UvA-DARE (Digital Academic Repository)

Evaluating FAIR Digital Object and Linked Data as distributed object systems

Soiland-Reyes, S.; Goble, C.; Groth, P.

DOI
10.48550/arXiv.2306.07436

Publication date
2023

Document Version
Other version

License
CC BY

Link to publication

Citation for published version (APA):
Soiland-Reyes, S., Goble, C., & Groth, P. (2023). *Evaluating FAIR Digital Object and Linked Data as distributed object systems*. (v1 ed.) ArXiv. https://doi.org/10.48550/arXiv.2306.07436

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:28 Sep 2023

arXiv:2306.07436v2 [cs.DC] 17 Jun 2023

Evaluating FAIR Digital Object and Linked Data as distributed object systems

Stian Soiland-Reyes[1,2], Carole Goble[1], and Paul Groth[2]

[1]**Department of Computer Science, The University of Manchester, UK**
[2]**Informatics Institute, Faculty of Science, University of Amsterdam, NL**

Corresponding author:
Stian Soiland-Reyes[1]

Email address: soiland-reyes@manchester.ac.uk

ABSTRACT

FAIR Digital Object (FDO) is an emerging concept that is highlighted by European Open Science Cloud (EOSC) as a potential candidate for building an ecosystem of machine-actionable research outputs. In this work we systematically evaluate FDO and its implementations as a global distributed object system, by using five different conceptual frameworks that cover interoperability, middleware, FAIR principles, EOSC requirements and FDO guidelines themself.
We compare the FDO approach with established Linked Data practices and the existing Web architecture, and provide a brief history of the Semantic Web while discussing why these technologies may have been difficult to adopt for FDO purposes. We conclude with recommendations for both Linked Data and FDO communities to further their adaptation and alignment.

INTRODUCTION

The FAIR principles (Mark D. Wilkinson et al. 2016) encourage sharing of scientific data with machine-readable metadata and the use of interoperable formats, and are being adapted by a wide range of research infrastructures. They have been recognised by the research community and policy makers as a goal to strive for (European Commission 2016). In particular, the European Open Science Cloud (EOSC) has promoted adaptation of FAIR data sharing of data resources across electronic research infrastructures (Mons et al. 2017). The EOSC Interoperability Framework (Corcho, Eriksson, et al. 2021) puts particular emphasis on how interoperability can be achieved technically, semantically, organisationally, and legally – laying out a vision of how data, publication, software and services can work together to form an ecosystem of rich digital objects.

Specifically, the EOSC Interoperability framework highlights the emerging FAIR Digital Object (FDO) concept (Schultes and Wittenburg 2019) as a possible foundation for building a semantically interoperable ecosystem to fully realise the FAIR principles beyond individual repositories and infrastructures. The FDO approach has great potential, as it proposes strong requirements for identifiers, types, access and formalises interactive operations on objects.

In other discourse, Linked Data (Bizer et al. 2009) has been seen as an established set of principles based on Semantic Web technologies that can achieve the vision of the FAIR principles (Bonino Da Silva Santos et al. 2016; Hasnain and Rebholz-Schuhmann 2018). Yet regular researchers and developers of emerging platforms for computation and data management are reluctant to adapt such a "FAIR Linked Data approach" fully (Verborgh and Vander Sande 2020), opting instead for custom in-house models and JSON-derived formats from RESTful Web services (Meroño-Peñuela et al. 2021a; Neumann et al. 2021). While such focus on simplicity allows for rapid development and highly specialised services, it raises wider concerns about interoperability (Turcoane 2014; S. R. Wilkinson et al. 2022).

One challenge that may, perhaps counter-intuitively, steer developers towards a not-invented-here mentality (Stefi 2015; Stefi and Hess 2015) when exposing their data on the Web is the heterogeneity and apparent complexity of Semantic Web approaches themselves (Meroño-Peñuela et al. 2021b).

These approaches – FDO and Linked Data – thus, form two of the major avenues for allowing developers and the wider research community to achieve the goal of FAIR data. Given their importance, in this article, we aim to compare FAIR Digital Objects with Linked Data and the Web architecture in the context of the discourse around FAIR data.

Concretely, the contribution of this paper is **a systematic comparison between FDO and Linked Data using 5 different conceptual frameworks** that capture different perspectives on interoperability and readiness for implementation.

BACKGROUND AND RELATED WORK

In the following, we discuss the related work with respect to FAIR Digital Objects and Linked Data. We do so by looking through the lens of development of these technologies over time, including future directions.

FAIR Digital Object

The concept of **FAIR Digital Objects** (Schultes and Wittenburg 2019) has been introduced as way to expose research data as active objects that conform to the FAIR principles (Mark D. Wilkinson et al. 2016). This builds on the *Digital Object* (DO) concept (Kahn and Wilensky 2006), first introduced by Kahn and Wilensky (1995) as a system of *repositories* containing *digital objects* identified by *handles* and described by *metadata* which may have references to other handles. DO was the inspiration for the *ITU-T X.1255* (2013) framework which introduced an abstract *Digital Entity Interface Protocol* for managing such objects programmatically, first realised by the Digital Object Interface Protocol (DOIP) (Reilly 2009).

In brief, the structure of a FAIR Digital Object (FDO) is to, given a *persistent identifier* (PID) such as a DOI, resolve to a *PID Record* that gives the object a *type* along with a mechanism to retrieve its *bit sequences*, *metadata* and references to further programmatic *operations*. The type of an FDO (itself an FDO) defines attributes to semantically describe and relate such FDOs to other concepts (typically other FDOs referenced by PIDs). The premise of systematically building an ecosystem of such digital objects is to give researchers a way to organise complex digital entities, associated with identifiers, metadata, and supporting automated processing (Wittenburg, Strawn, et al. 2019).

Recently, FDOs have been recognised by the European Open Science Cloud (EOSC) as a suggested part of its Interoperability Framework (Corcho, Eriksson, et al. 2021), in particular for deploying active and interoperable FAIR resources that are *machine actionable*. Development of the FDO concept continued within Research Data Alliance (RDA) groups and EOSC projects like GO-FAIR, concluding with a set of guidelines for implementing FDO (Bonino et al. 2019). The FAIR Digital Objects Forum has since taken over the maturing of FDO through focused working groups which have currently drafted several more detailed specification documents (see *Next steps for FDO* on the following page).

FDO approaches

FDO is an evolving concept. A set of FDO Demonstrators (Wittenburg, Anders, et al. 2022) highlight how current adapters are approaching implementations of FDO from different angles:

- Building on the Digital Object concept, using the simplified *DOIPV2.0* (2018) specification, which detail how to exchange JSON objects through a text-based protocol[1] (usually TCP/IP over TLS). The main DOIP operations are retrieving, creating and updating digital objects. These are mostly realised using the reference implementation Cordra (Tupelo-Schneck, Robert and Lannom 2022). FDO types are registered in the local Cordra instance, where they are specified using JSON Schema (Wright et al. 2022) and PIDs are assigned using the Handle system. Several type registries have been established.
- Following a Linked Data approach, but using the DOIP protocol, e.g. using JSON-LD and schema.org within DOIP in Materal Sciences archives (Riccardi et al. 2022).
- Approaching the FDO principles from existing Linked Data practices on the Web, e.g. WorkflowHub use of RO-Crate and schema.org (Soiland-Reyes, Sefton, et al. 2022).

From this it becomes apparent that there is a potentially large overlap between the goals and approaches of FAIR Digital Objects and Linked Data, which we will cover on page 4.

[1] For a brief introduction to DOIP 2.0, see CNRI (2023a)

Next steps for FDO

The FAIR Digital Object Forum (*FDO* 2022) working groups have prepared detailed requirement documents (*FDO Specs* 2022) setting out the path for realising FDOs, named *FDO Recommendations*. As of 2023-06-17, most of these documents are open for public review, while some are still in draft stages for internal review. As these documents clarify the future aims and focus of FAIR Digital Objects (Lannom, Schwardmann, Christophe Blanchi, et al. 2022), we provide a brief summary of each:

FAIR Digital Object Overview and Specifications (FDO-Overview-PEN-2.0) is a comprehensive overview of FAIR Digital Object specifications listed below. It serves as a primer that introduces FDO concepts and the remaining documents. It is accompanied by an FDO Glossary (Broeder and Wittenburg 2022).

The **FDO Forum Document Standards** (WD-DocProcessStd-1.1) documents the recommendation process within the forum, starting at *Working Draft* (WD) status within the closed working group and later within the open forum, then *Proposed Recommendation* (PR) published for public review, finalised as *FDO Forum Recommendation* (REC) following any revisions. In addition, the forum may choose to *endorse* existing third-party notes and specifications.

The **FDO Requirement Specifications** (PR-RequirementSpec-3.0) is an update of (Bonino et al. 2019) as the foundational definition of FDO. This sets the criteria for classifying an digital entity as a FAIR Digital Object, allowing for multiple implementations. The requirements shown in Table 3 on page 11 are largely equivalent, but in this specification clarified with references to other FDO documents.

The **Machine actionability** (PR-MachineActionDef-2.2) sets out to define what is meant by *machine actionability* for FDOs. *Machine readable* is defined as elements of bit-sequences defined by structural specification, *machine interpretable* elements that can be identified and related with semantic artefacts, while *machine actionable* are elements with a type with operations in a symbolic grammar. The document largely describes requirements for resolving an FDO to metadata, and how types should be related to possible operations.

Configuration Types (PR-ConfigurationTypes-2.1) classifies different granularities for organising FDOs in terms of PIDs, PID Records, Metadata and bit sequences, e.g. as a single FDO or several daisy-chained FDOs. Different patterns used by current DOIP deployments are considered, as well as FAIR Signposting (Van de Sompel, Klein, et al. 2022).

PID Profiles & Attributes (PR-PIDProfileAttributes-2.1) specifies that PIDs must be formally associated with a *PID Profile*, a separate FDO that defines attributes required and recommended by FDOs following said profile. This forms the *kernel attributes*, building on recommendations from RDA's *PID Information Types* working group (Weigel et al. 2018). This document makes a clear distinction between a minimal set of attributes needed for PID resolution and FDO navigation, which needs to be part of the *PID Record* (Islam 2023), compared with a richer set of more specific attributes as part of the *metadata* for an FDO, possibly represented as a separate FDO.

Kernel Attributes & Metadata (PR-KernelAtributues-2.0) elaborates on categories of FDO Mandatory, FDO Optional and Community Attributes, recommending kernel attributes like `dateCreated`, `ScientificDomain`, `PersistencePolicy`, `digitalObjectMutability`, etc. This document expands on RDA Recommendation on PID Kernel Information (Weigel et al. 2018). It is worth noting that both documents are relatively abstract and do not establish PIDs or namespaces for the kernel attributes.

Granularity, Versioning, Mutability (PR-Granularity-2.2) considers how granularity decisions for forming FDOs must be agreed by different communities depending on their pragmatic usage requirements. The affect on versioning, mutability and changes to PIDs are considered, based on use cases and existing PID practices.

DOIP Endorsement Request (PED-DOIPEndorsement-1.1) is an endorsement of the DOIP v2.0 (*DOIPV2.0* 2018) specification as a potential FDO implementation, as it has been applied by several institutions (Wittenburg, Anders, et al. 2022). The document proposes that DOIP shall be assessed for completeness against FDO – in this initial draft this is justified as *"we can state that DOIP is compliant with the FDO specification documents in process"* (the documents listed above).

Upload of FDO (PEN-FDO-Upload) illustrates the operations for uploading an FDO to a repository, what checks it should do (for instance conformance with the PID Profile, if PIDs resolve). ResourceSync (*ANSI Z39.99* 2017) is suggested as one type of service to list FDOs. This document highlights potential practices by repositories and their clients, without adding any particular requirements.

Typing FAIR Digital Objects (PR-TypingFDOs-2.0) defines what *type* means for FDOs, primarily to enable machine actionability and to define an FDO's purpose. This document lays out requirements for how *FDO Types* should themselves be specified as FDOs, and how an *FDO Type Framework* allows organising and locating types. Operations applicable to an FDO is not predefined for a type, however operations naturally will require certain FDO types to work. How to define such FDO operations is not specified.

Implementation of Attributes, Types, Profiles and Registries (WD-ImplAttributesTypesProfiles) details how to establish FDO registries for types and FDO profiles, with their association with PID systems. This document suggest policies and governance structures, together with guidelines for implementations, but without mandating any explicit technology choices. Differences in use of attributes are examplified using FDO PIDs for scientific instruments, and the proto-FDO approach of DARIAH-DE (Schwardmann and Kálmán 2022).

See bibliography on page 39 for the citation per document above. It is worth pointing out that, except for the DOIP endorsement, all of these documents are conceptual, in the sense that they permit any technical implementation of FDO, if used according to the recommendations. Existing FDO implementations (Wittenburg, Anders, et al. 2022) are thus not fully consolidated in choices such as protocols, type systems and serialisations – this divergence and corresponding additional technical requirements mean that FDOs are not yet in a single ecosystem.

From the Semantic Web to Linked Data

In order to describe *Linked Data* as it is used today, we'll start with an (opinionated) description of the evolution of its foundation, the *Semantic Web*.

A brief history of the Semantic Web

The **Semantic Web** was developed as a vision by Tim Berners-Lee (Berners-Lee and Fischetti 1999), at a time that the Web had already become widely established for information exchange, being a global set of hypermedia documents which are cross-related using universal links in the form of URLs. The foundations of the Web (e.g. URLs, HTTP, SSL/TLS, HTML, CSS, ECMAScript/JavaScript, media types) were standardised by W3C, Ecma, IETF and later WHATWG. The goal of Semantic Web was to further develop the machine-readable aspects of the Web, in particular adding *meaning* (or semantics) to not just the link relations, but also to the *resources* that the URLs identified, and for machines thus being able to meaningfully navigate across such resources, e.g. to answer a particular query.

Through W3C, the Semantic Web was realised with the Resource Description Framework (RDF) (Schreiber and Raimond 2014) that used *triples* of subject-predicate-object statements, with its initial serialisation format (Lassila and Swick 1999) being RDF/XML (XML was at the time seen as a natural data-focused evolution from the document-centric SGML and HTML).

While triple-based knowledge representations were not new (Stanczyk 1987), the main innovation of RDF was the use of global identifiers in the form of URIs[2] as the primary identifier of the *subject* (what the statement is about), *predicate* (relation/attribute of the subject) and *object* (what is pointed to). By using URIs not just for documents[3], the Semantic Web builds a self-described system of types and properties, where the meaning of a relation can be resolved by following its hyperlink to the definition within a *vocabulary*. By applying these principles as well to any kind of resource that could be described at a URL, this then forms a global distributed Semantic Web.

