FAIR Digital Object Framework

Version 1.01, November 2019

FDOF Technical Implementation Guideline

*"We need a set of principles that are sufficiently specific to be useful but sufficiently abstract to exclude specific software stacks, i.e., a document that will still make sense and still be useful ten years from now."*

This document includes some generic guidelines to be met (chapter 2), a normative part defining the FAIR Digital Object Framework (FDOF) at an abstract level which will develop over time (chapter 3), two appendices for illustration purposes indicating two possible implementations (chapter 4) and an appendix including a glossary defining terms (chapter 5). The illustration examples will be separated in one of the next versions and there may be more implementation suggestions being proposed.

## Change History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Intention | Actors |
| Version 1.0 | October 2019 | prepared for the consensus meetings in Washington and Paris in October 2019 | created by Luiz Bonino and Peter Wittenburg |
| Version 1.1 | 17.11. 2019 | created after the consensus meeting in Paris at 28/29.10.2019 | changes by Luiz Bonino and Peter Wittenburg |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Changes from V1.0 to V1.01**

* Restructuring the Document and improving some formulations.
* Leaving out concretization footnotes from the normative part.
* Leaving out footnotes about matters explained in the glossary.
* Making more statements about metadata to indicate their importance.

## Generic Guidelines

Some overall guidelines need to be met by the FAIR DO Framework (FDOF).

**G1**: Show a path for infrastructure investments for **many decades.**

**G2:** Demonstrate **trustworthiness** to researchers and developers to become engaged.

**G3**: Offer compliance with the **FAIR principles** being turned into indicators of FAIRness by an RDA Working Group (<https://www.rd-alliance.org/groups/fair-data-maturity-model-wg>).

**G4**: Support **machine actionability** which includes referential integrity, which states that all references need to be valid without temporal limitation, and explicitness of semantic relationships.

**G5**: Support the **abstraction principle**, i.e. abstract away from details that are not needed at a specific layer. At the management layer there is no difference to be made between data, metadata, software, semantic assertions, etc.

**G6**: Support **stable binding** between all informational entities that are required for machines to act.

**G7**: Support **encapsulation** which means that operations can be associated with types of FDOs.

**G8**: Support **technology independence** allowing implementations using different technologies

## Requirements for FDOF

The requirements for FDOF describe rules that need to be met by any implementation of the FDO concept. These requirements will develop dependent on the insights.

**FDOF1**: A PID, standing for a globally unique, persistent and resolvable identifier, is assumed to be the basis of the Internet of FAIR Data and Services.

**FDOF2**: A PID is resolved to a structured record with attributes which are semantically defined within a type ontology which can have different forms.

**FDOF3**: The structured record includes at least a reference to the locations where the bit-sequences encoding the content of a FAIR-DO (FDO) can be accessed, a PID pointing to the metadata FDO(s) describing properties of it and the DO's type.

**FDOF4**: The structured record can include other attributes that are important to characterize specific types of FDO or that are required by applications. It is required that these attributes are registered in a type registry.

**FDOF5**: Each FDO identified by a PID can be accessed or operated on using an interface protocol by specifying the PID of a registered operation and the PID of the access point.

**FDOF6**: This protocol offers the typical CRUD operations on FDOs and a possibility to use extended operations.

**FDOF7**: The relations between FDO Types and operations are maintained in a type ontology.

**FDOF8**: Metadata descriptions being FDOs and describing the properties of the FDO are made available as semantic assertions enabling machines to act.

**FDOF9**: Metadata assertions can be of different types such as descriptive, deep scientific, provenance, system, access permissions, transactions, etc.

**FDOF10**: Metadata schemas are maintained by communities of practice. FDOF requires that such metadata are FAIR.

**FDOF11**: A collection of FDOs is an FDO and semantic assertions are to be used to describe their construction, i.e. the relationships of their constituents.

**FDOF12**: The "Deletion" of a FDO leads to standardised and thus machine interpretable tombstone notes in the metadata and PID records, i.e. PIDs and metadata should not be deleted.

