FAIR Digital Object Framework

Version 1.02, 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) and a glossary of terms (chapter 4). Related documents such as implementation examples can be found at the Github site[[1]](#footnote-1) and a Google Doc version is used for open comments which will lead to new versions[[2]](#footnote-2).

## 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 1.01 | 17.11. 2019 | created after the consensus meeting in Paris at 28/29.10.2019 | changes by Luiz Bonino and Peter Wittenburg |
| Version 1.02 | 22.11. 2019 | created after various comments incl. polishing and adding clarity | changes by Peter Wittenburg, Bonnie Carroll, Alex Hardisty, Mark Leggott, Carlo Zwölf |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**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.

**Changes from V1.01 to V1.02**

* The illustration examples of possible implementations were separated from the FDOF core document.
* Metadata Statements were added to address the importance of metadata.
* The first editing group did improvements on the original text (polishing, clarity)
  + Throughout, tidy up of grammar and punctuation to improve clarity.
  + Definitions added in glossary for terms ‘FAIR-DO’, ‘FAIRness’ and ‘semantic assertion’.
  + Generic guideline G9 on using standards added.
  + Removed reference to RDA work in G3 and introduced the idea of indicators being measurable.
  + Moved definition of referential integrity out of G4, into the glossary.
  + In G5, replaced ‘layer’ by ‘level’ and qualified the ‘management level’ as meaning the level of managing objects.
  + The statement saying the FDOF requirements will evolve with experience is now a note.
  + FDOF03 clarified to make clear i) that the separation of metadata into a DO different from the FAIR-DO, with it’s own pointer in the structured record of the resolved PID is an optional element; and ii) that the type definition is also accessed via a PID.
  + FDOF04: made it mandatory that PID records with additional attributes beyond the standard ones must be registered in a type registry.
  + Throughout section 3, introduced the term ‘PID record’ as the correct term for the structured record returned by PID resolution.

## 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** through measurable indicators of FAIRness.

**G4**: Support **machine actionability,** which includes referential integrity and explicitness of semantic relationships.

**G5**: Support the **abstraction principle**, i.e. abstract away from the details that are not needed at a specific level. At the object management level 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 specific operations can be associated with different types of FDOs.

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

**G9**: Comply with agreed **standards** (e.g., for exchange of FDOs between systems, for interacting with FDOs, etc.) so that machine-machine interoperability can be achieved across heterogeneous systems.

## Requirements for FDOF

The requirements for FDOF describe rules that must be met by all implementation of the FDO framwork.

Note: Requirements will evolve, dependent on insights obtained from implementation experience.

**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 resolves to a structured record (PID record) with attributes that are semantically defined within a type ontology (which can have different forms).

**FDOF3**: The structured PID record includes at least a reference to the location(s) where the bit-sequences encoding the content of a FAIR-DO (FDO) and the type definition of the FDO can be accessed. The structured record may also contain a PID pointing to a metadata DO (itself an FDO) describing properties of the target FDO.

**FDOF4**: The PID record may include other attributes that are important to characterize specific types of FDO or that are required by applications. Additional attributes being used in PID records must be 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 standard Create, Read, Update, Delete (CRUD) operations on FDOs and a possibility to use extended/domain operations for specific applications.

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

**FDOF8**: Metadata descriptions being themselves FDOs and describing the properties of the FDO must be 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 also an FDO and semantic assertions must be used to describe their construction, i.e. the relationships of their constituents.

**FDOF12**: Deletion of a FDO must lead to standardised and thus machine interpretable tombstone notes in metadata and PID records, i.e. PIDs, PID records and metadata should normally not be deleted, but should be modified to indicate that the FDO associated with a particular PID no longer exists.

## 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, which 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 object oriented 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 |
| FAIR Digital Object (FAIR-DO) | FAIR Digital Objects can represent data, software, protocols or other research resources. They are accompanied by persistent identifiers (PID) and metadata rich enough to enable them to be reliably found, used and cited. (FAIR Implementation Report: [doi: 10.2777/1524](https://doi.org/10.2777/1524), and Wittenburg & Strawn 2019 [doi: 10.23728/b2share.2317b12321764f669c92ebbcf7518164](http://doi.org/10.23728/b2share.2317b12321764f669c92ebbcf7518164) ) |
| FAIRness | FAIRness is a characteristic exhibited by an infrastructure component when it maintains compliance with the principles of FAIR. Achievement of FAIRness is demonstrated, for example by achieving a score (passing a threshold) in an assessment against an agreed set of maturity indicators. |
| machine actionability | machine actionability means the capacity of computational systems 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. |
| referential integrity | The idea that all PID references must resolve and be valid without temporal limitation |
| semantic assertion | The attachment (perhaps by reference to a defined vocabulary) of a specific meaning to a resource, attribute, property, etc. |
| 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. <https://github.com/GEDE-RDA-Europe/GEDE/tree/master/FAIR%20Digital%20Objects/FDOF> [↑](#footnote-ref-1)
2. https://docs.google.com/document/d/17WQe5Y3sxcIBKQUzEhnWqcjVjB4UABXC/edit [↑](#footnote-ref-2)