FAIR **DIGITAL OBJECTS** FORUM
https://fairdo.org/

FDO Forum FDO Configuration Types Version 2.0

FDO Forum Proposed Recommendation 7 June 2022

Current and previous versions:

Current: PR-ConfigurationTypes-2.0-20220608,
https://docs.google.com/document/d/1ivvg3C_QWSO9PIQwkKT89xG4fBhNAs7_6b0Dz11EwDg/edit

Previous:
https://docs.google.com/document/d/1ivvg3C_QWSO9PIQwkKT89xG4fBhNAs7_6b0Dz11EwDg/editt

Editors:
Ivonne Anders, Andreas Pfeil, Peter Wittenburg

Authors:
Larry Lannom, Karsten Peters-von Gehlen, Ivonne Anders, Andreas Pfeil, Alexander Schlemmer, Zach Trautt, Peter Wittenburg

Abstract

Intensive discussions within the FDO Forum revealed that there are many different ways to organize FAIR Digital Objects and many of them listed in this document are being used in daily practice. Investigating these different organizations and checking the compliance with the FDO Requirement Specifications[1] and the FAIR Principles are essential. For illustration purposes this document includes simple diagrams and verbose descriptions of the different configurations. The major conclusions from this analysis need to be integrated into the full FDO Specifications.

Status of this document

This document is a Working Draft of a recommendation track document that will be discussed again in the FDO TSIG Working Group as WD 0.1 since it includes some

[1] The FDO Framework Requirements document has been renamed to FDO Requirement Specifications and the last version can be found in this folder:
https://drive.google.com/drive/u/0/folders/1aRXBgmgCMjcFDjDzNJes338fGtIXeB-x

major extensions compared to the previous TSIG version which has not yet been registered in the document process system.

Acknowledgments

The original internal version has been reviewed by many TSIG members.

Contents

1. Conclusions

In this chapter we will draw a set of conclusions from the descriptions below.

1. There are several different ways to organize FAIR Digital Objects (FDOs) and nevertheless adhere to the FDO Requirement Specifications. In all definitions of the term "FAIR Digital Object (FDO)" this variety needs to be considered.

2. Users request more guidance in terms of, e.g., implementing FDO systems. Therefore, we are introducing the term "Recommended FDO Configuration Types" and assign it to two organizational models.

    a. The first recommended model offers a high granularity in the case of closely bundled objects. Every component is represented as its own FDO and is aggregated in an overall FDO. This approach ensures that all components can be addressed separately.

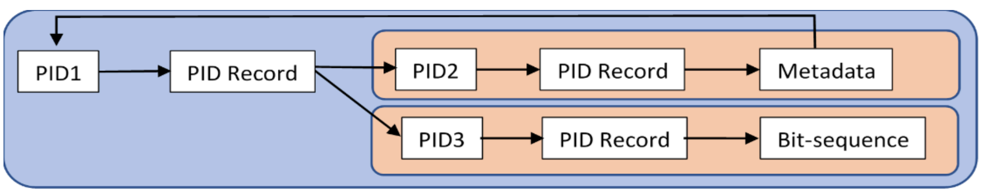
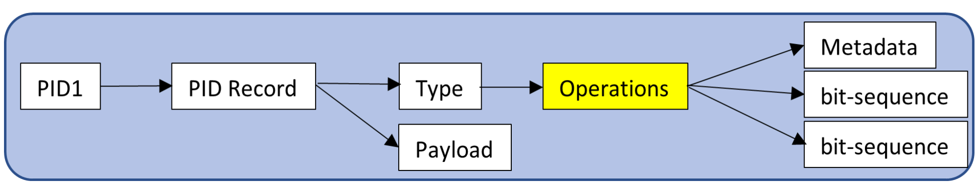


Figure 1

    b. The second recommended model offers a high flexibility in adding a variety of different components all linked together by defined operations



on information types.

Figure 2

2. Generics

2.1 Usage of Terms

This document is based on the FDO Framework which were originally defined by the Paris Meeting[2] which claims to be compliant with the FAIR Principles and has been extended to the FDO Requirement Specifications. Based on the FDOREC Requirements (see appendix) we create short FDO profiles that indicate the degree of compliance per model.

It should be noted that the FAIR principles require metadata to be machine actionable. The challenge we see is that many communities worked out their metadata practices often including schemas but without defining and registering the semantic categories being used in a way that machine actionability is guaranteed. The term "machine actionability" as being used in the FDO Forum and in this document are being defined in WD-MachineActionDef-1.1-20220225.pdf.