## Appendix: Implementation Examples for Illustration Purposes

### 4.1 Digital Object Architecture

Here we briefly summarise what is available, what is missing and what can be read.

|  |  |  |
| --- | --- | --- |
| Req | available | missing |
| G1 | basic construction & intention is meant to survive for many decades | broadest uptake to ensure survival |
| G2 | trustworthiness is a social concept and depends on reliability and uptake | solidity of all components is given, broad mobilisation is to be achieved |
| G3 | FAIR compliance to a certain extent (see below) | some higher level specifications are missing |
| G4 | all aspects of DOs specified yet are machine actionable | some higher level specifications are missing |
| G5 | abstraction is at the core of the DO concept | no |
| G6 | stable binding is realised by using the PID record | no |
| G7 | encapsulation is intended and implemented through DOIP | no |
| G8 | DOs are one implementation of FDOF |  |
| FDO1 | based on clearly defined PID systems such as Handle | no |
| FDO2 | resolution is a structured record and attributes should be defined and registered[[1]](#footnote-1) | miss an authority to maintain registry |
| FDO3 | should be specified by FDO service providers (repositories) | miss an authority to define best practices |
| FDO4 | model allows access through PIDs, DOIP allows association between types and operations | no |
| FDO5 | DOIP has these features | no |
| FDO6 | RDA specified a type registry, is being used | more complex ontologies might be necessary |
| FDO7 | metadata are indeed FDOs; metadata availability as assertions is possible | miss specifications associated with the DO model |
| FDO8 | collections are DOs; construction of collections not defined | miss specifications associated with the DO model |
| FDO9 | tombstone notes are possible, yet not defined | miss an authority to define best practices |

The digital object approach provides a framework for the various components needed for a FAIR DO Framework but any implementation will require further specification and one unspecified piece is how to get to the explicit semantics needed for machine understanding of structured metadata and collections.

**Available readings**

* R. Kahn, R. Wilensky (1995): A Framework for Distributed Digital Object Services; <https://www.cnri.reston.va.us/k-w.html>
* R. Kahn, R. Wilensky (2006): A Framework for Distributed Digital Object Services; <https://www.doi.org/topics/2006_05_02_Kahn_Framework.pdf>
* RDA DFT Group: DFT Core Terms and Model; <http://hdl.handle.net/11304/5d760a3e-991d-11e5-9bb4-2b0aad496318>
* RDA DTR Group: Data Type Registry¸ <https://www.rd-alliance.org/group/data-type-registries-wg/outcomes/data-type-registries>
* RDA Kernel Group: Recommendation on PPID Kernel Information; <https://www.rd-alliance.org/group/pid-kernel-information-wg/outcomes/recommendation-pid-kernel-information>
* RDA Research Collection Group: Recommendations; <https://www.rd-alliance.org/group/research-data-collections-wg/outcomes/rda-research-data-collections-wg-recommendations>
* DONA: DOIP V2.0, <https://www.dona.net/sites/default/files/2018-11/DOIPv2Spec_1.pdf>
* P. Wittenburg, G. Strawn: Common Patterns in Revolutionary Infrastructures and Data; <http://doi.org/10.23728/b2share.4e8ac36c0dd343da81fd9e83e72805a0>
* P. Wittenburg, G. Strawn, B. Mons, L. Bonino, E. Schultes: Digital Objects as Drivers towards Convergence in Data Infrastructures; <http://doi.org/10.23728/b2share.b605d85809ca45679b110719b6c6cb11>
* S. Hodson et. al.: Turning FAIR into Reality; <https://doi.org/10.2777/1524>
* E. Schultes, P. Wittenburg: FAIR Principles and Digital Objects: Accelerating Convergence on a Data Infrastructure; <http://doi.org/10.23728/b2share.166a074bff614a31b05e9df5bfd9809d>
* G. Strawn: Open Science, Business Analytics, and FAIR Digital Objects; <http://doi.org/10.23728/b2share.6ceeed13eb6340fcb132bcb5b5e3d69a>
* K. de Smedt, D. Koureas, P. Wittenburg: Analysis of Scientific Practice towards FAIR Digital Objects; <http://doi.org/10.23728/b2share.e14269d07ce84027a7f79ee06b994ef9>

## 4.2 Linked Data Platform

Here we briefly summarise what is available, what is missing and what can be read.