The early days of the Semantic Web saw fairly lightweight approaches with the establishment of vocabularies such as FOAF (to describe people and their affiliations) and Dublin Core (for bibliographic data). Vocabularies themselves were formalised using RDFS or simply as human-readable HTML web pages defining each term. The main approach of this *Web of Data* was that a URI identified a *resource* (e.g. an author) with a HTML *representation* for human readers, along with a RDF representation for

[2]URIs (Berners-Lee, Roy T. Fielding, et al. 2005) are generalised forms of URLs that include locator-less identifiers such as ISBN book numbers (URNs). The distinction between locator-full and locator-less identifiers have weakened in recent years (OCLC 2010), for instance DOI identifiers now are commonly expressed with the prefix `https://doi.org/` rather than as URNs with `info:doi`: given that the URL/URN gap has been bridged by HTTP resolvers and the use of Persistent Identifiers (PIDs) (Juty et al. 2011). RDF 1.1 formats use Unicode to support *IRI*s (Dürst and Suignard 2005), which extends URIs to include international characters and domain names.

[3]URIs can also identify *non-information resources* for any kind of physical object (e.g. people), such identifiers can resolve with `303 See Other` redirections to a corresponding *information resources* (Sauermann et al. 2008).

machine-readable data of the same resource. By using *content negotiation* in HTTP[4], the same identifier could be used in both views, avoiding `index.html` vs `index.rdf` exposure in the URLs. The concept of *namespaces* gave a way to give a group of RDF resources with the same URI base from a Semantic Web-aware service a common *prefix*, avoiding repeated long URLs.

The mid-2000s saw large academic interest and growth of the Semantic Web, with the development of more formal representation system for ontologies, such as OWL (W3C OWL Working Group 2012), allowing complex class hierarchies and logic inference rules following *open world* paradigm. More human-readable syntaxes for RDF such as Turtle evolved at this time, and conferences such as ISWC (Horrocks and Hendler 2002) gained traction, with a large interest in knowledge representation and logic systems based on Semantic Web technologies evolving at the same time.

Established Semantic Web services and standards include: SPARQL (SPARQL WG 2013) (pattern-based triple queries), named graphs (Wood et al. 2014) (triples expanded to *quads* to indicate statement source or represent conflicting views), triple/quad stores (graph databases such as OpenLink Virtuoso, GraphDB, 4Store), mature RDF libraries (including Redland RDF, Apache Jena, Eclipse RDF4J, RDFLib, RDF.rb, rdflib.js), and graph visualisation.

RDF is one way to implement *knowledge graphs*, a system of named edges and nodes[5] (Nurdiati and Hoede 2008), which when used to represent a sufficiently detailed model of the world, can then be queried and processed to answer detailed research questions. The creation of RDF-based knowledge graphs grew particularly in fields like bioinformatics, e.g. for describing genomes and proteins (Goble and Stevens 2008; Williams et al. 2012). In theory, the use of RDF by the life sciences would enable interoperability between the many data repositories and support combined views of the many aspects of bio-entities – however in practice most institutions ended up making their own ontologies and identifiers, for what to the untrained eye would mean roughly the same. One can argue that the toll of adding the semantic logic system of rich ontologies meant that small, but fundamental, differences in opinion (e.g. *should a gene identifier signify just the particular DNA sequence letters, or those letters as they appear in a particular position on a human chromosome?*) lead to large differences in representational granularity, and thus needed different identifiers.

Facing these challenges, thanks to the use of universal identifiers in the form of URIs, *mappings* could retrospectively be developed not just between resources, but also across vocabularies. Such mappings can be expressed themselves using lightweight and flexible RDF vocabularies such as SKOS (Isaac and Summers 2009) (e.g. `dct:title skos:closeMatch schema:name` to indicate near equivalence of two properties). Exemplifying the need for such cross-references, automated ontology mappings have identified large potential overlaps like 372 definitions of `Person` (Hu et al. 2011).

The move towards *Open Science* data sharing practices did from the late 2000s encourage knowledge providers to distribute collections of RDF descriptions as downloadable *datasets* [6], so that their clients can avoid thousands of HTTP requests for individual resources. This enabled local processing, mapping and data integration across datasets (e.g. Open PHACTS (Groth et al. 2014)), rather than relying on the providers' RDF and SPARQL endpoints (which could become overloaded when handling many concurrent, complex queries).

With these trends, an emerging problem was that adopters of the Semantic Web primarily utillised it as a set of graph technologies, with little consideration to existing Web resources. This meant that links stayed mainly within a single information system, with little URI reuse even with large term overlaps (Kamdar et al. 2017). Just like *link rot* affect regular Web pages and their citations from scholarly communication (Klein et al. 2014), for a majority of described RDF resources in the Linked Open Data (LOD) Cloud's gathering of more than thousand datasets, unfortunately do not actually link to (still) downloadable (*dereferenceable*) Linked Data (Polleres et al. 2020). Another challenge facing potential adopters is the plethora of choices, not just to navigate, understand and select to reuse the many possible vocabularies and ontologies (Carriero et al. 2020), but also technological choices on RDF serialisation (at least 7 formats), type system (RDFS (Guha and Brickley 2014), OWL (W3C OWL Working Group 2012), OBO (Tirmizi et al. 2011), SKOS (Isaac and Summers 2009)), and deployment challenges (Sauermann et al. 2008) (e.g. hash vs slash in namespaces, HTTP status codes and PID redirection strategies).

[4] `https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_negotiation`

[5] In RDF, each triple represent an edge that is named using its property URI, and the nodes are subject/object as URIs, blank nodes or (for objects) typed literal values (Schreiber and Raimond 2014).

[6] *Datasets* that distribute RDF graphs should not be confused with *RDF Datasets* used for partitioning *named graphs*.

Linked Data: Rebuilding the Web of Data

The **Linked Data** (LD) concept (Bizer et al. 2009) was kickstarted as a set of best practices (Berners-Lee 2006) to bring the Web aspect of the Semantic Web back into focus. Crucial to Linked Data is the *reuse of existing URIs*, rather than making new identifiers. This means a loosening of the semantic restrictions previously applied, and an emphasis on building navigable data resources, rather than elaborate graph representations.

Vocabularies like schema.org evolved not long after, intended for lightweight semantic markup of existing Web pages, primarily to improve search engines' understanding of types and embedded data. In addition to several such embedded *microformats* (*OGP* 2022; Sporny, Herman, et al. 2015; *Microdata* 2023), we find JSON-LD (Sporny, Longley, et al. 2020) as a Web-focused RDF serialisation that aims for improved programmatic generation and consumption, including from Web applications. JSON-LD is as of 2023-05-18 used[7] by 45% of the top 10 million websites (W3Techs 2023).

Recently there has been a renewed emphasis to improve the *Developer Experience* (Verborgh 2018) for consumption of Linked Data, for instance RDF Shapes – expressed in SHACL (Kontokostas and Knublauch 2017) or ShEx (Baker and Prud'hommeaux 2019) – can be used to validate RDF Data (Gayo et al. 2017; Thornton et al. 2019) before consuming it programmatically, or reshaping data to fit other models. While a varied set of tools for Linked Data consumptions have been identified, most of them still require developers to gain significant knowledge of the underlying Semantic Web technologies, which hampers adaption by non-LD experts (Klímek et al. 2019), which then tend to prefer non-semantic two-dimensional formats such as CSV files.

A valid concern is that the Semantic Web research community has still not fully embraced the Web, and that the "final 20%" engineering effort is frequently overlooked in favour of chasing new trends such as Big Data and AI, rather than making powerful Linked Data technologies available to the wider groups of Web developers (Verborgh and Vander Sande 2020). One bridging gap here by the Linked Data movement has been "Linked Data by stealth" approaches such as structured data entry spreadsheets powered by ontologies (Katy Wolstencroft et al. 2011), the use of Linked Data as part of REST Web APIs (Page et al. 2011), and as shown by the big uptake by publishers to annotate the Web using schema.org (Bernstein et al. 2016), with vocabulary use patterns documented by copy-pastable JSON-LD examples, rather than by formalised ontologies or developer requirements to understand the full Semantic Web stack.

Linked Data provides technologies that have evolved over time to satisfy its primary purpose of data interoperability. The needs to embrace the Web and developer experience have been central lessons learned. In contrast, FDO is a new approach with many different potential paths forward, and having a partial overlap with the aims of Linked Data.

METHOD

Our main motivation for this article is to investigate how FAIR Digital Objects may differ from the learnt experiences of Linked Data and the Web. We also aim to reflect back from FDO's motivation of machine-actionability to consider the Web as a distributed computational system.

To better understand the relationship between the FDO framework and other existing approaches, we use the following for analysis:

1. An Interoperability Framework and Distributed Platform for Fast Data Applications (Delgado 2016), which proposes quality measurements for comparing how frameworks support interoperability, particularly from a service architectural view.
2. The FAIR Digital Object guidelines (Bonino et al. 2019), validated against its current implementations for completeness.
3. A Comparison Framework for Middleware Infrastructures (Zarras 2004), which suggest dimensions like openness, performance and transparency, mainly focused on remote computational methods.
4. Cross-checks against RDA's FAIR Data Maturity Model (Bahim et al. 2020) to find how the FAIR principles are achieved in FDO, in particular considering access, sharing and openness.

[7] Presumably this large uptake of JSON-LD is mainly for the purpose of Search Engine Optimisation (SEO), with typically small amounts of metadata which may not constitute Linked Data as introduced above, however this deployment nevertheless constitute machine-actionable structured data.

5. EOSC Interoperability Framework (Corcho, Eriksson, et al. 2021) which gives recommendations for technical, semantic, organisational and legal interoperability, particularly from a metadata perspective.

The reason for this wide-ranged comparison is to exercise the different dimensions that together form FAIR Digital Objects: Data, Metadata, Service, Access, Operations, Computation. We have left out further considerations on type systems, persistent identifiers and social aspects as principles and practices within these dimensions are still taking form within the FDO community (as detailed on page 3).

Some of these frameworks invite a comparison on a conceptual level, while others relate better to implementations and current practices. For conceptual comparisons we consider FAIR Digital Objects and the Web broadly. For implementations, we contrast the main FDO realisation using the DOIPv2 protocol (*DOIPV2.0* 2018) against Linked Data as implemented in general practice[8].

Considering FDO/Web as interoperability framework for Fast Data

The Interoperability Framework for Fast Data Applications (Delgado 2016) categorises interoperability between applications along 6 strands, covering different architectural levels: from *symbiotic* (agreement to cooperate) and *pragmatic* (ability to choreograph processes), through *semantic* (common understanding) and *syntactic* (common message formats), to low-level *connective* (transport-level) and *environmental* (deployment practices).

We have chosen to investigate using this framework as it covers the higher levels of the OSI Model (Stallings 1990) better with regards to automated machine-to-machine interaction (and thus interoperability), which is a crucial aspect of the FAIR principles. In Table 1 on the next page we use the interoperability framework to compare the current FAIR Digital Object approach against the Web and its Linked Data practices.

Based on the analysis shown in Table 1, we draw the following conclusions:

The Web has already showed us how one can compose workflows of hetereogeneous Web Services (Katherine Wolstencroft et al. 2013). However, this is mostly done via developer or human interaction (Lamprecht et al. 2021). Similiarly, FDO does not enable automatic composition because operation semantics are not well defined. There is a question as to whether the extensive documentation and broad developer usage that is available for Web APIs could potentially be utilised for FDO.

A difference between Web technologies and FDO is the stringency of the requirements for both syntax and semantics. Whereas the Web allows many different syntactic formats (e.g. from HTML to XML, PDFs), FDO realised with DOIP requires JSON. On the semantic front, FDO mandates that every object have a well-defined type and structured form. This is clearly not the case on the Web.

In terms of connectivity and the deployment of applications, the Web has a plethora of software, services, and protocols that are widely deployed. These have shown interoperability. The Web standards bodies (e.g. IETF and W3C) follow the OpenStand principles (*OpenStand* 2017) to embrace openness, transparency, and broad consensus. In contrast, FDO has a small number of implementations and corresponding protocols, although with a growing community, as evidenced at the first international FDO conference (Loo 2022). This is not to say that it is not worth developing further Handle+DOIP implementations in the future, but we note that the current FDO functionality can easily be implemented using Web technologies, even as DOIP-over-HTTP (CNRI 2023b).

It is also a question as to whether a highly constrained protocol revolving around persistent identifiers is in fact necessary. For example, DOIs are mostly resolved on the web using HTTP redirects with the common `https://doi.org/` prefix, hiding their Handle nature as an implementation detail (DOI 2019).

[8] For further background on FDO implemented with Linked Data see (Bonino da Silva Santos et al. 2022; Soiland-Reyes, Castro, et al. 2022)

Table 1. Considering FDO and Web according to the quality levels of the Interoperability Framework for Fast Data (Delgado 2016).

Quality	FDO w/ DOIP	Web w/ Linked Data
Symbiotic: *to what extent multiple applications can agree to interact, align, collaborate or cooperate*	The purpose of FDO is to enable federated machine actionable digital objects for scholarly purposes, in practice this also requires agreement of compatibility between FDO types. FDO encourages research communities to develop common type registries to be shared across instances. In current DOIP practice, each service have their own types, attributes and operations. The wider symbiosis is consistent use of PIDs.	The Web is loosely coupled and encourages collaboration and linking by URL. In practice, REST APIs (Roy Thomas Fielding 2000) end up being mandated centrally by dominant (often commercial) providers (Roy T. Fielding, Taylor, et al. 2017), and the clients are required to use each API as-is with special code per service. Use of Linked Data enables common tooling and semantic mapping across differences.
Pragmatic: *using interaction contracts so processes can be choreographed in workflows*	FDO types and operations enable detailed choreography (Canonical Workflows; CWFR Group (2021)). `0.TYPE/DOIPOperation` has lightweight definition of operation, `0.DOIP/Request` or `0.DOIP/Response` may give FDO Type or any other kind of "specifics" (incl. human readable docs). Semantics/purpose of operations not formalised (similar operations can be grouped with `0.DOIP/OperationReference`).	"Follow your nose" crawler navigation, which may lead to frequent dead ends. Operational composition, typically within a single API provider, documented by OpenAPI 3 (D. Miller et al. 2021), schema.org Actions (*Schema.Org Actions* 2022), WSDL/SOAP (Liu and Booth 2007), but frequently just as human-readable developer documentation with examples.
Semantic: *ensuring consistent understanding of messages, interoperability of rules, knowledge and ontologies*	FDO semantic enable navigation and typing. Every FDO has a type. Types maintained in FDO Type registries, which may add additional semantics, e.g. the ePIC PID-InfoType for Model. No single type semantic, Type FDOs can link to existing vocabularies & ontologies. JSON-LD used within some FDO objects (e.g. DISSCO Digital Specimen, NIST Material Science schema) (Wittenburg, Anders, et al. 2022)	Lightweight HTTP semantics for authenticity/navigation. Semantic Type not commonly expressed on PID/header level, may be declared within Linked Data metadata. Semantic of type implied by Linked Data formats (e.g. OWL2, RDFS), although choice of type system may not be explicit.
Syntactic: *serialising messages for digital exchange, structure representation*	DOIP serialise FDOs as JSON, metadata commonly use JSON, typed with JSON Schema. Multiple byte stream attachments of any media type.	Textual HTTP headers (including any signposting), single byte stream of any media type, e.g. Linked Data formats (JSON-LD, Turtle, RDF/XML) or embedded in document (HTML with RDFa, JSON-LD or Microdata). XML was previously the main syntax used by APIs, JSON is now dominant.