This paper also anticipates the definition of the Base Definition of FDOs as defined by the FDO Forum. However, it recognizes that this definition will have to be overworked based on the conclusions drawn from this document. In the following we cite the base and the slightly extended technical definitions:

A base definition
A FAIR digital object is a unit composed of data and/or metadata regulated by structures or schemas, and with an assigned globally unique and persistent identifier[2] (PID), which is findable, accessible, interoperable and reusable both by humans and computers for the reliable interpretation and processing of the data represented by the object.

A full technical definition
A FAIR digital object is a unit composed of data that is a sequence of bits, or a set of sequences of bits, each of the sequences being structured (typed) in a way that is interpretable by one or more computer systems, and having as essential elements an assigned globally unique and persistent identifier (PID), a type definition for the object as a whole and a metadata description (which itself can be another FAIR digital object) of the properties of the object, making the whole findable, accessible, interoperable and reusable both by humans and computers for the reliable interpretation and processing of the data represented by the object

2.2 Used Graphical Indication

We are deliberately using a simple graphic illustration allowing also laypersons to understand the diagrams – no a priori knowledge is required. The diagram style is explained with help of figure 3.

- PID3 identifies an FDO having a bit-sequence of any type (data, software, etc.). The arrows indicate that PID3 will resolve to a PID record and this record amongst other attributes includes a reference to the location of the bit-sequence of this FDO.

---

2 https://github.com/GEDE-RDA-Europe/GEDE/tree/master/FAIR%20Digital%20Objects/Paris-FDO-workshop

- PID2 identifies a FDO having a bit-sequence encoding the metadata of the FDO in some structure and including a reference to its location.
- PID3 identifies the bundle FDO which in this case includes a metadata and another object (of course many more objects all having their own PIDs could be included).
- To support proper referencing the metadata of the bundle FDO needs to include its PID1. This is necessary since people will often carry out search operations on metadata using some portal and then want to access the FDO by using the found PID1.

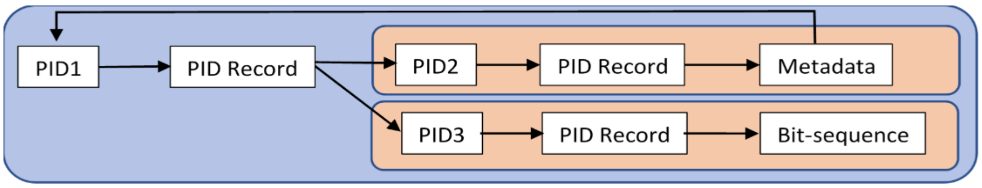


Figure 3

Therefore, the colored areas represent FDOs. In case that the illustrations deviate from this simple form it will be marked.

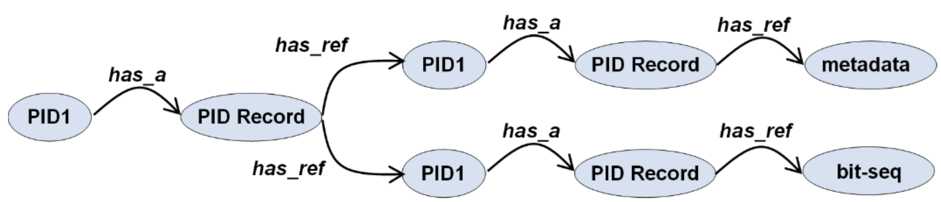


Figure 4

This can easily be turned into a graph as it is used in the Semantic Web domain.

2.3 FDO Requirement Specification (FDOREC) Compliance

As indicated in the above illustration example, not all categories which are included for example in the metadata description are of relevance at FDO-level processing. Such metadata may include specifications about, e.g. “energy categories” or “handedness of deaf speakers”. Such metadata elements are only of use for deep scientific operations. Therefore, we only need to check whether those elements which for example represent references to components relevant for FDO-level processing are machine actionable. The scope of this has not yet been defined by the FDO Forum.

3. FDO Configuration Types

3.1 Configuration Type 1

In this configuration type we assume that the path to the bit-sequence is in the metadata[3]. In terms of the definition this can be called an FDO if FAIRness can be shown[5]. In many cases database management systems are being used, i.e., the path to the metadata description can be either a record pointer or an SQL query dependent on the database system.

