FDO Conference 2026

Placeholder Session title

https://doi.org/10.52825/xxxx........ DOI placeholder (WILL BE FILLED IN BY TIB Open Publishing)

Published: (WILL BE FILLED IN BY TIB Open Publishing)

Am I an FDO?

A Comparison across Handle, Nanopublications, and Webby FDOs

Leyla Jael Castro[1,*] https://orcid.org/0000-0003-3986-0510, Jana Böhm[2] https://orcid.org/0009-0004-9802-113X, Ivonne Anders[3] https://orcid.org/0000-0001-7337-3009, Christophe Blanchi[4] https://orcid.org/0000-0003-2277-5176, Beate Krüss[3], Andreas Pfeil[5] https://orcid.org/0000-0001-6575-1022, Heinrich Widmann[3] https://orcid.org/0000-0001-9871-2687

[1]ZB MED - Information Centre for Life Sciences, Cologne, Germany

[2]Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen, Göttingen, Germany

[3]German Climate Computing Center, Hamburg, Germany

[4]Corporation for National Research Initiatives, DONA Foundation

[5]Karlsruhe Institute of Technology, Helmholtz Metadata Collaboration, Research Data Alliance

*Correspondence: ljgarcia@zbmed.de

Abstract. This paper compares three FDO implementations, namely Handle-based FDOs, Nanopublications, and Webby FDOs, analyzing their approach and compliance to the minimum FDO elements to be called an FDO. It presents a comparison on how these implementations deal with the minimum set from the full FDO specification corresponding to a persistent identifier (PID), an FDO record consisting of a collection of attributes, an FDO profile and type. It also includes additional work that would support seamlessly using one implementation to the other to navigate their FDO records.

Keywords: FAIR Digital Objects, FDO Specifications

1. Introduction

In 2022, the FAIR Digital Object (FDO) configuration types presented a collection of 13 ways to organize FDOs [1]. Since then, the FDO specification has matured while some FDO implementation technologies have gained momentum over others, namely Handle-based FDOs, Nanopublicactions, and Webby FDOs. Focusing on the minimum required elements in the current FDO specification, our aim is to find a parallel between these three technologies. Such a parallel would make it easier for FDO clients, e.g., libraries or APIs accessing FDOs, to interact in the same way with FDOs regardless of their underlying implementation.

In this paper, we compare the aforementioned technologies one to another, analyzing their approach and compliance to the minimum elements from the FDO specification for an FDO to be called, for real, an FDO. This minimum set corresponds to a **persistent identifier (PID)**, an **FDO record** consisting of a collection of one or more **attributes** (which are type value pairs), an **FDO profile,** and an **FDO type**. An attribute's type is a reference to an attribute definition (which may be expressed as an FDO) and the attribute's value should be valid according to the attribute definition. The FDO record may include other attributes that fall outside the scope of the FDO requirements and should describe the object that the record represents. An FDO profile defines the set of mandatory and optional attributes that must or may be instantiated in an FDO record compliant to such a profile. An **attribute definition** contains a label and semantic, syntactic or other formal rules wrt the attribute's permitted values. Here we will present how Handle, Nanopublications, and Webby FDOs deal with this minimum set, concluding with our findings and an outlook on future work.

2. Comparison of different approaches covering the FDO Specs

Handle-based FDOs rely on the Handle System [2] for the persistent identification and storage of FDO information. A Handle record is easily used to hold an FDO record with the Handle values in the record used to store FDO attributes such as the FDO profile, FDO type, and data bit-sequence(s) as needed. Attribute definitions and profiles may be implemented in the Data Type Registry (DTR) [3] to support machine actionability of information. Nanopublications [4] are RDF knowledge graphs including metadata on assertions, provenance, and bibliographic information about the Nanopublication itself. The Nanopublication Framework aligns also with the FDO definitions and requirements as shown in [5]. Webby FDOs [6] rely on FAIR Signposting [7] and RO-Crate [8]. FAIR Signposting provides a clear pointer to easily identify where the FDO corresponding to a landing page is located, while RO-Crate facilitates packaging research outputs together with their metadata.

FDO information (including its record, data, attributes, profile, type) is representable with respect to either of the three described FDO implementations. This means, for one and the same FDO information, it should be possible to create a Handle-based FDO, a Nanopublication, and a Webby FDO which carry the same FDO information independent of the implementation. Ideally, there should also be a defined procedure that transforms an FDO represented in one implementation into another without losing any information.

The process described here aims to align each implementation with the others and the FDO specification. The first step is to check how the basic terms from the FDO specifications (e.g., attributes, attribute definitions, profiles, types) are realized in different implementations (and if at all). Afterwards, more in-depth inspections are necessary to check implementations against qualitative specification requirements ("a profile is itself an FDO", "attributes are openly registered", etc.). Finally, a procedure on how to represent "pure" FDO information in each of the implementations needs to be formally defined. Some of the described steps were already part of the FDO Connect project, which defined a mapping between the terms from the FDO specification, Handle-based FDOs, and Nanopublications. Also, FDO operations were implemented which start with Handle-based FDOs respectively Nanopublications and transform it into the corresponding other implementation. We are now making progress aligning these to Webby FDOs. Table 1 presents a side-by-side comparison across the three implementations, detailing how they tackle the main elements of an FDO: PID, record, type, profile, and attribute, value pairs.