Quality	FDO w/ DOIP	Web w/ Linked Data
Connective: *transferring messages to another application, e.g. wrapping in other protocols*	*DOIPV2.0* (2018) is transport-independent, commonly TLS TCP/IP port 9000, DOIP over HTTP (CNRI 2023b)	HTTP/1.1, TCP/IP port 80 (Roy T. Fielding, Gettys, et al. 1999); HTTP/1.1+TLS, TCP/IP 443 (Rescorla 2000); HTTP/2, as HTTP/1* but binary (Belshe et al. 2015); HTTP/3, like HTTP/2+TLS but UDP (Bishop 2022)
Environmental: *how applications are deployed and affected by its environment, portability*	Main DOIP implementation is *Cordra*, which can be single-instance or distributed. Cordra storage backends include file system, S3, MongoDB (itself scalable). Unique DOIP protocol can be hard to add to existing Web application frameworks, although proxy services have been developed (e.g. B2SHARE adapter).	HTTP services widely deployed in a myriad of ways, ranging from single instance servers, horizontally & vertically scaled application servers, to multi-cloud Content-Delivery Networks (CDN). Current scalable cloud technologies for Web hosting may not support HTTP features previously seen as important for Semantic Web, e.g. content negotiation and semantic HTTP status codes.

Mapping of Metamodel concepts

The Interoperability Framework for Fast Data also provides a brief *metamodel* which we use in Table 2 to map and examplify corresponding concepts in FDO's DOIP realization and the Web using HTTP semantics (Roy T. Fielding, Nottingham, et al. 2022).

From this mapping we can identify the conceptual similarities between DOIP and HTTP, often with common terminology. Notable are that neither DOIP or HTTP have strong support for transactions (explored further on page 14), as well that HTTP has poor direct support for processes, as the Web is primarily stateless by design.

Table 2. Mapping the Metamodel concepts from the Interoperability Framework for Fast Data (Delgado 2016) to equivalent concepts for FDO and Web.

Metamodel concept	FDO/DOIP concept	Web/HTTP concept
Resource	FDO/DO	Resource
Service	DOIP service	Server/endpoint
Transaction	(not supported)	Conditional requests, `409 Conflict`
Process	Extended operations	(primarily stateless), `100 Continue`, `202 Accepted`
Operation	DOIP Operation	Method, query parameters
Request	DOIP Request	Request
Response	DOIP Response	Response
Message	Segment, `requestId`	Message, Representation
Channel	DOIP Transport protocol (e.g. TCP/IP, TLS). JSWS signatures.	TCP/IP, TLS, UDP
Protocol	DOIP 2.0, ++	HTTP/1.1, HTTP/2, HTTP/3
Link	PID/Handle	URL

Assessing FDO implementations

The FAIR Digital Object guidelines (Bonino et al. 2019) sets out recommendations for FDO implementations. Note that the proposed update to FDO specification (PR-RequirementSpec-3.0) clarifies these definitions with equivalent identifiers[9] and relates them to further FDO requirements such as FDO Data Type Registries.

In Table 3 on the following page we evaluate completeness of the guidelines in two current FDO realizations, using DOIPv2 (*DOIPV2.0* 2018) and using Linked Data Platform (Speicher et al. 2015), as proposed by Bonino da Silva Santos et al. (2022).

A key observation from this is that simply using DOIP does not achieve many of the FDO guidelines. Rather the guidelines set out how a protocol like DOIPs should be used to achieve FAIR Digital Object goals. The DOIP Endorsement (PED-DOIPEndorsement-1.1) set out that to comply, DOIP must be used according to the set of FDO requirement documents (details on page 3), and notes *Achieving FDO compliance requires more than DOIP and full compliance is thus left to system designers*. Likewise, a Linked Data approach will need to follow the same requirements to actually comply as an FDO implementation.

From our evaluation, we can observe:

- G1 and G2 call for stability and trustworthiness. While the foundations of both DOIP and Linked Data approaches are now well established – the FDO requirements and in particular how they can be implemented are still taking shape and subject to change.
- Machine actionability (G4, G6) is a core feature of both FDOs and Linked Data. Conceptually they differ in the which way types and operations are discovered, with FDO seemingly more rigorous. In practice, however, we see that DOIP also relies on dynamic discovery of operations and that operation expectations for types (FDOF7) have not yet been defined.
- FDO proposes that types can have additional operations beyond CRUD (FDOF5, FDOF6), while Linked Data mainly achieves this with RESTful patterns using CRUD on additional resources, e.g. `order/152/items`. These are mainly stylistics but affect the architectural view – FDOs have more of an object-oriented approach.
- FDO puts strong emphasis on the use of PIDs (FDOF1, FDOF2, FDOF3, FDOF5), but in current practice DOIP use local types, local extended operations (FDOF5) and attributes (FDOF4) that are not bound to any global namespace.
- Linked Data have a strong emphasis on semantics (FDOF8), and metadata schemas developed by community agreements (FDOF10). FDO types need to be made reusable across servers.
- While FDO recommends nested metadata FDOs (FDOF8, FDOF9), in practice this is not found (or linked with custom keys), particularly due to lack of namespaces and the favouring of local types rather than type/property re-use. Linked Data frequently have multiple representations, but often not sufficiently linked (link relation `alternate` (Nottingham 2017)) or related (`prov:specializationOf` from Lebo, McGuinness, et al. (2013)).
- FDO collections are not yet defined for DOIP, while Linked Data seemingly have too many alternatives. LDP has specific native support for containers.
- Tombstones for deleted resources are not well supported, nor specified, for either approach, although the continued availability of metadata when data is removed is a requirement for FAIR principles (see RDA-A2-01M in Table 5 on page 21).
- DOIP supports multiple chunks of data for an object (FDOF3), while Linked Data can support content-negotiation. In either case it can be unclear to clients what is the meaning or equivalence of any additional chunks.

[9] Newer (PR-RequirementSpec-3.0) renames `FDOF*` to `FDOR*` but follows same ordering.

Table 3. Checking FDO guidelines Bonino et al. 2019; PR-RequirementSpec-3.0 against its current implementations as DOIP *DOIPV2.0* 2018 and Linked Data Platform (LDP) Bonino da Silva Santos et al. 2022, with suggestions for required additions.

FDO Guideline	DOIP 2.0	FDO suggestions	Linked Data Platform	LDP suggestion
G1: *invest for many decades*	Handle system stable for 20 years, DOIP 2.0 since 2017.	Ensure FDO types will not be protocol-bound as DOIP might be updated/replaced	HTTP stable for 30 years, Semantic Web for 20 years. `http://` URIs mostly replaced by `https://`.	Keep flexibility of RDF serialisation formats which may change more frequently
G2: *trustworthiness*	DOI/Handle trusted by all major academic publishers and data repositories. DOIP relatively unknown, in effect only one implementation.	Further promote DOIP and justify its benefits. Build tutorials and OSI open source implementations. Standardise DOIP-over-HTTP alternative.	JSON-LD used by half of all websites (W3Techs 2023), however previous bad experiences with Semantic Web may deter adopters	Ensure simplicity for end developers, rather than semantic overengineering. Example-driven documentation.
G3: *follows FAIR principles*	See Table 5 on page 19	Ensure all FAIR principles are covered, build complete examples.	Touched briefly, see Table 5 on page 19	Add explicit expression to show each FAIR principle fulfilled.
G4: *machine actionability*	CRUD and extension operations dynamically listed (see Table 4 on page 15)	Specify which operations should work for a given type, to reduce need for dynamic lookup. Specify input/output expectations formally (e.g. JSON Schema).	HTTP CRUD operations, Open API (see Table 4 on page 15)	Document operations so client can make subsequent HTTP calls.
G5: *abstraction principle*	Handle PIDs as abstraction base. DOIP operations can use any transport protocol.	Document transport protocols as FDOs, recommend which transport to use.	URI as abstraction base. Does not specify PID requirements.	Give stronger deployment recommendations.
G6: *stable binding between entities*	Machine-navigation through PIDs and operations specified per type. Unclear when metadata field is a PID or plain text.	Make datatype of fields explicit to support navigation.	Machine-navigation through URIs via properties and types. Unclear when URI should be followed or is just identifier, but always distinct from plain text.	
G7: *encapsulation*	Operations discovered at runtime (`0.DOIP/Op.ListOperations`).	Allow method discovery by type FDOs in advance (PR-TypingFDOs-2.0).	HTTP methods discovered at runtime (`OPTIONS`), indempotent methods attempted directly. Unsupported methods reported using LDP constraints to human-readable text.	Declare supported methods in advance, e.g. OpenAPI (D. Miller et al. 2021)
G8: *technology independence*	In theory independent, in reality depends on single implementations of Handle system and DOIP	Encourage open source DOIP testbeds and lighter reference implementations	Multiple HTTP implementations, multiple LDP implementations. No FDOF implementations.	Develop demonstrator of FDOF usage based on existing LDP server.

FDO Guideline	DOIP 2.0	FDO suggestions	Linked Data Platform	LDP suggestion
G9: *standard compliance*	Handle (Sun, Lannom, et al. 2003), DOIP (*DOIPV2.0* 2018). FDO requirements not standardised yet.	Formalise standard process of FDO requirements (WD-DocProcessStd-1.1)	HTTP, LDP. However FDOF is not yet standardised.	Formalise FDOF from FDOF-SEM working group.
FDOF1: *PID as basis*	Extensive use of Handle system.	Clarify how local testing handles can be used during development, how “temporary” FDOs should evolve (PR-PIDProfileAttributes-2.1). Register `0.DOIP/*` and `0.FDO/*` as actual PIDs.	HTTP URLs as basis for identifiers, but they are frequently not persistent.	Add strong guidance for PID services like w3id and persistence policies (McMurry et al. 2017).
FDOF2: *PID record w/ type*	Unspecified how to resolve from Handle to DOIP Service (!), in practice `10320/loc`, `0.TYPE/DOIPService`, `URL`, `URL_REPLICA`	Document requirements for PID Record	w3id/purl PIDs redirect without giving any metadata. Datacite DOIs content-negotiate to give registered metadata.	Add FAIR Signposting (Van de Sompel, Klein, et al. 2022) at PID provider for minimal PID record
FDOF3: *PID resolvable to bytestream & metadata*	Byte stream resolvable (`0.DOIP/Retrieve`), `includeElementData` option can retrieve bytestream or full object structure. No method/attribute defined for separate metadata, only directly in PID Record. Unclear meaning of multiple items and bytestream chunks.	Clarify expectations for multiple items. Recommend chunks to not be used.	URIs resolvable by default. Multiple ways to resolve metadata, unclear preference.	Add FAIR Signposting and preference order.
FDOF4: *Additional attributes*	Freetext attribute keys. Attributes should be defined for FDO type.	Require that attribute keys should be PIDs (or have predefined mapping to PIDs). Explicitly allow attributes not already defined in type.	All attributes individually identified. Any Linked Data attributes can be used by URI or with mapped prefix.	Clarify type expectations of required/recommended/optional attributes.
FDOF5: *Interface: operation by PID*	Extended operations use PID, but “pid-like” DOIP operations/types are not registered as handles.	Register `0.DOIP/*` and `0.FDO/*` as PIDs. Clarify that operations can be mapped to protocol directly.	CRUD operations used directly in HTTP (e.g. `PUT`). Unclear how to provide PID for additional operations.	Specify how additional operations should be called over HTTP.

FDO Guideline	DOIP 2.0	FDO suggestions	Linked Data Platform	LDP suggestion
FDOF6: *CRUD operations + extensions*	`0.DOIP/Op.Create`, `Op.Retrieve`, `Op.Update`, `Op.Delete` but also `0.DOIP/Op.Search`.	Document	`PUT`, `GET`, `POST`, `DELETE`, `PATCH`, `HEAD` – extension operations (e.g. WebDAV `COPY`) not used, resource patterns (Ekuan et al. 2023) are used instead.	Document how operation resources can be discovered from an LPD container. Document search API.
FDOF7: *FDOF Types related to operations*	Not yet formalised, by DOIP discoverable on a given FDO rather than type. PR-TypingFDOs leaves this open.	Add explicit relation between type and operations	`OPTIONS` per LDP Resource, but not by type. Common types (`ldp:Resource`, `ldp:Container`) indicate LDP support, but are not required.	Always make LDP types explicit in FDO profile.
FDOF8: *Metadata as FDO, semantic assertions*	DOIP includes all metadata in PID Record. Separate Metadata FDO need custom property.	Specify a `0.FDO/metadata` or similar to point to Metadata FDOs.	Assertions are always with semantics, using RDF vocabularies. Unspecified how to find additional metadata resources, `rdfs:seeAlso` is common.	Use FAIR Signposting `describedby` link relation to additional metadata PIDs
FDOF9: *Different metadata levels*	Defines open-ended "Response Attributes" without namespaces, but mandated as "None" for all CRUD operations. Metadata would need to be bundled within custom FDO types or attributes. Unclear how levels are separated within single FDO representation (may need FDOF8).	Declare which metadata are expected within response attribute or within FDO object. Require PIDs for custom attributes. Define how alternate metadata levels can be represented separately.	Undefined how to handle multiple metadata granularities or domains, alternative LDP containers can present different views on same stored objects.	Define how to navigate to alternate views and their semantic implications, e.g. `owl:sameAs`
FDOF10: *Metadata schemas by community*	Metadata schemas are in practice managed on single Cordra server as local types, using JSON Schema.	Require types to be FDOs with registered PIDs, implement shared types.	Plethora of existing RDF vocabularies/ontologies managed by larger communities, e.g. OBO Foundry (Smith et al. 2007)	Rather document better how individual ad-hoc schemas can be started for prototypes.
FDOF11: *FDO collections w/ semantic relations*	Collection type undefined by DOIP. Informal use of `HAS_PARTS` Handle attribute (e.g. (Semmler et al. 2022)).		LDP Containers required by specification, also user-created (eg. `BasicContainer`).	Clarify relation to other collections like DCAT 3 (Dataset Exchange Working Group 2023), Schema.org Dataset, OAI-ORE (Lagoze et al. 2008)
FDOF12: *Deleted FDO preserve PID w/ tombstone*	Tombstone for deleted resource undefined by DOIP. `0.DOIP/Status.104` status code does not distinguish "Not Found" or "Gone"	Formalise tombstone requirements with new FDO type	`410 Gone` recommended, but `404 Not Found` common. No requirement for tombstone serialisation	Formalise tombstone requirements and serialisation

Comparing FDO and Web as middleware infrastructures

In this section, we take the perspective that FDO principles are in effect proposing a global infrastructure of machine-actionable digital objects. As such we can consider implementations of FDO as **middleware infrastructures** for programmatic usage, and can evaluate them based on expectations for client and server developers.