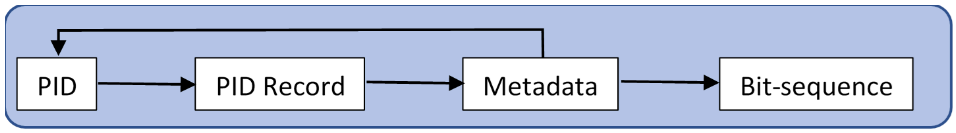


Figure 5

***Example**: This model is being used by many repositories that started assigning PIDs, but did not want to change their existing organization.*

FDOREC Compliance:

(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The DO Type must be defined and registered.
(3) The metadata must be accessible, schema-based and at least the attributes used to link to the bit-sequence and the metadata category containing the PID of the FDO must be defined and registered.
(4) The bit-sequence must exist and be accessible.

3.2 Configuration Type 2

In this configuration type the PID record also contains the information where the bit-sequence can be found. In terms of the definition this can be called an FDO if FDOREC Compliance can be shown. Since the access to the FDO could happen via a metadata search the metadata needs to include the PID of the FDO.

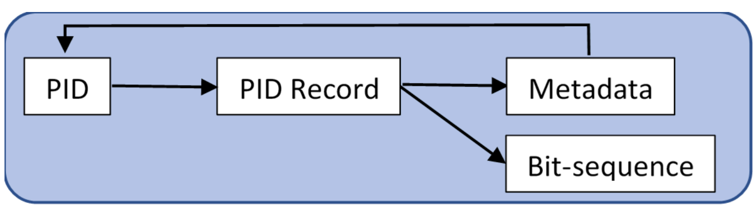


Figure 6

***Example**: Also, this model is being used by many repositories that started assigning PIDs, but did not want to change their existing organization.*

[3] It should be noted that metadata in this note covers the usual assertions about descriptive, deep scientific, provenance, license and access permission attributes. In the strong sense of FAIR such metadata should be assigned a PID (F1) and is treated just as any other type of data, i.e. it would be an FDO.

FDOREC Compliance:

(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The DO Type must be defined and registered.
(3) The metadata must be schema-based and accessible, and the element that contains the PID must be defined and registered.
(4) The bit-sequence must exist and be accessible.

3.3 Configuration Type 3

In this configuration type the metadata description itself is an FDO. In terms of the definition this can be called an FDO if FDOREC Compliance can be shown. Since metadata is embedded in an own FDO in this case, it must contain the PID1 to be able to resolve to the FDO itself from the metadata.

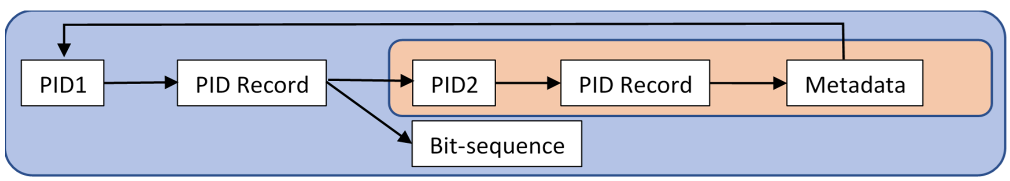


Figure 7

***Example**: Also, this model is being used by some repositories.*

FDOREC Compliance:

(1) The DO Types must be defined and registered.
(2) The PID1 record must be guided by an explicit PID profile and only include defined and registered attributes.
(3) The bit-sequence must exist and be accessible.
(4) The PID2 record must be guided by an explicit PID profile and only include defined and registered attributes.
(5) The metadata must be schema-based and accessible, and the element that contains the PID1 must be defined and registered.

3.4 Configuration Type 4

In this configuration type the metadata description and the bit-sequence(s) are both embedded in FDOs. In terms of the definition this can be called an FDO if FDOREC Compliance can be shown. Since metadata is embedded in an own FDO in this case, it must contain the PID1 to be able to resolve to the FDO itself from the metadata.

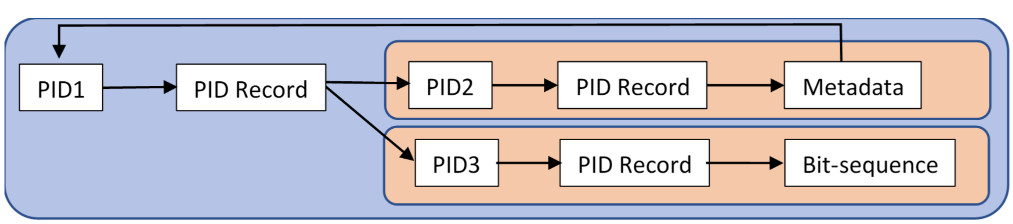


Figure 8

Example: *This organization model has been implemented in the DOBES project[4] as it was briefly described in the "Turning FAIR into Practices" document of the EC[5]. A PID (Handle) indicates a Digital Object and is associated with a record that includes amongst other attributes PIDs that point to a set of files that contain bit-sequences of closely related content (e.g. audio, video and annotation files sharing the same time scale, hundreds of photos taken at the same trip, etc.) and to the metadata description of the whole bundle which can be seen as a collection. Each of the bit-sequences can themselves bei related with metadata which is not shown here. It is up to the researcher to decide how to use this flexible organization model and thus about granularity.*

