications and implications for and on FDPs.

- FDPs themselves can disappear and be relaunched at a different location. To provide a stable reference to users PIDs can be used here. Those PIDs should resolve to a webpage under which the FDP can be accessed. E.g. http://145.100.59.120:8082/fdp/swagger-ui.html#/

- How to make sure that the FDP is still the same FDP? Data in the FDP can change over time, so it is not possible to verify the identity of a FDP by its hash. In contrast to repositories where you verify the identity and integrity of single data objects, we would like to verify here the identity and integrity of the whole FDP, i.e. data repository.
- In contrast to a data repository and as an additional aggravating factor, the FDPs are not the data owner of the data linked in the triples. I.e. FDPs have no influence on when data changes and where data might be moved to in the future.
- Goals:
    - When links are broken since data has been moved by data owner, decrease downtime of that link
    - Develop strategies and workflows what needs to happen when data that disappeared and data needs to be restored from a back resource.
        - Add triple/annotation indicating that this is not the original data
        - Use PIDs to refer to data objects that are not managed in the EUDAT network. PIDs would provide a stable link to data in triples. When link is broken data can be restored from a previous version of the the FDP (that is kept in B2SAFE)

PID workflows:
As indicated above there is need for certain workflows how to update PIDs in case a FDP moves to a different address or a FDP gets updated with new (meta)data. To this end we need to define roles with respect to who is allowed to move/update FDPs and who is the owner of the PIDs. **However, an initial investigation of the B2HANDLE API has confirmed that it supports all the necessary operations to implement the PID workflows envisioned for FDPs.**

Conclusion

General outline: the first phase of the pilot has focused on knowledge exchange and exploring/evaluating EUDAT services. In the second phase we would like to take the next steps toward the EUDAT infrastructure adopting the FDP concept (as describe in B2SHARE/B2SAFE to FDP). Secondly we would finalize the nearly-completed integration of FDP in B2FIND (FDP to B2FIND). The adoption of PIDs (B2HANDLE for FDPs) will be taken up by the DTL FAIR development team. Whether this should be part of a data pilot needs to be discussed, perhaps it is development effort that can be performed by the DTL team with “regular” EUDAT support. Regarding the use of B2SAFE for FDP/dataset migration: this requires further discussion.

Acknowledgements

Heinrich Widmann (DKRZ, Germany)
Hannes Thiemann (DKRZ, Germany)
Yann Le Franc (e-Science Data Factory S.A.S.U., France)