We argue that the Web, with its now ubiquitous use of REST API (Roy Thomas Fielding 2000), can be compared as a similar global middleware. Note that while early moves for developing Semantic Web Services (Fensel et al. 2011) attempted to merge the Web Service and RDF aspects, we are here considering mainly the current programmatic Web and its mostly light-weight use of 3 out of possible *5 stars Linked Data* (Michael Hausenblas et al. 2012).

For this purpose, we here utillise the Comparison Framework for Middleware Infrastructures (Zarras 2004) that formalise multiple dimensions of openness, scalability, transparency, as well as characteristics known from Object-oriented programming such as modularity, encapsulation and inheritance.

Based on the analysis in Table 4 on the next page, we make the following observations:

- With respect to the aspect of *Performance*, it is interesting to note that while the first version of DOIP (Reilly 2009) supported multiplexed channels similar to HTTP/2 (allowing concurrent transfer of several digital objects). Multiplexing was removed for the much simplified DOIP 2.0 (*DOIPV2.0* 2018). Unlike DOIP 1.0, DOIP 2.0 will require a DO response to be sent back completely, as a series of segments (which again can be split the bytes of each binary *element* into sized *chunks*), before transmission of another DO response can start on the transport channel. It is unclear what is the purpose of splitting a binary into chunks on a channel which no longer can be multiplexed and the only property of a chunk is its size[10].
- HTTP has strong support for scalability and caching, but this mostly assumes read-operations from static resources. FDO has no view on immutability or validity of retrieved objects, but this should be taken into consideration to support large-scale usage.
- HTTP optimisations for performance (e.g. HTTP/2, multiplexing) is largely used for commercial media distribution (e.g. Netflix), and not commonly used by providers of FAIR data
- Cloud deployment of Web applications give many middleware benefits (Scalability, Distribution, Access transparancy, Location transparancy) – it is unclear how DOIP as a custom protocol would perform in a cloud setting as most of this infrastructure assumes HTTP as the protocol.
- Programmatically the Web is rather unstructured as middleware, as there are many implementation choices. Usually it is undeclared what to expect for a given URI/service, and programmers follow documented examples for a particular service rather than automated programmatic exploration across providers. This mean one can consider the Web as an ecosystem of smaller middlewares with commonalities.
- Many providers of FAIR Linked Data also provide programmatic REST API endpoints, e.g. UNIPROT, ChEMBL, but keeping the FAIR aspects such as retrieving metadata in such a scenario may require combining different services using multiple formats and identifier conventions.

[10] Although it is possible with `0.DOIP/Op.Retrieve` to request only particular individual elements of an DO (e.g. one file), unlike HTTP's `Range` request, it is not possible to select individual chunks of an element's bytestream.

Table 4. Comparing FAIR Digital Object (with the DOIP 2.0 protocol *DOIPV2.0* 2018) and Web technologies (using Linked Data) as middleware infrastructures Zarras 2004

Quality	FDO w/ DOIP	Web w/ Linked Data
Openness: *framework enable extension of applications*	FDOs can be cross-linked using PIDs, pointing to multiple FDO endpoints. Custom DOIP operations can be exposed, although it is unclear if these can be outside the FDO server. PID minting requires Handle.net prefix subscription, or use of services like Datacite, B2Handle.	The Web is inherently open and made by cross-linked URLs. Participation requires DNS domain purchase (many free alternatives also exists). PID minting can be free using PURL/ARK services, or can use DOI/Handle with HTTP redirects.
Scalability: *application should be effective at many different scales*	No defined methods for caching or mirroring, although this could be handled by backend, depending on exposed FDO operations (e.g. Cordra can scale to multiple backend nodes)	Cache control headers reduce repeated transfer and assist explicit and transparent proxies for speed-up. HTTP `GET` can be scaled to world-population-wide with Content-Delivery Networks (CDNs), while write-access scalability is typically manage by backend.
Performance: *efficient and predictable execution*	DOIP has been shown moderately scalable to 100 millions of objects, create operation at 900 requests/second . DOIP protocol is reusable for many operations, multiple requests may be answered out of order (by `requestId`). Multiple connections possible. Setup is typically through TCP and TLS which adds latency.	HTTP traffic is about 10% of global Internet traffic, excluding video and social networks (Sandvine 2022). HTTP 1 connections are serial and reusable, and concurrent connections is common. HTTP/2 adds asynchronous responses and multiplexed streams (Belshe et al. 2015) but still has TCP+TLS startup costs. For reduced latency, HTTP/3 (Bishop 2022) use QUIC (Iyengar and Thomson 2021) rather than TCP, already adapted heavily (30% of EMEA traffic) of which Instagram & Facebook video is the majority of traffic (Joras and Chi 2020).
Distribution transparency: *application perceived as a consistent whole rather than independent elements.*	Each FDO is accessed separately along with its components (typically from the same endpoint). FDOs should provide the mandatory kernel metadata fields. FDOs of the same declared type typically share additional attributes (although that schema may not be declared). DOIP does not enforce metadata typing constraints, this need to be established as FDO conventions.	Each URL accessed separately. Common HTTP headers provide basic metadata, although it is often not reliable. A multitude of schemas and serializations for metadata exists, conventions might be implied by a declared profile or certain media types. Metadata is not always machine findable, may need pre-agreed API URI Templates (Gregorio, Roy T. Fielding, et al. 2012), content-negotiation (MDN 2023) or FAIR Signposting (Van de Sompel, Klein, et al. 2022).
Access transparency: *local/remote elements accessed similarly*	FDOs should be accessed through PID indirection, this means difficult to make private test setup. Commonly a fixed DOIP server is used directly, which permits local non-PID identifiers.	Global HTTP protocol frequently used locally and behind firewalls, but at risk of non-global URIs (e.g. `http://localhost/object/1`) and SSL issues (e.g. self-signed certificates, local CAs)
Location transparency: *elements accessed without knowledge of physical location*	FDOs always accessed through PIDs. Multiple locations possible in Handle system, can expose geo-info.	PIDs and URL redirects. DNS aliases and IP routing can hide location. Geo-localised servers common for large cloud deployments.

Quality	FDO w/ DOIP	Web w/ Linked Data
Concurrency transparency: *concurrent processing without interference*	No explicit concurrency measures. FDO kernel metadata can include checksum and date.	HTTP operations are classified as being stateless/idempotent or not (e.g. `PUT` changes state, but can be repeated on failure), although these constraints are occassionally violated by Web applications. Cache control, `ETag` (e.g. checksum) and modification date in HTTP headers allows detection of concurrent changes on a single resource.
Failure transparency: *service provisioning resilient to failures*	DOIP status codes, e.g. `0.DOIP/Status.104`, additional codes can be added as custom attributes	HTTP status codes e.g. `404 Not Found`, specific meaning of standard codes can be documented in Open API. Custom codes uncommon.
Migration transparency: *allow relocating elements without interfering application*	Update of PID record URLs, indirection through `0.TYPE/DOIPServiceInfo` (not always used consistently). No redirection from DOIP service.	HTTP `30x` status codes provide temporary or permanent redirections, commonly used for PURLs but also by endpoints.
Persistence transparency: *conceal deactivation/reactivation of elements from their users*	FDO requires use of PIDs for object persistence, including a tombstone response for deleted objects. There is no guarantee that an FDO is immutable or will even stay the same type (note: CORDRA extends DOIP with version tracking).	URLs are not required to persist, although encouraged (Berners-Lee 1998). Persistence requires convention to use PIDs/PURLs and HTTP `410 Gone`. An URL may change its content, change in type may sometimes force new URLs if exposing extensions like `.json`. Memento (Van de Sompel, Nelson, et al. 2013) expose versioned snapshots. WebDAV `VERSION-CONTROL` method (Clemm et al. 2002) (used by SVN).
Transaction transparency: *coordinate execution of atomic/isolated transactions*	No transaction capabilities declared by FDO or DOIP. Internal synchronisation possible in backend for Extended operations.	Limited transaction capabilities (e.g. `If-Unmodified-Since`) on same resource. WebDAV locking mechanisms (Dusseault 2007) with `LOCK` and `UNLOCK` methods.
Modularity: *application as collection of connected/distributed elements*	FDOs are inheritedly modular using global PID spaces and their cross-references. In practice, FDOs of a given type are exposed through a single server shared within a particular community/institution.	The Web is inheritently modular in that distributed objects are cross-referenced within a global URI space. In practice, an API's set of resources will be exposed through a single HTTP service, but modularity enables fine-grained scalability in backend.
Encapsulation: *separate interface from implementation. Specify interface as contract, multiple implementations possible*	Indirection by PID gives separation. FDO principles are protocol independent, although it may be unclear which protocol to use for which FDO (although `0.DOIP/Transport` can be specified after already contacting DOIP). Cordra supports native DOIP, DOIP over HTTP and Cordra REST API)	HTTP/1.1 semantics can seemlessly upgrade to HTTP/2 and HTTP/3. `http` vs `https` URIs exposes encryption detail[11]. Implementation details may leak into URIs (e.g. `search.aspx`), countered by deliberate design of URI patterns (Berners-Lee 1998)) and PIDs via Persistent URLs (PURL).

[11] The `http` protocol (port 80) can in theory also upgrade (Khare and Lawrence 2000) to TLS encryption, as commonly used by Internet Printing Protocol for `ipp` URIs, but on the Web, best practice is explicit

Quality	FDO w/ DOIP	Web w/ Linked Data
Inheritance: *Deriving specialised interface from another type*	DOIP types nested with parents, implying shared FDO structures (unclear if operations are inherited). FDO establishes need for multiple Data Type Registries (e.g. managed by a community for a particular domain). Semantics of type system currently undefined for FDO and DOIP, syntactic types can also piggyback of FDO type's schema (e.g. CORDRA `$ref` use of JSON Schema references (Wright et al. 2022))	Syntactically Media Type with multiple suffixes (Sporny and Guy 2023) (mainly used with `+json`), declaration of subtypes as profiles (RFC6906) *The 'profile' Link Relation Type*. In metadata, semantic type systems (RDFS (Guha and Brickley 2014)), OWL2 (W3C OWL Working Group 2012), SKOS (Isaac and Summers 2009)). OpenAPI 3 (D. Miller et al. 2021) inheritance and Polymorphism. XML `xsd:schemaLocation` or `xsd:type` (Thompson et al. 2012), JSON `$schema` (Wright et al. 2022)), JSON-LD `@context` (Sporny, Longley, et al. 2020). Large number of domain-specific and general ontologies define semantic types, but finding and selecting remains a challenge.
Signal interfaces: *asynchronous handling of messages*	DOIP 2.0 is synchronous, in FDO async operations undefined. Could be handled as custom jobs/futures FDOs	HTTP/2 multiplexed streams (Belshe et al. 2015), Web Sockets (Rice et al. 2022), Linked Data Notifications (Capadisli and Guy 2017), AtomPub (Gregorio and de hÓra 2007), SWORD (Jones and Jefferies 2021), Micropub (Parecki 2017), more typically ad-hoc jobs/futures REST resources
Operation interfaces: *defining operations possible on an instance, interface of request/response messages*	CRUD predefined in DOIP, custom operations through `0.DOIP/Op.ListOperations` (can be FDOs of type `0.TYPE/DOIPOperation`, more typically local identifiers like "`getProvenance`")	CRUD predefined in HTTP methods (Roy T. Fielding and Reschke 2014b), (extended by registration), URI Templates (Gregorio, Roy T. Fielding, et al. 2012), OpenAPI operations (D. Miller et al. 2021), HATEOAS[12] incl. Hydra (Lanthaler 2021), schema.org Actions (*Schema.Org Actions* 2022), JSON HAL (Kelly 2016) & Link headers (RFC8288) (Nottingham 2017)
Stream interfaces: *operations that can handle continuous information streams*	Undefined in FDO. DOIP can support multiple byte stream elements (need custom FDO type to determine stream semantics)	HTTP 1.1 (Roy T. Fielding and Reschke 2014a) chunked transfer, HLS (RFC8216) (Pantos and May 2017), MPEG-DASH (*ISO/IEC 23009-1* 2022)

`https` (port 443) URLs to ensure following links stay secure.

[12]HATEOAS: Hypermedia as the Engine of Application State (Roy Thomas Fielding 2000), an important element of the REST architectural style.

Assessing FDO against FAIR

In addition to having "FAIR" in its name, the FAIR Digital Object guidelines (PR-RequirementSpec-3.0) also include *G3: FDOs must offer compliance with the FAIR principles through measurable indicators of FAIRness*.

Here we evaluate to what extent the FDO guidelines and its implementation with DOIP and Linked Data Platform (Bonino da Silva Santos et al. 2022) comply with the FAIR principles (Mark D. Wilkinson et al. 2016). Here we've used the RDA's FAIR Data Maturity Model (FAIR Data Maturity Model Working Group 2020) as it has decomposed the FAIR principles to a structured list of FAIR indicators (Bahim et al. 2020), importantly considering *Data* and *Metadata* separately. In our interpretation for Table 5 on the following page we have for simplicity chosen to interpret "data" in FDOs as the associated bytestream of arbitrary formats, with remaining JSON or RDF structures always considered as metadata.