FDOREC Compliance:

(1) The DO Types must be defined and registered.
(2) The PID1 record must be guided by an explicit PID profile and only include defined and registered attributes.
(3) The PID2 record must be guided by an explicit PID profile and only include defined and registered attributes.
(4) The metadata must be schema-based and accessible, and the element that contains the PID1 must be defined and registered.
(5) The PID3 record must be guided by an explicit PID profile and only include defined and registered attributes.
(6) The bit-sequence must exist and be accessible.

3.5 Configuration Type 5

Some communities may not make a difference between metadata and data and treat them as being integrated. In this case the PID record would refer to a structured bit-sequence that may include metadata and some data.

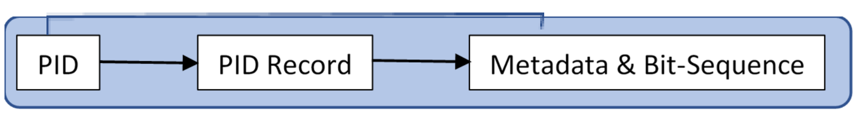


Figure 9

Example: *In social sciences the DDI schema is used and can include metadata on surveys as well as the survey data itself in a structured form. Another example of this approach would be the NetCDF file format common in climate science. NetCDF files are self-describing and can contain all necessary metadata in their header. As in the other examples a reference back from the metadata to the FDO needs to be made available.*

FDOREC Compliance:

(1) The PID1 record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The DO Type must be defined and registered.
(3) The metadata & bit-sequence must exist, be accessible and be schema-based.The metadata must conform to the machine actionability requirements presented within this specification.

[4] https://dobes.mpi.nl/
[5] https://ec.europa.eu/info/sites/default/files/turning_fair_into_reality_1.pdf

3.6 Configuration Type 6

In this configuration type the PID record contains some minimal metadata: (1) one of the elements being the type of the object and (2) a thumbnail as payload element that can be directly visualized. A separate community-managed registry allows relating the type with a set of operations which can be used to, e.g., extract more detailed information such as deep metadata, high resolution images, etc. This organization model heavily relies on operations to access rich information sources in contrast to many others sketched in this document. It should be ensured that the metadata contains a reference back to the FDO.

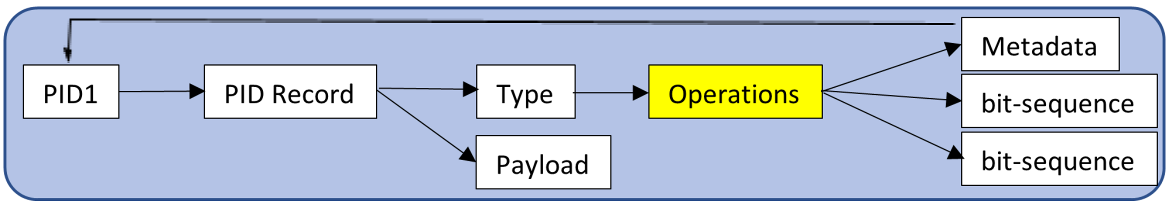


Figure 10

***Example**: This data organization model is used by DISSCO to design a digital specimen repository for natural science collections data with components stored in distributed locations. As is shown in figure 11, a digital specimen representing the physical specimen of a cone snail is identified by a Handle (20.5000.1025/xyxy123). This persistent identifier can be resolved to a rich information bundle: a local identifier (MNHN-IM-2013-53462 is stored in the French National Museum of Natural History), a digital object type definition (ODSType1803), some metadata according to an agreed standard, and the thumbnail image of the snail as payload. Besides this minimal metadata, a variety of operations have been defined for this digital object type which gives access to many different pieces of information associated with this type. The DISSCO community has the advantage that they were able to design the specimen database from scratch as a super layer on top of the different organization forms used in the different biodiversity centers.*