Table 1*. A side-by-side comparison of Handle, Nanopublications, and Webby FDOs.*

Implementations	PID system	Format of FDO record and Attributes	Type	Profile	Attribute Definitions and values
Handle	Handle	List of tuples (first element is a PID referencing an attribute definition, second element is a value complying	The type is given as an attribute within the FDO record, if applicable.	Profiles are identified by Handle PIDs and defined in the DTR. They use JSON schema to define a set of mandatory and optional attribute	Attribute definitions are identified by Handle PIDs and defined in the DTR. They have a label (e.g.

		to the attribute definition) FDO records are represented in JSON format		definitions for the FDO. A profile is given as a reference in an attribute in the FDO record.	"author") and a JSON schema to define syntactic validation rules for the value. Values should conform to the rules of the corresponding attribute definition.
Nanopublications	w3id	List of triples (first element refers to the FDO itself, second is a PID referencing an attribute definition, third element is a value complying to the attribute definition) FDO records are represented as RDF triples	The type is given as an attribute within the FDO record, if applicable.	Profiles are implemented as Nanopublications. They always have the attribute fdoc:hasShape that refers to a SHACL shape defining the profile. All FDOs point to their profile via the attribute dct:conformsTo.	FDO attribute definitions correspond to DatatypeProperty / ObjectProperty definitions (in separate Nanopublications). Values should conform to the rules of the corresponding attribute definition.
Webby FDOs	Any PID including w3id, DOIs, and Handles	JSON-LD (i.e., RDF triples)	Type is indicated at the Signposting level	Profiles are referenced using dcterms:conformsTo. An RO-Crate profile [9] must be used. A second community-agreed profile could be used to describe the main entity referred to from the RO-Crate.	Attribute PIDs correspond to properties in a controlled vocabulary, schema.org [10] by default. Values should conform to the rules defined in the controlled vocabulary in use.

3. Conclusions and future work

The three implementations presented here fulfill the FDO implementation and could be used interchangeably. Handle presents FDO records as a table, making it easier for humans to grasp the information, while Nanopublications and Webby FDOs rely on Semantic Web technologies such as ontologies and RDF. The operational level of FDOs still needs better coverage by these three implementations. While Handle is already going in this direction, Nanopublications and Webby FDOs have some more work in this regard. Achieving this operational level will enable concrete research use cases to demonstrate the benefits of FDO implementations through improved interoperability and machine-actionability. Future work should also include, e.g., validators understanding the three implementations so the FDO community can easily, and programmatically, decide whether something is (or not) an FDO. This side-by-side comparison is the first step in understanding commonalities and differences but common tools are needed.

Data availability statement

This submission is not based on data as it is a comparison of three FDO implementations.

Author contributions

All the co-authors contributed to the conceptualization and writing – original draft and review & editing.

Competing interests

The authors declare that they have no competing interests.

Funding

Work corresponding to Webby FDOs done by LJC has been partially supported by the NFDI4DataScience consortium funded by the German Research Foundation (DFG –Deutsche Forschungsgemeinschaft), project number 460234259.

This research has been partially supported by the research program 'Engineering Digital Futures' of the Helmholtz Association of German Research Centers and the Helmholtz Metadata Collaboration Platform (HMC).

References

[1] Larry, L., Karsten, P.-. von G., Ivonne, A., Andreas, P., Alexander, S., Zach, T., & Peter, W. (2022). FDO Forum FDO Configuration Types. https://doi.org/10.5281/zenodo.7825703

[2] HDL.NET® Information Services. "Handle.Net Registry" https://www.handle.net/. (accessed 2025-10-30)

[3] GWDG. "FDO Testbed Typeregistry" https://typeregistry.testbed.pid.gwdg.de/. (accessed 2025-10-30)

[4] NanoPub Net. "Nanopublications" https://nanopub.net/. (accessed 2025-10-30)

[5] Magagna, B., Bonino da Silva Santos, L. O., Kuhn, T., Ferreira Pires, L., & Schultes, E. (2025). Nanopublications as FAIR Digital Object Implementations. *Open Conference Proceedings*, *5*. https://doi.org/10.52825/ocp.v5i.1417

[6] Soiland-Reyes, S., Sefton, P., Leo, S., Castro, L. J., Weiland, C., & Van de Sompel, H. (2025). Practical webby FDOs With RO-Crate and FAIR Signposting: Experiences and Lessons Learned. Open Conference Proceedings, 5. https://doi.org/10.52825/ocp.v5i.1273

[7] Van de Sompel, H. (2023, May 27). FAIR Digital Objects and FAIR Signposting. Zenodo. https://doi.org/10.5281/zenodo.7977333

[8] Soiland-Reyes S, Sefton P, et al. Packaging research artefacts with RO-Crate. Data Science. 2022;5(2):97-138. https://doi.org/10.3233/DS-210053

[9] RO-Crate. "RO-Crate profiles" https://www.researchobject.org/ro-crate/specification/1.2/profiles.html. (accessed 2025-10-30)

[10] Schema.org. "Schema.org" https://schema.org/. (accessed 2025-10-30)