From this evaluation we observe:

- Linked Data in general is strong on metadata indicators, but LDP approach is weak as it has little concrete metadata guidance.
- FDO/DOIP are stronger on identifier indicators, while Linked Data approach for identifiers relies on best practices.
- Indicators on standard protocols (RDA-A1-04M, RDA-A1-04D, RDA-A1.1-01M, RDA-A1.1-01D) favour LDP's mature standards (HTTP, URI) – the DOIPv2 specification (*DOIPV2.0* 2018) has currently only a couple of implementations and is expressed informally. The underlying Handle system for PIDs is arguably mature and commonly used by researchers (this article alone references about 80 DOIs), however DOIs are more commonly accessed as HTTP redirects through resolvers like `https://doi.org/` and `http://hdl.handle.net/` rather than the Handle protocol.
- RDA-A1-02M and RDA-A1-02D highlights access by manual intervention, which is common for http/https URIs, but also using above PID resolvers for DOIP implementation CORDRA (e.g. `https://hdl.handle.net/21.14100/90ec1c7b-6f5e-4e12-9137-0cedd16d1bce`), yet neither LDP, FDO nor DOIP specifications recommends human-readable representations to be provided
- Neither DOIP nor LDP require license to be expressed (RDA-R1.1-01M, RDA-R1.1-02M, RDA-R1.1-03M), yet this is crucial for re-use and machine actionability of FAIR data and metadata to be legal
- Machine-understandable types, provenance and data/metadata standards (RDA-R1.1-03M RDA-R1.3-02M, RDA-R1.3-02M, RDA-R1.3-02D) are important for machine actionability, but are currently unspecified for FDOs. (WD-ImplAttributesTypesProfiles) explores possible machine-readable FDO types, however the type systems themselves have not yet been formalised. Linked Data on the other side have too many semantic and syntactic type systems, making it difficult to write consistent clients.
- Indicators for FAIR data are weak for either approach, as too much reliance is put on metadata. For instance in Linked Data, given a URL of a CSV file, what is its persistant identifier or license information? FAIR Signposting (Van de Sompel, Klein, et al. 2022) can improve findability of metadata using HTTP Link relations, which enable an FDO-like overlay for any HTTP resource. In DOIP, responses for bytestreams can include the data identifier: if that is a PID (not enforced by DOIP), its metadata is accessible.
- Resolving FDOs via Handle PIDs to the corresponding DOIP server is currently undefined by FDO and DOIP specifications. `0.TYPE/DOIPServiceInfo` lookup is only possible once DOIP server is known.

Table 5. Assessing RDA's FAIR Data Maturity Model FAIR Data Maturity Model Working Group 2020; Bahim et al. 2020 (first 2 columns) against the FDO guidelines Bonino et al. 2019, FDO implemented with the protocol DOIPv2 *DOIPV2.0* 2018, Linked Data Platform (LDP) Bonino da Silva Santos et al. 2022 and examples from Linked Data practices in general. (— indicates *Unspecified*, may be possible with additional conventions)

FAIR ID	Indicator	FDO guidelines	FDO/DOIP	FDO/LDP	Linked Data examples
RDA-F1-01M	Metadata is identified by a persistent identifier	FDOF4	Optional *Metadata FDO* w/separate PID	Content-negotiation to URL, not required to be PID	Metadata typically don't have own PID
RDA-F1-01D	Data is identified by a persistent identifier	FDOF1	PIDs required (FDOF1). Handle, DOI.	FDOF-IR (Identifier Record). PID can be any URI	"Cool" URIs (Berners-Lee 1998), PURL services incl. `purl.org`, `w3id.org`
RDA-F1-02M	Metadata is identified by a globally unique identifier	FDOR4 FDOF8	Optional *Metadata FDO*, unspecified how to indicate	Content-negotiation to URL	Not required, content-negotiation can redirect to URL or `Content-Location`. FAIR Signposting.
RDA-F1-02D	Data is identified by a globally unique identifier	FDOF1	All FDOs have PIDs (FDOR1), DOIP uses Handle system	FDOF-IR (Identifier Record)	Always accessed by URL
RDA-F2-01M	Rich metadata is provided to allow discovery	FDOF2 FDOF4 FDOF8 FDOF9	FDO has key-value metadata. Unclear how to link to additional metadata.	FDOF-IR links to multiple metadata records	RDF-based metadata by content negotiation or FAIR Signposting. Embedded in landing page (RDFa).
RDA-F3-01M	Metadata includes the identifier for the data	—	`id` and `type` are required metadata elements PIDs, also implicit as requests must use PID	PID only required in FDOF-IR record.	PID inclusion typical, but often inconsistent (e.g. `www.example.com` vs `example.com`) or missing (use of <> as *this* subject)
RDA-F4-01M	Metadata is offered in such a way that it can be harvested and indexed	FDOF10	No, registries not required (except Data Type Registries). Handle registry only searchable by PID.	—	Not specified, several registries/catalogues for vocabularies/types (e.g. (*NCBO BioPortal* 2022)). Indexing by search engines if exposing HTML w/schema.org.

FAIR ID	Indicator	FDO guidelines	FDO/DOIP	FDO/LDP	Linked Data examples
RDA-A1-01M	Metadata contains information to enable the user to get access to the data	FDOF3 FDOF6	Directly by DOIP, but not included in FDO metadata. `handle.net` HTTP resolution may redirect to landing page	Any property can point to URIs, but unclear if it is data	Common with clickable “follow your nose” URLs
RDA-A1-02M	Metadata can be accessed manually (i.e. with human intervention)	—	(Cordra HTML landing page from `handle.net` URIs)	Optional content-negotiation, e.g. by Apache Marmotta, OpenLink Virtuoso	HTTP content-negotiation to HTML is common
RDA-A1-02D	Data can be accessed manually (i.e. with human intervention)	—	(Cordra HTML landing page from `handle.net` URIs)	Optional content-negotiation	Direct download, HTML landing pages common for DOIs
RDA-A1-03M	Metadata identifier resolves to a metadata record	FDOF8+FDOF2	—	—	`Content-Location` or HTTP redirection may indicate metadata URI
RDA-A1-03D	Data identifier resolves to a digital object	FDOF2	Required, but frequently not directly resolvable	Recommended, but any URI acceptable	Resolvable HTTP/HTTPS URIs are most common, now infrequent URNs are not directly resolvable
RDA-A1-04M	Metadata is accessed through standardised protocol	G9 FDOF3	Retrievable from PID (FDOF3). Informal DOIP standard maintained by DONA Foundation	LDP standard maintained by W3C, HTTP standards maintained by IETF, FDO components resolved by informal proposals (custom vocabulary, extra HTTP methods) or HTTP content negotiation)	Formal HTTP standards maintained by IETF, HTTP content negotiation, informal FAIR Signposting
RDA-A1-04D	Data is accessible through standardised protocol	G9	(see above)	HTTP (Roy T. Fielding, Nottingham, et al. 2022)	HTTP/HTTPS, FTP (now less common), GridFTP (Allcock et al. 2005) (for large data), ARK (Kunze and Bermès 2022)
RDA-A1-05D	Data can be accessed automatically (i.e. by a computer program)	G4 FDOF3 FDOF6	Required, but few client libraries	HTTP `GET`, content-negotiation for `fdof/object`	Ubiquitous, hundreds of HTTP libraries

FAIR ID	Indicator	FDO guidelines	FDO/DOIP	FDO/LDP	Linked Data examples
RDA-A1.1-01M	Metadata is accessible through a free access protocol	G1 G8 G9	Partially realised: Handle system is open[13] protocol (Sun, Reilly, et al. 2003). One server implementation (CNRI 2022), free[14]. One DOIPv2 implementation (Cordra): free under BSD-like license (not recognised as Open Source).	LDP is open W3C recommendation (Speicher et al. 2015). Multiple LDP implementations.	DNS, HTTP, TLS, RDF standards are open, free and universal, large number of Open Source clients and servers.
RDA-A1.1-01D	Data is accessible through a free access protocol	G9	(see above)	URI, DNS, HTTP, TLS	URI, DNS, HTTP, TLS. Non-free DRM may be used (e.g. subscription video streaming)
RDA-A1.2-01D	Data is accessible through an access protocol that supports authentication and authorisation	(FDOR9)	TLS certificates, `authentication` field (details unspecified)	Implied	HTTP authentication, TLS certificates
RDA-A2-01M	Metadata is guaranteed to remain available after data is no longer available	FDOF12	—	Unspecified, however FDOF-IR links to separate metadata records	—
RDA-I1-01M	Metadata uses knowledge representation expressed in standardised format	FDOF8	Required, but not currently defined	—	Always implied by use of RDF syntaxes.
RDA-I1-01D	Data uses knowledge representation expressed in standardised format	—	—	—	Common (e.g. HDF5, JSON, XML), yet common scientific data formats frequently not standardised
RDA-I1-02M	Metadata uses machine-understandable knowledge representation	FDOF8	Required	Optional RDF metadata with any vocabulary	Always implied by use of RDF syntaxes.

[13]The `Handle.net` system was previously covered by software patent US6135646A which expired in 2013.

[14]The Handle.net public license is not OSI-approved (*OSI* 2022) as an open source license – it includes usage restrictions and requires Service Agreements. It is not a DOIP requirement to host a local Handle instance, e.g. EOSC provides the B2HANDLE service for acquiring Handle prefixes.

FAIR ID	Indicator	FDO guidelines	FDO/DOIP	FDO/LDP	Linked Data examples
RDA-I1-02D	Data uses machine-understandable knowledge representation	G4 G7 FDOR2	No requirements on binary data formats	Only indirectly, LDP Basic Container reference only information resources	Common, specially for scientific data formats
RDA-I2-01M	Metadata uses FAIR-compliant vocabularies	G3 FDOF10	Informally required	Unspecified, implied by use of RDF?	FAIR practices for LD vocabularies increasingly common, sometimes inconsistent (e.g. PURLs that don't resolve) or incomplete (e.g. unknown license)
RDA-I2-01D	Data uses FAIR-compliant vocabularies	—	—	—	Uncommon, except for some XML and RDF-embedding formats, e.g. Extensible Metadata Platform (XMP) (*ISO 16684-1* 2019)
RDA-I3-01M	Metadata includes references to other metadata	FDOR8	Implied (attributes to PIDs), currently unspecified if given attribute is value or reference	—	By definition (Linked Data reference existing URIs (*Linked Data* 2015)), `rdfs:seeAlso`, FAIR signposting (Van de Sompel, Klein, et al. 2022) `describedby`
RDA-I3-01D	Data includes references to other data	G6 FDOR3 FDOR11	—	—	URL hyperlinks common in several formats (HTML, PDF, JSON, XML).
RDA-I3-02M	Metadata includes references to other data	G6 FDOR3 FDOR8	Implied from custom FDO type's attribute	LDP Direct Container members can be any resources	URI objects are frequently data references, may be indirect via PID
RDA-I3-02D	Data includes qualified references to other data	FDOR3 FDOR11	Only indirectly through FDO metadata	Indirectly through LDP membership	Uncommon: Link relations, FAIR Signposting
RDA-I3-03M	Metadata includes qualified references to other metadata	(FDOR3)	Qualification by attribute keys defined per FDO Type	LDP Direct Container	Qualifications by property, PROV bundles (Lebo and Moreau 2013), schema.org/Role

FAIR ID	Indicator	FDO guidelines	FDO/DOIP	FDO/LDP	Linked Data examples
RDA-I3-04M	Metadata include qualified references to other data	(FDOR3)	Qualification by attribute keys defined per FDO type	LDP Indirect Container	Qualifications by property, n-ary indirection (schema.org Role (Holland and Johnson 2014), `prov:specializationOf` (Lebo, McGuinness, et al. 2013), OAI-ORE Proxy (Lagoze et al. 2008))
RDA-R1-01M	Plurality of accurate and relevant attributes are provided to allow reuse	FDOF4	Required. Kernel metadata attributes desired (PR-KernelAtributues-2.0) but not assigned PIDs yet.	Unspecified. Multiple metadata records can allow multiple semantic profiles.	Large number of general and domain-specific vocabularies can make it hard to find relevant attributes. Rough consensus on kernel metadata: schema.org (*Schema.Org - Schema.Org* 2022), Dublin Core Terms (DCMI Usage Board 2020), DCAT (Browning et al. 2020), FOAF (Brickley and L. Miller 2014)
RDA-R1.1-01M	Metadata includes information about the licence under which the data can be reused	—	`licenseConditions` URL/PID in kernel metadata (PR-KernelAtributues-2.0)	—	Dublin Core Terms `dct:license` frequently recommended, frequently not required, e.g. by DCAT 2 (Browning et al. 2020)
RDA-R1.1-02M	Metadata refers to a standard reuse licence	—	—	—	SPDX and Creative Commons URIs common, identifiers often inconsistent
RDA-R1.1-03M	Metadata refers to a machine-understandable reuse licence	—	—	—	SPDX documents uncommon
RDA-R1.2-01M	Metadata includes provenance information according to community-specific standards	FDOR9 FDOR10	Unspecified (some Cordra types add getProvenance methods). PID Kernel attributes?	—	W3C PROV-O, PAV

FAIR ID	Indicator	FDO guidelines	FDO/DOIP	FDO/LDP	Linked Data examples
RDA-R1.2-02M	Metadata includes provenance information according to a cross-community language	FDOR9 FDOR8	—	—	W3C PROV-O (Lebo, McGuinness, et al. 2013), PAV (Ciccarese et al. 2013), Dublin Core Terms (DCMI Usage Board 2020)
RDA-R1.3-01M	Metadata complies with a community standard	FDOR10 FROR8	(Emerging, e.g. DiSSCo Digital Specimen (Hardisty et al. 2022))	—	Common, e.g. DCAT 2 (Browning et al. 2020), BioSchemas (Gray et al. 2017)
RDA-R1.3-01D	Data complies with a community standard	(FDOR3)	—	—	Common, HTTP use registered IANA media types, additional scientific file formats frequently not standardised or identified
RDA-R1.3-02M	Metadata is expressed in compliance with a machine-understandable community standard	FDOF4 FDOF10	Recommended	—	Common practice for ontologies, specially in bioinformatics, e.g. BioPortal (*NCBO BioPortal* 2022), Darwin Core (Wieczorek et al. 2012)
RDA-R1.3-02D	Data is expressed in compliance with a machine-understandable community standard	(FDOR2)	No, FDO is typed but data can be any bytestream	—	Occassionally, (e.g. GFF3, FITS, ESRI)

EOSC INTEROPERABILITY FRAMEWORK

The European Open Science Cloud (EOSC) is a large EU initiative to promote Open Science by implementing a joint research infrastructure by federating existing and new services and focusing on interoperability, accessability, best practices as well as technical infrastructure (Ayris et al. 2016). The EOSC Interoperability Framework (Corcho, Eriksson, et al. 2021) details the principles for creating a common way to achieve interoperability between all digital aspects of research activities in EOSC, including data, protocols and software. The recommendations are realized through 4 layers, Technical (e.g. protocols), Semantic (e.g. metadata models), Organisational (e.g. recommendations) and Legal (e.g. agreements), with a particular aim to address the FAIR interoperability principles and building on the concept of FAIR Digital Objects.