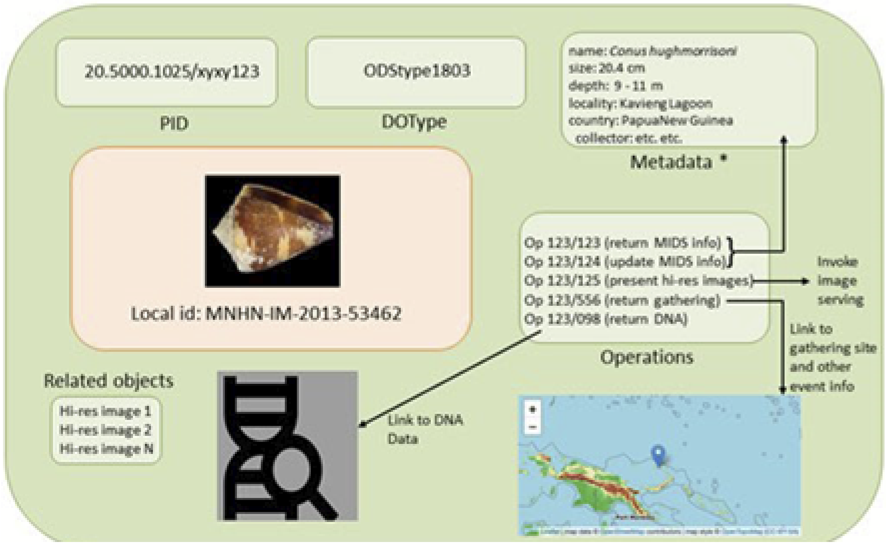


Figure 11

FDOREC Compliance:

(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The DO Type must be defined and registered.
(3) The registry relating types and operations must be findable and formatted following community specifications.
(4) The bit-sequence contained as payload must be machine actionable.

3.7 Configuration Type 7

In this organization model a human-readable landing page encoded in HTML is seen as the anchor point for access. It includes metadata elements which are generated from a metadata description. The landing page then contains the PIDs that point to the metadata record and to the bit-sequence[6].

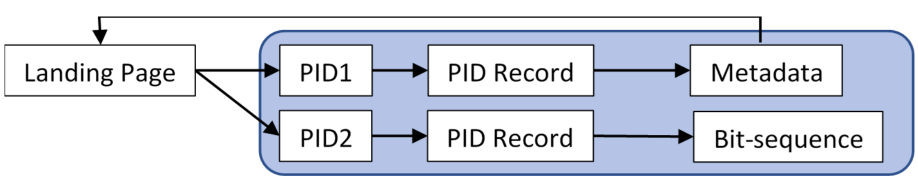


Figure 12

***Example**: The B2Share repository from EUDAT has chosen a variant which focuses on a landing page generated from the stored metadata and is meant for human consumption. An API where the PIDs can be resolved to the available information is also made available. It is important that the metadata includes PID2 to be able to access the*

[6] It should be noted that many different bit-sequences can be included, all having their own PIDs.

bit-sequence. The current implementation does not provide the link from the PID2 record to the metadata.

FDOREC Compliance:

(1) The landing page concept is not compliant with the FDOREC Requirements. In case that the PIDs are available the following conditions must be met:
(2) The PID1 record must be guided by an explicit PID profile and only include defined and registered attributes.
(3) The DO Type must be defined and registered.
(4) The metadata must be schema-based and accessible and it needs to be machine actionable since it contains essential references.
(5) The PID2 record must be guided by an explicit PID profile and only include defined and registered attributes.
(6) The bit-sequence must exist and be accessible.

3.8 Configuration Type 8

In many cases PIDs are used to point to other PIDs which may point to some other information. In terms of the definition, we can speak about an FDO if FDOREC Compliance can be shown for, e.g., the "first" FDO. It would include the second PID which is the result of the resolution process. In such a case, the second PID could also imply an FDO if FAIR criteria are fulfilled. It is not obvious how the reference back to the PID1 or PID2 is being integrated.

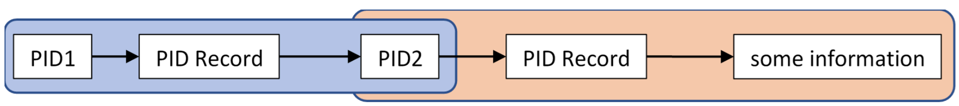


Figure 13

Example: This case is used by some repositories that use a PID indirection scheme.

FDOREC Compliance:

(1) The PID1 record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The PID2 record must be guided by an explicit PID profile and only include defined and registered attributes.
(3) If there is an associated bit-sequence it must exist and be accessible.
(4) The DO Types must be defined and registered.

3.9 Configuration Type 9

This configuration type is comparable to the one presented in configuration type 1 except that the URL will lead to an HTML landing page[7]. In terms of the definition this can be called an FDO if FDOREC Compliance can be shown. The Linked Data community is working hard to indicate that FDOs can be created following the Web protocols and mechanisms. It needs to be clarified how the reference back to the URL is being realized.