|  |  |  |
| --- | --- | --- |
| Req | available | missing |
| G1 | The corpus of LDP-related recommendations provide a clear investment path for a reasonable amount of time. | Increase the adoption base of the technology. |
| G2 | LDP is an W3C recommendation and W3C is recognised as a trustworthy standardization organization. |  |
| G3 | LDP and associated W3C standards facilitate the compliance with the FAIR Principles, but users need to behave in certain ways to do so. | More strict guidelines on how to use LDP in ways to better follow the FAIR Principles |
| G4 | LDP and Linked Data provides the technological ground for users to provide explicit semantics with qualified references. |  |
| G5 | Layers of abstractions can be introduced using LDP/RDF. | More strict guidelines on how to use LDP in ways to better support the abstraction principle. |
| G6 | Once the relations/bindings are defined, they are there until the resource is removed or the users make changes. |  |
| G7 | LDP supports the HTTP methods/operations |  |
| G8 | LDP is a technology based on RDF and, therefore, is technology dependent. |  |
| FDO1 | LDP adopts URI as a globally unique, persistent and resolvable identifier. | A consistent resolution behaviour that doesn’t depend on user’s best practices is still missing. |
| FDO2 | What is resolved from the URI is up to the creator of the resource. It is possible to define the resolution to this structured record and LDP/RDF provides infrastructure for semantically describe this structured record. | Instructions on how to use LDP to return the URI’s structure record including the semantic references to the record’s elements. |
| FDO3 | Same as above | Same as above |
| FDO4 | LDP/RDF can be used to provide the semantic description of the operations supported by each DO type. | The type ontology, including the description of the operations need to be defined. |
| FDO5 | LDP supports HTTP methods that provide CRUD functionality. | Extended operations need to be defined. |
| FDO6 | LDP/RDF the definition of the semantic descriptions required by the type ontology. | no |
| FDO7 | LDP/RDF supports semantic descriptions of metadata elements through qualified references to existing volcabularies/ontologies. | no |
| FDO8 | LDP defines the concept of container, including three types of containers and the relations between container and its member elements. | no |
| FDO9 | Tombstone notes can be semantically described using LDP/RDF. | Instructions on how to construct the tombstone notes using LDP/RDF and update the identifier’s structure record to point to this note. |

**Available readings**

* Linked Data Platform 1.0 - <https://www.w3.org/TR/ldp/>
* Linked Data Platform 1.0 Primer - <https://www.w3.org/TR/ldp-primer/>
* Linked Data Platform Best Practices and Guidelines - <https://www.w3.org/2012/ldp/hg/ldp-bp/ldp-bp.html>
* RDF 1.1 Primer - <https://www.w3.org/TR/rdf11-primer/>

## Appendix: Glossary

A short glossary with explanations about crucial terms such as "repository", "encapsulation" etc. will help in clarifications, since some terms may be interpreted differently by the participants.

|  |  |
| --- | --- |
| Term | Explanation |
| abstraction | Abstraction is a conceptual process where general rules and concepts are derived from the usage and classification of specific examples. literal signifiers, first principles or other methods (Wikipedia) |
| binding | With binding we mean the possibility for humans and machines to find other relevant entities of a DO when being exposed to another, i.e. when an actor receives a PID of a DO it must find the PID of the corresponding metadata DO and the access rights information, since otherwise interpretation and access is impossible |
| collection | A collection is a complex DO consisting of other DOs, that have a PID and metadata. |
| CRUD operations | These are the usual primary type of operations such as create, read/retrieve, update and delete |
| encapsulation | Encapsulation is known from abstract data types and oo programming where internals of data objects are hidden to the user and where the user can only influence the internal state by using defined methods |
| **Note**: in the FDO case DO types can be associated with registered operations that can be used to operate on DO's content |
| machine actionability | With machine actionability the capacity of computational systems is meant to find, access, interoperate and reuse data and services without human intervention (GOFAIR) |
| metadata | Metadata descriptions of DOs are sets of assertions describing properties of DOs content which are required for finding, accessing, interpreting and reusing, these assertions can cover a wide range such as descriptive to support finding, deep scientific to support science, systemic to support management, rights to prevent unauthorized access, etc. |
| **Note**: Yet the domain of metadata is not structured very well, i.e. terminology is not well-defined. |
| **Note**: Basic interoperability assumptions are that the schemas are registered and the concepts defined and registered. |
| repository | **DO View**: from the perspective of Digital Objects repositories are nothing else than a complex DO associated with a PID, metadata of different kinds and functions to offer DOs |
| **Common View**: from the most common point of view repositories are entities that host data, metadata etc., apply trustworthy management procedures, offer a search and access interface, have a team of experts taking care and have a sustainability plan |
| **Note**: repositories can be associated with research organisations, communities or projects, they can be small or big in terms of the collections they hold. |
| type | "Type" is an attribute of digital objects which tells computational actors how the content of the DO needs to be parsed, i.e. it defines the operations that can be done on the data, the meaning of the data, and the way values of that type can be interpreted |
| **Note**: A MIME type is a standard that indicates the nature and format of a document, file, or assortment of bytes, i.e. it is a restricted concept of type. |
| **Note**: A type of a DO implies a summary of otherwise complex metadata assertions describing the format, encoding etc. of a content. |

1. It should noted that some Handle Services do not support machine actionable types in the Handle record. [↑](#footnote-ref-1)