As covered in our introduction on page 1, EOSC proposes FAIR Digital Objects as a way to improve interoperability, for instance invoked by scientific workflows, carried by metadata frameworks and semantic artefacts. Therefore we here find it important to summarize how FDO and Linked Data can help satisfy the EOSC requirements.

In Table 6 on page 27 we review the EOSC Interoperability Framework (EOSC IF) recommendations, and evaluate to what extent they are addressed by the principles of FDO and Linked Data or their common implementations.

Firstly, we observe that the EOSC IF recommendations are at a high level, mainly affecting governance and practices by communities. This *Organizational* level is also highlighted by the FDO recommendations, for instance the FDO Typing (PR-TypingFDOs-2.0) propose a governance structure to recognize community-endorsed services. While these community aspects are not mandated by Linked Data practices, best practices have become established for aspects like ontology development (Norris et al. 2021). EOSC IF's *Technical* layer is likewise at a architecturally high level, such as service-level agreements, but also highlight PID policies which is strongly required by FDO, while Linked Data communities choose PID practices separately. The recommendations for the *Semantic* layer is largely already implemented by Linked Data practices, yet for FDO mostly consist of encouragements. For instance *clear definitions of semantic concepts* is required by FDO guidelines, but how to technically define them has not been formalised by FDO specifications.

The *Legal* layer of interoperability is perhaps the one most emphasised by EOSC, by enabling collaboration across organizational barriers to joinly build a research infrastructure, but this is an area that both FDO and Linked Data are relatively weak in directly supporting. The EOSC IF recommendations in this layer are largely related to governance practices and metadata, for instance licensing, privacy and usage policies; these are also essential for cross-institutional and cross-repository access of FAIR objects.

Likewise, search and indexing is important FAIR aspect for Findability, but is poorly supported globally by FDO and Linked Data. Efforts such as Open Research Knowledge Graph (ORKG) (Jaradeh et al. 2019), DataCite's PID Graph (Fenner and Aryani 2019) and Google Knowledge Graph (Singhal 2012) have improved programmatic findability to some degree, however not significantly for domain-specific semantic artefacts, currently scattered across multiple semantic catalogues (Corcho, Ekaputra, et al. 2023). There is a strong role for organizations like EOSC to provide such broader registries, moving beyond scholarly output metadata federations. The EOSC Marketplace[15] has for instance recently been expanded to include training material, software and data sources.

[15] https://marketplace.eosc-portal.eu/

Table 6. Assessing EOSC Interoperability Framework (Corcho, Eriksson, et al. 2021, section 3.6) against the FDO guidelines (Bonino et al. 2019) and Linked Data practices.

Layer	Recommendation	FDO	Linked Data
Technical	Open Specification	FDO specifications are semi-open, process gradually more transparent	Open and transparent standard processes through W3C & IETF
Technical	Common security & privacy framework	Unspecified	TLS for encryption, multiple approaches for single-sign-on (e.g. ORCID, Life Science Login). Privacy largely unspecified.
Technical	Easy SLAs for service providers	Unspecified	None
Technical	Access data in different formats	None formalised, custom operations or relations	Content-negotiation, `rel=alternate` relations
Technical	Coarse-grained/fine-grained search tools	Freetext `0.DOIP/Op.Search` on local DOIP, no federation	Coarse-grained e.g. Google Dataset Search, fine-grained (e.g. federated SPARQL) require detailed vocabulary/metadata insight
Technical	Clear PID policy	Strong FDO requirements, tends towards Handle system.	Not required, different communities set policies
Semantic	Clear definitions for concepts/metadata/schemas	Required by FDO requirements, but not yet formalised	Ontologies, SKOS, OWL
Semantic	Semantic artefacts w/ open licenses	All artefacts are PIDs, license not yet required by kernel metadata	Open License is best practice for ontology publishing
Semantic	Documentation for each semantic artefact	No direct rendering from FDO, no requirement for human-readable description	Ontology rendering, content-negotiation
Semantic	Repositories of artefacts	Required, but not formalised	Bioontologies, otherwise not usually federated
Semantic	Repositories w/ clear governance	Recommended	Largely self-governed repositories, if well-established may have clear governance.
Semantic	Minimal metadata model for federated discovery	Kernel metadata (PR-KernelAtributues-2.0) based on RDA recommendations (Weigel et al. 2018).	DCAT, schema.org, Dublin Core
Semantic	Crosswalks from minimal metadata model	FDO Typing recommends referencing existing type definitions, but not as separate crosswalks	Multiple crosswalks for common metadata models, but frequently not in semantic format
Semantic	Extensibility options for diciplinary metadata	Communities encouraged to establish own types	Extensible by design, domain-specific metadata may be at different granularity
Semantic	Clear protocols/building blocks for federation/harvesting of artefact catalogues	Collection types not yet defined	SWORD, OAI-PMH

Table 6. Assessing EOSC Interoperability Framework (Corcho, Eriksson, et al. 2021, section 3.6) against the FDO guidelines (Bonino et al. 2019) and Linked Data practices.

Layer	Recommendation	FDO	Linked Data
Organisational	Interoperability-focused rules of participation recommendations	Recommended	Implied only by some communities, tendency to specialise
Organisational	Usage recommendations of standardised data formats	None	None – but common for metadata (e.g. JSON-LD)
Organisational	Usage recommendations of vocabularies	Recommended by community	Common (see RDMKit)
Organisational	Usage recommendations of metadata	Recommended by community	RO-Crate, Bioschemas
Organisational	Management of permanent organization names/functions	Handle owner, but unclear contact. Contact info in DOIP service provider	ROR. DCAT contacts.
Legal	Standardised human and machine-readable licenses	None	SPDX, but not that frequently used
Legal	Permissive licenses for metadata (CC0, CC-BY-4.0)	Undefined	Both CC0, CC-BY-4.0 common, e.g. in DCAT
Legal	Different licenses for different parts	Each part as separate FDO can have separate license	DCAT, RO-Crate, Named graphs for splitting metadata
Legal	Mark expired/inexistent copyright	Undefined	Unclear, semantics assume copyright valid
Legal	Mark orphaned data	Tombstone for deleted data, but no owner of DOIP server means FDO disappears	Frequently data and endpoint has no known maintainer, archiving in common repositories becoming common
Legal	List recommended licenses	Undefined	Best practice recommendations
Legal	Track license evolution for dataset	Undefined	Versioning with PAV/PROV/DCAT
Legal	Policy/guidance for patent/trade secrets violation	Undefined	Undefined, legal owner may be specified. ODRL can express policies.
Legal	GDPR compliance for personal data	Undefined	Undefined
Legal	Restrict access/use if legally required	By transport protocol (undefined by FDO/DOIP)	Diverging approaches, typically landing pages w/ auth&auth or click-thru
Legal	Harmonised terms-of-use	Undefined	Undefined
Legal	Alignment between EOSC and national legislation	Not applicable	Not applicable

DISCUSSION

We have evaluated the FAIR Digital Object concept using multiple frameworks, and contrasted FDO against existing experiences from Linked Data on the Web. In this section we discuss the implications of this evaluation, and propose how these two approaches can be better combined.

Framework evaluation

Having considered FDO and the Web architecture as interoperability frameworks (on page 7), we observe that neither are magic bullets, but each bring different aspects of interoperability. The Web comes with a large degree of flexibility and openness, however this means interoperability can suffer as services have different APIs and data models, although with common patterns. This is also true for Linked Data on the Web, with many overlapping ontologies and frequent inconsistencies in resolution mechanisms; although somewhat alleviated in recent years by schema.org becoming common metadata model for semantic markup inline in Web pages. The Web is based on a common HTTP protocol which has remained stable architecturally throughout its 32 years of largely backwards-compatible evolution. FDO on the other side sets down multiple rigid rules for identifiers, types, methods etc. that are advanterous for interoperability and predictability for FAIR consumption. Yet there is a large degree of freedom in how the FDO rules can be implemented by a given community, for instance there is no common metadata model or identifier resolution mechanism, and DOIP is just one possible transport method for FDOs, which itself does not enforce these rules.

When evaluating FDO implementations against the FDO guidelines (on page 10) we see that several technical pieces and community practices still need to be developed and further defined, for instance the FDO type system, how to declare FDO actions, how to resolve persistent identifiers, or how to know which pattern of FDO composition is used. Achieving fully interoperable FAIR Digital Objects would require further convergence on implementation practices, and it is not given that this needs to diverge from the established Web architecture. It is not clear from FDO guidelines if moving from HTTP/DNS to DOIP/Handle as a way to expose distributed digital objects will benefit FAIR practitioners, when both approaches require additional equably implementable restrictions and conventions, such as using persistent identifiers or pre-defining an object's type.

Considering this, by comparing FDO and Web as middleware (on page 14) we saw that programmatic access to digital objects, a core promise of FDO, is not particularly improved by the use of the protocol DOIP as compared to HTTP, e.g. lack of concurrency and transparancy. Recent updates to HTTP have added many features needed for large-scale usage such as video streaming services (e.g. caching, multiplexing, cloud deployments), and having the option to transparently apply these also to FDOs seems like a strong incentive. Many programmatic features for distributed objects are however missing or needing custom extensions in both aspects, such as transactions, asynchronous operations and streaming.

By assessing FDO against the FAIR principles (on page 18) we found that both FDO implementations are underspecified ian several aspects (licences, provenance, data references, data vocabularies, metadata persistence). While there are implementations of each of these in general Linked Data examples, there is no single set of implementation guides that fully realizes the FAIR principles. *FAIRification* efforts like the FAIR Cookbook (Rocca-Serra et al. 2023) and FAIR Implementation Profiles (Schultes, Magagna, et al. 2020) are bringing existing practices together, but there remains a potential role for FDO in giving a coherent set of implementation practices that can practically achieve FAIR. Significant effort, also within EOSC, is now moving towards FAIR metrics (Devaraju et al. 2021), which in practice need to make additional assumptions on how FAIR principles are implemented, but these are not always formalized (Mark D Wilkinson et al. 2022) nor can they be taken to be universally correct (Verburg et al. 2023). Given that most of the existing FAIR guides and assessment tools are focused on Web and Linked Data, it would be reasonable for FDO to then provide a profile of such implementation choices that can achieve best of both worlds.

EOSC has been largely supportive of FDO, FAIR and related services. By contrasting the EOSC Interoperability Framework (on page 25) with FDO, we found that there are important dimensions that are not solved at a technical level, but through organization collaboration, legal requirements and building community practices. FDO recommendations highlight community aspects, but at the same time the largest FAIR communities in many science domains are already producing and consuming Linked Data. Just as the Linked Data community has a challenge in convincing more research fields to use Semantic Web technologies, FDO currently need to build many new communities in areas that have shown interest in that approach (e.g. material science). It may be advantageous for both these effort to be aligned and jointly promoted under the EOSC umbrella.

What does FDO mean for Linked Data?

The FAIR Digital Object approach raises many important points for Linked Data practictioners. At first glance, the explicit requirements of FDOs may seem to be easy to furfill by different parts of the Semantic Web Cake (Berners-Lee 2000, slide 10), as has previously been proposed (Soiland-Reyes, Castro, et al.

2022). However, this deeper investigation, based on multiple frameworks, highlights that the openness and variability of how Linked Data is deployed can make it difficult to achieve the FDO goals without significant effort.

While RDF and Linked Data have been suggested as prime candidates for making FAIR data, we argue that when different developers have too many degrees of freedom (such as serialization formats, vocabularies, identifiers, navigation), interoperability is hampered – this makes it hard for machines to reliably consume multiple FAIR resources across repositories and data providers. Indeed, this may be one reason why the initial FDO effort steered away from Linked Data approaches, but now seems in a danger of opening the many same degrees of freedom within FDO.

We therefore identify the need for a new explicit FDO profile of Linked Data that sets pragmatic constraints and stronger recommendations for consistent and developer-friendly deployment of digital objects. Such a combination of efforts could utillise both the benefits of mature Semantic Web technologies (e.g. federated knowledge graph queries and rich validation) and data management practices that follow FDO guidance in order to grow an ecosystem of machine-actionable objects. It is beyond the scope of this work to detail such a profile, but we suggest the following potential key aspects:

- Use HTTP(S) as protocol
- Use URIs as identifiers, with persistent identifier promises
- Provide consistent identifier resolution that does not require heuristics
- Common core metadata model
- References are always URIs, and should be persistent identifiers
- Types, attributes and actions are self-defined by their identifier
- Use Web approaches directly where possible, rather than wrap in a new model

The FAIR and Linked Data communities likewise need to recognize the need for simpler, more pragmatic approaches that make it easier for FAIR practitioners to adapt the technologies with ”just enough” semantics.

CONCLUSION

In this work, we have considered FAIR Digital Objects (FDO) as a potential distributed object system for FAIR data and compared it with established Web approaches focusing on Linked Data. We have described the background of the Semantic Web and FAIR Digital Objects, and evaluated both using multiple conceptual frameworks.

We find that both FDO and Linked Data approaches can significantly benefit from each-other and should be aligned further. Namely, Linked Data proponents need to make their technologies more approachable, agreeing on predictable and consistent implementations of FAIR principles.

The FDO recommendations show that FAIR thinking in this regard need to move beyond data publishing and into machine actionability across digital objects, and with broader community consensus. As flexibility for extensions is a necessary ingredient alongside rigidity for core concepts, the FDO community likewise need to settle on directly implementable specifications rather than just guidelines, and avoid making similar mistakes as learnt by early Semantic Web adopters.

By implementing the goals of FAIR Digital Objects with the mature technology stack developed for Linked Data, EOSC research infrastructures and researchers in general can create and use FAIR machine-actionable research outputs for decades to come.

ACKNOWLEDGMENTS

This work was funded by the European Union programmes *Horizon 2020* under grant agreements H2020-INFRAEDI-02-2018 823830 (BioExcel-2), H2020-INFRAEOSC-2018-2 824087 (EOSC-Life) and *Horizon Europe* under grant agreements HORIZON-INFRA-2021-EMERGENCY-01 101046203 (BY-COVID), HORIZON-INFRA-2021-EOSC-01 101057388 (EuroScienceGateway), HORIZON-INFRA-2021-EOSC-01-05 101057344 (FAIR-IMPACT), HORIZON-INFRA-2021-TECH-01 101057437 (BioDT); HORIZON-CL4-2021-HUMAN-01-01 101070305 (ENEXA) and by UK Research and Innovation (UKRI) under the UK government's *Horizon Europe funding guarantee* grants 10038963 (EuroScienceGateway), 10038992(FAIR-IMPACT), 10038930 (BioDT).