---

[7] It should be noted that some institutions rely on persistent URLs (PURLs).

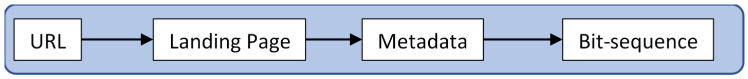


Figure 14

***Example**: The Signpost[8] approach is meant to fill the gap by suggesting a structured landing page and by using standard elements as registered by IANA and schema.org to point to different information entities and to include kernel metadata. The advantage of this approach is that all required elements are already standardized.*

FDOREC Compliance:

(1) The URL needs to have a predictable and persistent resolution behavior and lead to a landing page that has a defined structure with defined and registered attributes.
(2) The metadata must be schema-based and accessible and at least the attribute that contains the link to the bit-sequence must be defined and registered.
(3) The bit-sequence must exist and be accessible.

3.10 Configuration Type 10

This configuration type is comparable to the one presented in configuration type 9 except that the URL will lead to an HTML landing page pointing not only to the metadata but also to bit-sequences. It needs to be clarified how the reference back to the URL is being realized.

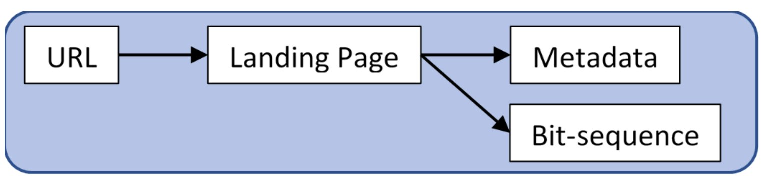


Figure 15

***Example**: Also in this example, the Signpost approach is meant to fill the gap.*

FDOREC Compliance:

(1) The URL needs to have a predictable and persistent resolution behavior and lead to a landing page that has a defined structure with defined and registered attributes.
(2) The metadata must be schema-based and accessible.
(3) The bit-sequence must exist and be accessible.

3.11 Configuration Type 11

Increasingly often cloud systems (mostly offered by companies such as MS, Amazon, etc.) are being used to manage large data sets. The question emerges in how far cloud stores can be made FDOREC compliant. Cloud stores basically hide internal complexity by offering a simple API where access is given via an internal ID or exported core

[8] https://signposting.org/

metadata which is generated from a fast internal database. The administration layer may be organized differently. In the diagram we assume that the internal ID is resolved to pointers to the metadata and to the associated bit sequence. Since cloud administration is proprietary in the case of commercial systems, we can only make statements about FDOREC Compliance if for some reason a PID (globally unique, resolvable and persistent identifier) is being used by the research community. Actually, this PID could be used to link to deep[9] scientific metadata and could contain other attributes about the stored object.

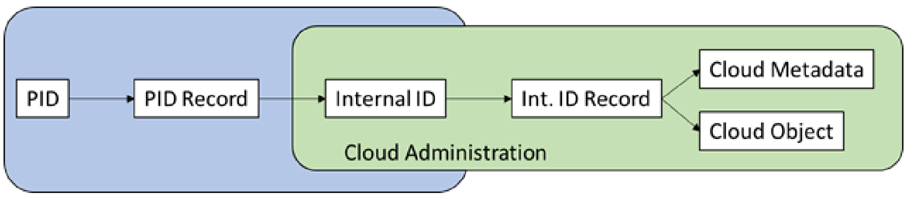


Figure 16

FDOREC Compliance:

(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The DO Type must be defined and registered.
(3) The internal ID must exist and be resolved into useful information/data.

3.12 Configuration Type 12

Traditionally much data is stored in SQL databases which in general contain a set of related tables and SQL queries give access to a selected set of structured data. The logical structure (structure, attributes, relations) of such a database is well-defined locally. In many cases there will be no link to community wide ontologies.

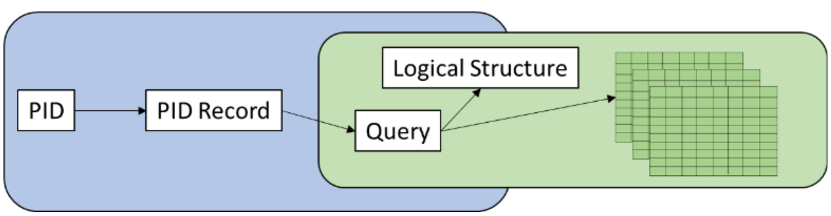


Figure 17

FDOREC Compliance :

(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The DO Type must be defined and registered.
(3) There must be SQL statements that can be executed to retrieve the metadata and the data.
(4) Full FDOREC Compliance will only be given if the variables used in the logical structure are linked to community wide definitions.

[9] The term "deep" scientific metadata indicates that communities are often using detailed scientific categories in their metadata description that go far beyond the DublinCore semantic to enable science.

3.13 Configuration Type 13

Increasingly often SQL databases are replaced by NoSQL databases such as Rasdaman[10], MongoDB[11] etc. Rasdaman for example is a flexible and scalable database to store array/cube-like data, i.e., it is optimized to store and access large series of multivariable records. The access languages allow to address row groups, column groups or even down to voxels. Metadata is used to store the variable names etc. Internally, identifiers are used to indicate structural elements.

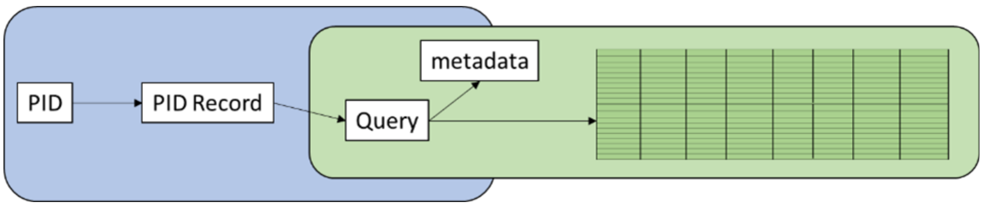


Figure 18

***Example**: This type of database is currently very often used to store endless numbers of structurally identical records such as SMS messages, tweets, etc..*

FDOREC Compliance :

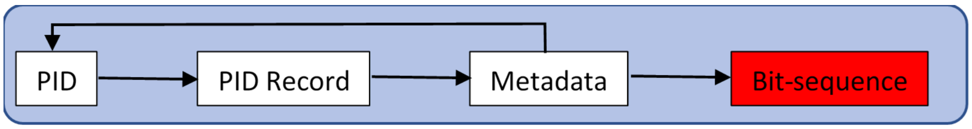
(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes.
(2) The DO Type must be defined and registered.
(3) Executable queries to retrieve the metadata and the data must exist.
(4) Full FDOREC Compliance will only be given if the variables used in the logical structure are linked to community wide definitions.

3.14 Delete Operations

"Deletion" is an operation that is and will be frequently used when bit-sequences have become obsolete because the same phenomena have been recorded with higher resolution after some years, for example. In the case of FDOs, the deletion operation is associated with some requirements which will be demonstrated for a few configuration types only. The problem is that some researchers could have been referred to exactly this bit-sequence for whatever reasons. Deleting the "whole FDO" is therefore not allowed since it would result in link rot.

3.14.1 Deletion Operation 1

When in configuration type 1 the once existing bit-sequence has been deleted, we need to indicate what FDOREC Compliance means.



[10] https://www.rasdaman.com/
[11] https://www.mongodb.com/

Figure 19

FDOREC Compliance [12]:

(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes.

(2) The metadata must be accessible, schema-based and at least the attribute used to link to the bit-sequence must be defined and registered and contain an indicator that the bit-sequence is not available anymore (tombstone indicator). This indicator (value of the corresponding element) must be widely agreed so that machines can take actions.

In cases where a new version exists, it may be desired to refer to a new version.

3.14.2 Deletion Operation 2

In this configuration type the PID record also contains the information where the bit-sequence can be found as long as it existed. When in configuration type 2 the once existing bit-sequence has been deleted, we need to indicate what FDOREC Compliance in this case means.

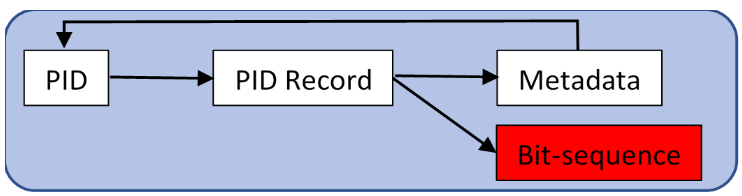


Figure 20

FDOREC Compliance :

(1) The PID record must be guided by an explicit PID profile and only include defined and registered attributes and accessible.

(3) The link that pointed to the bit-sequence must have been replaced by an indicator (tombstone indicator) that is widely agreed.

In those cases where a new version exists, it may be desired to refer to a new version.

4. Changes from previous versions

This version 0.1 has the status of a WD to be discussed in the FDO Forum.