We would like to acknowledge the FAIR Digital Object Forum (*FDO* 2022) community and working groups, where SSR and CG are members.

AUTHOR CONTRIBUTIONS

Contributions to this article according to the CASRAI CRediT taxonomy[16]:

Stian Soiland-Reyes Conceptualization, Formal Analysis, Funding acquisition, Investigation, Methodology, Writing – original draft, Writing – review & editing
Carole Goble Funding acquisition, Supervision, Writing – review & editing
Paul Groth Conceptualization, Methodology, Supervision, Writing – review & editing

REFERENCES

Allcock, W., J. Bresnahan, R. Kettimuthu, and M. Link (2005-12-19). "The Globus Striped GridFTP Framework and Server". In: *SC '05: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing*. ACM/IEEE SC 2005 Conference (SC'05) (2005-11-12). Seattle, WA, USA: IEEE. https://doi.org/10.1109/sc.2

Anders, Ivonne, Christophe Blanchi, Daan Broder, Maggie Hellström, Sharif Islam, Thomas Jejkal, Larry Lannom, Karsten Peters-von Gehlen, et al. (2023-01-12). *FDO Forum FDO Requirement Specifications*. Version 3.0. Proposed Recommendation PR-RequirementSpec-3.0. https://doi.org/10.5281/zenodo.7782262.

Anders, Ivonne, Christophe Blanchi, Daan Broder, Maggie Hellström, Sharif Islam, Thomas Jejkal, Larry Lannom, Karsten Peters-von Gehlen, et al. (2023-01-18). *FAIR Digital Object Technical Overview*. Version PEN 2.0. Proposed Recommendation Full FDO Overview PEN-2.0-v2. https://doi.org/10.5281/zenodo.782471

Anders, Ivonne, Maggie Hellström, Sharif Islam, Thomas Jejkal, Larry Lannom, Ulrich Schwardmann, and Peter Wittenburg (2022-10-17). *FDO PID Profiles & Attributes*. Proposed Recommendation PR-PIDProfileAttributes-2.1-20221017. https://doi.org/10.5281/zenodo.7825630.

ANSI Z39.99 (2017-02-02). *ANSI/NISO Z39.99-2017, ResourceSync Framework Specification*. National Information Standards Organization ResourceSync Standing Committee. https://doi.org/10.3789/ansi.niso.z39.99-2017. eprint: http://www.openarchives.org/rs/1.1/resourcesync.

Ayris, Paul et al. (2016). *Realising the European Open Science Cloud. First report and recommendations of the Commission High Level Expert Group of the European Open Science Cloud*. https://doi.org/10.2777/940154.

Bahim, Christophe et al. (2020-10-27). "The FAIR Data Maturity Model: An Approach to Harmonise FAIR Assessments". In: *Data Science Journal* 19 (1). ISSN: 1683-1470. https://doi.org/10.5334/dsj-2020-041.

Baker, Thomas and Eric Prud'hommeaux (2019-10). *Shape Expressions (ShEx) 2.1 Primer*. URL: http://shex.io/shex-prime (visited on 2022-05-26).

Belshe, Mike, Roberto Peon, and Martin Thomson (2015-05). *Hypertext Transfer Protocol Version 2 (HTTP/2)*. Request for Comments. RFC 7540. RFC Editor. https://doi.org/10.17487/rfc7540.

Berners-Lee, Tim (1998). *Cool URIs don't change*. URL: https://www.w3.org/Provider/Style/URI (visited on 2023-02-02).

— (2000-12-06). *Semantic Web on XML*. URL: https://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html (visited on 2023-01-24).

— (2006-07-27). *Linked Data - Design Issues*. W3C Design Issues. URL: https://www.w3.org/DesignIssues/LinkedData.htm (visited on 2022-05-26).

Berners-Lee, Tim, Roy T. Fielding, and Larry M Masinter (2005-01). *Uniform Resource Identifier (URI): Generic Syntax*. Request for Comments. RFC 3986. RFC Editor. https://doi.org/10.17487/rfc3986.

Berners-Lee, Tim and Mark Fischetti (1999). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. 1st ed. HarperSanFrancisco. 226 pp. ISBN: 978-0-06-251586-5.

Bernstein, Abraham, James Hendler, and Natalya Noy (2016-08-24). "A New Look at the Semantic Web". In: *Communications of the ACM* 59.9, pp. 35–37. ISSN: 00010782. https://doi.org/10.1145/2890489.

Bishop, Mike (2022-06). *HTTP/3*. Request for Comments. RFC 9114. RFC Editor. https://doi.org/10.17487/rfc9114.

Bizer, Christian, Tom Heath, and Tim Berners-Lee (2009-07). "Linked Data - the Story so Far". In: *International journal on Semantic Web and information systems* 5.3, pp. 1–22. ISSN: 1552-6283. https://doi.org/10.4018/jswis.2009081901.

Blanchi, Christophe, Daan Broeder, Thomas Jejkal, Islam Sharif, Alexander Schlemmer, Dieter van Uytvanck, and Peter Wittenburg (2022-10-17). *FDO – Upload of FDO*. Proposed Endorsement Note PEN-FDO-Upload-1.1-20221017. https://doi.org/10.5281/zenodo.7825549.

[16] https://credit.niso.org/

Blanchi, Christophe, Maggie Hellström, Larry Lannom, Andreas Pfeil, Ulrich Schwardmann, and Peter Wittenburg (2023-03-14). *Implementation of Attributes, Types, Profiles and Registries*. Working Draft WD-Implementation-of-Attributes-0.4-20230314. https://doi.org/10.5281/zenodo.7825572.

Bonino, Luiz, Oeter Wittenburg, Bonnie Carroll, Alex Hardisty, Mark Leggott, and Carlo Zwölf (2019-11-22). *FAIR Digital Object Framework*. FDOF Technical Implementation Guideline. URL: https://github.com/GEDE-

Bonino da Silva Santos, Luiz Olavo, Giancarlo Guizzardi, and Tiago Prince Sales (2022-10-27). *FAIR Digital Object Framework Documentation*. Ed. by Luiz Olavo Bonino da Silva Santos. URL: https://fairdigitalobjectfra (visited on 2022-05-26).

Bonino Da Silva Santos, Luiz Olavo, Mark D. Wilkinson, Arnold Kuzniar, Rajaram Kaliyaperumal, Mark Thompson, Michel Dumontier, and Kees Burger (2016). "FAIR Data Points Supporting Big Data Interoperability". In: *Enterprise Interoperability in the Digitized and Networked Factory of the Future*. Ed. by Martin Zelm, Guy Doumeingts, and Joao Pedro Mendonça. iSTE Press, pp. 270–279. ISBN: 978-1-84704-044-2. URL: https://www.researchgate.net/publication/309468587_FAIR_Data_Points_Supporting_Big_Data_Int (visited on 2022-11-30).

Brickley, Dan and Libby Miller (2014-01-14). *FOAF Vocabulary Specification*. URL: http://xmlns.com/foaf/spec/ (visited on 2022-05-26).

Broeder, Daan and Peter Wittenburg (2022-11-19). *FDO Glossary November 2022*. Spreadsheet FDO Glossary Nov 2022. HDL: 20.500.14132/fdo-spec-docs. (Visited on 2023-02-02).

Broeder, Daan, Peter Wittenburg, Ivonne Anders, and Karsten Peters-von Gehlen (2022-10-17). *FDO – Kernel Attributes & Metadata*. Proposed Recommendation PR-FDO-KernelAttributesAndMetadata-2.0-20221017. https://doi.org/10.5281/zenodo.7825693.

Browning, David, Peter Winstanley, Andrea Perego, Simon Cox, Riccardo Albertoni, and Alejandra Gonzalez Beltran (2020-02-04). *Data Catalog Vocabulary (DCAT) - Version 2*. W3C Recommendation. URL: https://www.w3.org/TR/2020/REC-vocab-dcat-2-20200204/.

Capadisli, Sarven and Amy Guy, eds. (2017-05-02). *Linked Data Notifications*. W3C Recommendation. W3C Social Web Working Group. URL: https://www.w3.org/TR/2017/REC-ldn-20170502/.

Carriero, Valentina Anita, Marilena Daquino, Aldo Gangemi, Andrea Giovanni Nuzzolese, Silvio Peroni, Valentina Presutti, and Francesca Tomasi (2020-11-12). "The Landscape of Ontology Reuse Approaches". In: *Applications and Practices in Ontology Design, Extraction, and Reasoning*. Ed. by Giuseppe Cota, Marilena Daquino, and Gian Luca Pozzato. Studies on the Semantic Web. IOS Press. ISBN: 978-1-64368-142-9. https://doi.org/10.3233/ssw200033.

Ciccarese, Paolo, Stian Soiland-Reyes, Khalid Belhajjame, Alasdair JG Gray, Carole Goble, and Tim Clark (2013). "PAV Ontology: Provenance, Authoring and Versioning". In: *Journal of Biomedical Semantics* 4.1, p. 37. https://doi.org/10.1186/2041-1480-4-37.

Clemm, Geoffrey M., Jim Amsden, Tim Ellison, Christopher Kaler, and Jim Whitehead (2002-03). *Versioning Extensions to WebDAV (Web Distributed Authoring and Versioning)*. Request for Comments. RFC 3253. RFC Editor. https://doi.org/10.17487/rfc3253.

CNRI (2022). *Handle.Net Software*. Corporation for National Research Initiatives. URL: https://www.handle.net/download_ (visited on 2023-01-24).

— (2023a-04). "DOIP and Examples — Cordra Documentation". In: *Cordra® Software Technical Manual Version 2.5.0*. URL: https://www.cordra.org/documentation/api/doip.html (visited on 2023-06-13).

— (2023b-04). "DOIP API for HTTP Clients". In: *Cordra® Software Technical Manual Version 2.5.0*. URL: https://www.cordra.org/documentation/api/doip-api-for-http-clients.html (visited on 2023-06-13).

Corcho, Oscar, Fajar J. Ekaputra, Ivan Heibi, Clement Jonquet, Andras Micsik, Silvio Peroni, and Emanuele Storti (2023). *A maturity model for catalogues of semantic artefacts*. https://doi.org/10.48550/arXiv.2305.067 arXiv: 2305.06746 [cs.DL].

Corcho, Oscar, Magnus Eriksson, et al. (2021-02-05). *EOSC Interoperability Framework*. https://doi.org/10.2777/620649.

CWFR Group (2021-01-06). *Canonical Workflow Frameworks for Research. Position Paper*. Ed. by Alex Hardisty and Peter Wittenburg. Version 2. URL: https://osf.io/3rekv/.

Dataset Exchange Working Group (2023-03-07). *Data Catalog Vocabulary (DCAT) - Version 3*. Ed. by Riccardo Albertoni, David Browning, Simon Cox, Alejandra Gonzalez Beltran, Andrea Perego, and Peter Winstanley. W3C Working Draft. URL: https://www.w3.org/TR/2023/WD-vocab-dcat-3-20230307/.

DCMI Usage Board (2020-01-20). *DCMI Metadata Terms*. DCMI Recommendation. URL: https://www.dublincore.org/spe

Delgado, José Carlos Martins (2016). "An Interoperability Framework and Distributed Platform for Fast Data Applications". In: *Data Science and Big Data Computing*, pp. 3–39. https://doi.org/10.1007/978-3-319-31861-5_1.

Devaraju, Anusuriya et al. (2021). "From Conceptualization to Implementation: FAIR Assessment of Research Data Objects". In: *Data Science Journal* 20. https://doi.org/10.5334/dsj-2021-004.

DOI (2019-12-19). "DOI Handbook - Resolution". In: *DOI Handbook*. https://doi.org/10.1000/182. URL: https://www.doi.org/doi_handbook/3_Resolution.html (visited on 2023-01-24).

DOIPV2.0 (2018-11-12). *Digital Object Interface Protocol Specification, Version 2.0*. HDL: 0.DOIP/DOIPV2.0. URL: https://hdl.handle.net/0.DOIP/DOIPV2.0.

Dürst, Martin J. and Michel Suignard (2005-01). *Internationalized Resource Identifiers (IRIs)*. Request for Comments. RFC 3987. RFC Editor. https://doi.org/10.17487/rfc3987.

Dusseault, Lisa M. (2007-06). *HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)*. Request for Comments. RFC 4918. RFC Editor. https://doi.org/10.17487/rfc4918.

Ekuan, Martin, Mick Alberts, Tim Sherer, Udi Dahan, and Mike Kistler et al. (2023-03-28). "Web API Design Best Practices". In: *Azure Architecture Center*. Microsoft. URL: https://learn.microsoft.com/en-us/azure/architect (visited on 2023-06-14).

European Commission (2016-07). *Guidelines on FAIR Data Management in Horizon 2020. H2020 Programme*. Version 3.0. URL: https://ec.europa.eu/research/participants/data/ref/h2020/grants_manual/hi/oa_pilot/h2020-hi-oa-

FAIR Data Maturity Model Working Group (2020-06-25). "FAIR Data Maturity Model: Specification and Guidelines". In: ed. by Edit Herczog, Keith Russell, and Shelley Stall. https://doi.org/10.15497/rda00050.

FDO Specs (2022-11-19). *FDO Specification Documents - November 2022*. HDL: 20.500.14132/fdo-spec-docs. URL: https://fairdo.org/specifications/ (visited on 2023-02-02).

FDO (2022). *FAIR Digital Objects Forum*. URL: https://fairdo.org/ (visited on 2022-05-26).

Fenner, Martin and Amir Aryani (2019). "Introducing the PID Graph". In: https://doi.org/10.5438/jwvf-8a66.

Fensel, Dieter, Federico Michele Facca, Elena Simperl, and Ioan Toma (2011). *Semantic Web Services*. Springer Berlin Heidelberg. ISBN: 978-3-642-19193-0. https://doi.org/10.1007/978-3-642-19193-0.

Fielding, Roy T., Jim Gettys, Jeffrey Mogul, Henrik Frystyk Nielsen, Larry M Masinter, Paul J. Leach, and Tim Berners-Lee (1999-06). *Hypertext Transfer Protocol – HTTP/1.1*. Request for Comments. RFC 2616. RFC Editor. https://doi.org/10.17487/rfc2616.

Fielding, Roy T., Mark Nottingham, and Julian Reschke (2022-06). *HTTP Semantics*. Request for Comments. RFC 9110. RFC Editor. https://doi.org/10.17487/rfc9110.

Fielding, Roy T. and Julian Reschke (2014a-06). *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*. Request for Comments. RFC 7230. RFC Editor. https://doi.org/10.17487/rfc7230.