Version	who	when	changes
TSIG Note	TSIG members	5.10.2021	The internal draft document emerged from intensive TSIG interactions.
0.1	Editors	17.12.2021	This version was created by transforming the internal document to the FDOF document style and by adding conclusions.
2.0	Editors	7.6.2022	- Many language improvements have been done. US spelling has been applied. Typos were corrected.

			- This requirement has been systematically added to all configuration types: “The DO Type must be defined and registered.” - Some missing references were added. - The term “FDO F” has been replaced by “FDOREC”, since the “FDO Framework” has been renamed to “FDO Requirement Specifications”. - The style was adapted to Google Doc style. - Figure numbers were introduced

5. Appendix FDO Profiles

Here we add FDO Profiles in a table overview form and describe per FDO Framework requirement whether we can speak about compliance. We start with configuration type 6, since for the example of a Digital Specimen FDO this analysis has been provided in detail by A. Hardisty.

5.1 FDO Profile for Configuration Type 6

The DISSCO analysis revealed that a few requirements are not specific enough to be able to apply them. This has led to starting a revisioning process for the FDO Framework which has not finished yet.

Profile specification statement	**FDOF requirement**
A Digital Specimen FDO must use a Digital Object Identifier (DOI) as its PID.	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.
The PID of a DS FDO must resolve to a PID Record with a kernel information type profile[13] for objects of type 'ODSType1803' in which the attributes are those defined by the Handle System, as adapted for DOIs, as further adapted for Digital Specimen objects.	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).
The PID Record must contain at least: · A Handle pointing to the DS object type definition 'ODSType1803'; · A URI pointing to the storage location of the DS object. Note: DS object contains the metadata and data.	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.
The PID record must include typed attributes institution and physicalSpecimenIdentifier pointing to a location where the physical specimen can be accessed.	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.
The protocol to be used to access and operate on DS FDOs must be Digital Object Interface Protocol (DOIP). For an interim period and with the possibility that functionality is reduced it may be possible to access and operate on DS FDOs via an HTTP REST API.	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.
In additional to the standard operations (CRUD+ListOps) the following extended/domain operations must be supported[14]: · Fetch (return) MIDS information · Update MIDS information · Fetch (return) thumbnail image · Present hi-res images · Fetch (return) gathering event information · Fetch (return) NCBI sequence data	FDOF6: This protocol offers standard Create, Read, Update, Delete (CRUD) operations on FDOs and a possibility to use extended/domain operations for specific applications.

The type definition for the DS FDO object type as whole, and all associated type definitions must be maintained in the ePIC Data Type Registry[15].	FDOF7: The relations between FDO Types and operations are maintained in a type ontology.
– unclear. What is meant by 'semantic assertions'? Example needed.	FDOF8: Metadata descriptions being themselves FDOs and describing the properties of the FDO must be made available as semantic assertions, enabling machines to act.
Metadata assertions for DS FDOs *– list of classes needed, with references out to where these are defined.*	FDOF9: Metadata assertions can be of different types such as descriptive, deep scientific, provenance, system, access permissions, transactions, etc.
References out to the metadata schemas for each of the classes listed about. *– unclear what is needed to fulfil second sentence.*	FDOF10: Metadata schemas are maintained by communities of practice. FDOF requires that such metadata are FAIR.
– again, unclear what is needed here.	FDOF11: A collection of FDOs is also an FDO and semantic assertions must be used to describe their configuration, i.e. the relationships of their constituents.
The tombstone object for DS FDOs can be found at the location indicated by <this handle>.	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.

5.2 FDO Profile for other Configuration Types

As indicated above, we assume that when PIDs are used the FDO has a structured bit-sequence encoding the content, has an assigned PID with Kernel Information according to a registered profile and associated metadata. In practice this is mostly not the case yet, but several teams are working hard to improve their PID usage[12]. We also assume that the PID of the FDO is included in the metadata as well.

Configuration Type 7 uses a human-readable landing page as anchor point and as shown, the PIDs do not give access to all kinds of information in all cases. A systematic extension would resolve this shortcoming and make the approach FAIR.

For configuration types 9 and 10 with a landing page approach it is obvious that a well-structured Signpost type of HTML description using standardized vocabularies and relation types is needed to fulfill the requirements. It is not clear how far the research communities will be able to convince ICANN[13] to accept new variables.

[12] Wittenburg, P. et al.. (2022). FAIR Digital Object Demonstrators 2021 (final). Zenodo. https://doi.org/10.5281/zenodo.5872645
[13] https://www.icann.org/

For configuration types 11, 12 and 13 PIDs need to be translated to internal IDs and or queries and the used internal variables need to be linked with community wide standards.