— (2014b-06). *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content*. Request for Comments. RFC 7231. RFC Editor. https://doi.org/10.17487/rfc7231.

Fielding, Roy T., Richard N. Taylor, Justin R. Erenkrantz, Michael M. Gorlick, Jim Whitehead, Rohit Khare, and Peyman Oreizy (2017-09-04). "Reflections on the REST Architectural Style and "Principled Design of the Modern Web Architecture" (Impact Paper Award)". In: *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2017*, pp. 4–14. ISBN: 978-1-4503-5105-8. https://doi.org/10.1145/3106237.3121282.

Fielding, Roy Thomas (2000). "Architectural Styles and the Design of Network-Based Software Architectures". Doctoral Thesis. URL: https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm (visited on 2022-06-28).

Gayo, Jose Emilio Labra, Eric Prud'hommeaux, Iovka Boneva, and Dimitris Kontokostas (2017-09-28). "Validating RDF Data". In: *Synthesis Lectures on the Semantic Web: Theory and Technology* 7.1, pp. 1–328. https://doi.org/10.2200/s00786ed1v01y201707wbe016.

Goble, Carole and Robert Stevens (2008-10-01). "State of the Nation in Data Integration for Bioinformatics." In: *Journal of Biomedical Informatics* 41.5, pp. 687–693. https://doi.org/10.1016/j.jbi.2008.01.008.

Gray, Alasdair, Carole Goble, Rafael Jimenez, and Bioschemas Community (2017-10-23). "Bioschemas: From Potato Salad to Protein Annotation". In: *Proceedings of the ISWC 2017 Posters & Demonstrations and Industry Tracks co-located with 16th International Semantic Web Conference (ISWC 2017)*. Ed. by Nadeschda Nikitina, Dezhao Song, Achille Fokoue, and Peter Haase. Vol. 1963. CEUR Workshop Proceedings. Vienna, Austria. URL: https://ceur-ws.org/Vol-1963/paper579.pdf.

Gregorio, Joe and Bill de hÓra (2007-10). *The Atom Publishing Protocol*. Request for Comments. RFCC 5023. RFC Editor. https://doi.org/10.17487/rfc5023.

Gregorio, Joe, Roy T. Fielding, Marc Hadley, Mark Nottingham, and David Orchard (2012-03). *URI Template*. Request for Comments. RFC 6570. RFC Editor. https://doi.org/10.17487/rfc6570.

Groth, Paul, Antonis Loizou, Alasdair J.G. Gray, Carole Goble, Lee Harland, and Steve Pettifer (2014-12). "API-centric Linked Data Integration: The Open PHACTS Discovery Platform Case Study". In: *Journal of Web Semantics* 29, pp. 12–18. https://doi.org/10.1016/j.websem.2014.03.003.

Guha, Ramanathan and Dan Brickley (2014-02). *RDF Schema 1.1*. W3C Recommendation. URL: http://www.w3.org/TR/2

Hardisty, Alex et al. (2022). "The Specimen Data Refinery: A Canonical Workflow Framework and FAIR Digital Object Approach to Speeding up Digital Mobilisation of Natural History Collections". In: *Data Intelligence* 4.2, pp. 320–341. https://doi.org/10.1162/dint_a_00134.

Hasnain, Ali and Dietrich Rebholz-Schuhmann (2018). "Assessing FAIR Data Principles against the 5-Star Open Data Principles". In: *The Semantic Web: ESWC 2018 Satellite Events: ESWC 2018 Satellite Events, Heraklion, Crete, Greece, June 3-7, 2018, Revised Selected Papers*. Ed. by Aldo Gangemi et al. Vol. 11155. Lecture Notes in Computer Science, pp. 469–477. ISBN: 978-3-319-98191-8. https://doi.org/10.1007/978-3-319-98192-5_60.

Hellström, Maggie, Carlo Zwölf, and Peter Wittenburg (2022-10-17). *FDO – Granularity, Versioning, Mutability*. Proposed Recommendation PR-Granularity-2.2-20221017. https://doi.org/10.5281/zenodo.7825686.

Holland, Vicki Tardif and Jason Johnson (2014-06-14). *Introducing 'Role'*. URL: http://blog.schema.org/2014/06/introducin (visited on 2023-01-24).

Horrocks, Ian and James Hendler, eds. (2002). *The Semantic Web — ISWC 2002*. Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-48005-6.

Hu, Wei, Jianfeng Chen, Hang Zhang, and Yuzhong Qu (2011). "How Matchable Are Four Thousand Ontologies on the Semantic Web". In: *The Semantic Web: Research and Applications*. Ed. by Grigoris Antoniou, Marko Grobelnik, Elena Simperl, Bijan Parsia, Dimitris Plexousakis, Pieter De Leenheer, and Jeff Pan. Vol. 6643. Lecture Notes in Computer Science, pp. 290–304.

Isaac, Antoine and Ed Summers (2009-08). *SKOS Simple Knowledge Organization System Primer*. W3C Note. URL: https://www.w3.org/TR/2009/NOTE-skos-primer-20090818/.

Islam, Sharif (2023-01-20). "FAIR digital objects, persistent identifiers and machine actionability". In: *FAIR Connect* 1.1, pp. 29–34. ISSN: 2949799X. https://doi.org/10.3233/FC-230001.

ISO 16684-1 (2019-04). *ISO 16684-1:2019 — Graphic technology — Extensible metadata platform (XMP) — Part 1: Data model, serialization and core properties*. ISO. URL: https://www.iso.org/standard/75163.html.

ISO/IEC 23009-1 (2022-08). *ISO/IEC 23009-1:2022 — Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats*. ISO. URL: https://www.iso.org/standard/83314.html.

ITU-T X.1255 (2013-09-04). *X.1255 : Framework for Discovery of Identity Management Information*. Recommendation. ITU-T X.1255. URL: https://www.itu.int/rec/T-REC-X.1255-201309-I (visited on 2023-03-21).

Iyengar, Jana and Martin Thomson (2021-05). *QUIC: A UDP-Based Multiplexed and Secure Transport*. Request for Comments. RFC 9000. RFC Editor. https://doi.org/10.17487/rfc9000.

Jaradeh, Mohamad Yaser, Allard Oelen, Manuel Prinz, Markus Stocker, and SÃ¶ren Auer (2019). "Open Research Knowledge Graph: A System Walkthrough". In: *Digital Libraries for Open Knowledge*, pp. 348–351. https://doi.org/10.1007/978-3-030-30760-8_31.

Jones, Richard and Neil Jefferies, eds. (2021-09-01). *SWORD 3.0 Specification*. URL: https://swordapp.github.io/swordv3/s (visited on 2022-05-26).

Joras, Matt and Yang Chi (2020-10-21). *How Facebook is bringing QUIC to billions*. URL: https://engineering.fb.com/2020 (visited on 2023-06-13).

Juty, Nick, Nicolas Le Novere, and Camille Laibe (2011-12-02). "Identifiers.Org and MIRIAM Registry: Community Resources to Provide Persistent Identification". In: *Nucleic Acids Research* 40.D1, pp. D580–D586. https://doi.org/10.1093/nar/gkr1097.

Kahn, Robert and Robert Wilensky (1995-05-13). *A Framework for Distributed Digital Object Services*. URL: http://www.cnri.reston.va.us/k-w.html (visited on 2022-05-09).

— (2006-04). "A Framework for Distributed Digital Object Services". In: *International Journal on Digital Libraries* 6.2, pp. 115–123. ISSN: 1432-5012. https://doi.org/10.1007/s00799-005-0128-x. URL: https://www.doi.org/topics/2006_05_02_Kahn_Framework.pdf (visited on 2022-05-05).

Kamdar, Maulik R., Tania Tudorache, and Mark A. Musen (2017-08-07). "A Systematic Analysis of Term Reuse and Term Overlap across Biomedical Ontologies". In: *Semantic Web* 8.6. Ed. by Guo-Qiang Zhang, pp. 853–871. https://doi.org/10.3233/sw-160238.

Kelly, Mike (2016-05-12). *JSON Hypertext Application Language*. Internet-Draft draft-kelly-json-hal-08. Work in Progress. Internet Engineering Task Force. 11 pp. URL: https://datatracker.ietf.org/doc/draft-kelly-json-hal/08/.

Khare, Rohit and Scott Lawrence (2000-05). *Upgrading to TLS Within HTTP/1.1*. Request for Comments. RFC 2817. RFC Editor. https://doi.org/10.17487/rfc2817.

Klein, Martin, Herbert Van de Sompel, Robert Sanderson, Harihar Shankar, Lyudmila Balakireva, Ke Zhou, and Richard Tobin (2014-12-26). "Scholarly Context Not Found: One in Five Articles Suffers from Reference Rot". In: *PLOS ONE* 9.12. Ed. by Judit Bar-Ilan, e115253. https://doi.org/10.1371/journal.pone.0115253

Klímek, Jakub, Petr Škoda, and Martin Nečaský (2019-05-23). "Survey of Tools for Linked Data Consumption". In: *Satya Widya* 10.4, pp. 665–720. ISSN: 22104968. https://doi.org/10.3233/SW-180316.

Kontokostas, Dimitris and Holger Knublauch (2017-07). *Shapes Constraint Language (SHACL)*. W3C Recommendation. URL: https://www.w3.org/TR/2017/REC-shacl-20170720/.

Kunze, John A. and Emmanuelle Bermès (2022-10-17). *The ARK Identifier Scheme*. Internet-Draft draft-kunze-ark-36. Work in Progress. Internet Engineering Task Force. 45 pp. URL: https://datatracker.ietf.org/doc/draft-kunz

Lagoze, Carl, Herbert Van de Sompel, Pete Johnston, Michael Nelson, Robert Sanderson, and Simeon Warner (2008-10-17). *ORE Specification - Abstract Data Model*. Open Archives Initiative. URL: http://www.openarchives.org/ore/1.0/datamodel#Proxies (visited on 2023-01-24).

Lamprecht, Anna-Lena et al. (2021-09-07). "Perspectives on Automated Composition of Workflows in the Life Sciences". In: *F1000Research* 10, p. 897. https://doi.org/10.12688/f1000research.54159.1.

Lannom, Larry, Karsten Peters-von Gehlen, Ivonne Anders, Andreas Pfeil, Alexander Schlemmer, Zach Trautt, and Peter Wittenburg (2022-10-17). *FDO Configuration Types*. Proposed Recommendation PR-ConfigurationTypes-2.1-20221017. https://doi.org/10.5281/zenodo.7825703.

Lannom, Larry, Ulrich Schwardmann, Christophe Blanchi, Ivonne Anders, Claus Weiland, and Peter Wittenburg (2022-11-19). *FAIR Digital Objects Roadmap*. version 5 November 2022. Draft FAIR Digital Objects Roadmap-v5-22-11-19. https://doi.org/10.5281/zenodo.7824673.

Lannom, Larry, Ulrich Schwardmann, Cristophe Blanchi, and Peter Wittenburg (2022-06-08). *Typing FAIR Digital Objects*. Proposed Recommendation PR-TypingFDOs-2.0-20220608. https://doi.org/10.5281/zenodo.782

Lanthaler, Markus (2021-07-13). *Hydra Core Vocabulary*. URL: http://www.hydra-cg.com/spec/latest/core/ (visited on 2023-06-14).

Lassila, Ora and Ralph R. Swick (1999-02). *Resource Description Framework (RDF) Model and Syntax Specification*. W3C Recommendation. URL: http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/.

Lebo, Timothy, Deborah McGuinness, and Satya Sahoo (2013-04). *PROV-O: The PROV Ontology*. W3C Recommendation. URL: https://www.w3.org/TR/2013/REC-prov-o-20130430/.

Lebo, Timothy and Luc Moreau (2013-04). *Linking Across Provenance Bundles*. W3C Note. URL: https://www.w3.org/TR

OSI (2022-10-11). *Licenses & Standards*. Open Source Initiative. URL: https://opensource.org/licenses (visited on 2023-01-24).

Linked Data (2015). W3C. URL: https://www.w3.org/standards/semanticweb/data (visited on 2022-05-26).

Liu, Kevin and David Booth (2007-06). *Web Services Description Language (WSDL) Version 2.0 Part 0: Primer*. W3C Recommendation. URL: https://www.w3.org/TR/2007/REC-wsdl20-primer-20070626/.

Loo, Tina, ed. (2022). *First International Conference on FAIR Digital Objects*. Pensoft Publishers. https://doi.org/10.3897/

McMurry, Julie A. et al. (2017-06). "Identifiers for the 21st century: How to design, provision, and reuse persistent identifiers to maximize utility and impact of life science data". In: *PLOS Biology* 15.6, e2001414. https://doi.org/10.1371/journal.pbio.2001414.

MDN (2023-04-10). "HTTP Content Negotiation". In: *Web technology for developers*. URL: https://developer.mozilla.org/e (visited on 2023-06-13).

Meroño-Peñuela, Albert, Pasquale Lisena, and Carlos Martínez-Ortiz (2021a). "Conclusion and Future Challenges". In: *Web Data Apis for Knowledge Graphs: Easing Access to Semantic Data for Application Developers*. Synthesis Lectures on Data, Semantics, and Knowledge, pp. 77–80. https://doi.org/10.1007/978-3-031

— (2021b). "Web Data Apis over SPARQL". In: *Web Data Apis for Knowledge Graphs: Easing Access to Semantic Data for Application Developers*. Synthesis Lectures on Data, Semantics, and Knowledge, pp. 27–38. https://doi.org/10.1007/978-3-031-01917-3_3.

Michael Hausenblas et al. (2012-01-22). *5-star Open Data*. URL: http://5stardata.info/ (visited on 2023-01-24).

Microdata (2023-06-12). In: *HTML Living Standard*. WHATWG. URL: https://html.spec.whatwg.org/multipage/microdata.h (visited on 2023-06-13).

Miller, Darrel, Jeremy Whitlock, Marsh Gardiner, Mike Ralphson, Ron Ratovsky, and Uri Sarid, eds. (2021-02-15). *OpenAPI Specification v3.1.0*. OpenAPI Initiative. URL: https://spec.openapis.org/oas/v3.1.0.html (visited on 2023-03-21).

Mons, Barend, Cameron Neylon, Jan Velterop, Michel Dumontier, Luiz Olavo Bonino Silva Santos, and Mark D. Wilkinson (2017-03-07). "Cloudy, Increasingly FAIR; Revisiting the FAIR Data Guiding Principles for the European Open Science Cloud". In: *Information Services & Use* 37.1, pp. 49–56. ISSN: 18758789. https://doi.org/10.3233/ISU-170824.

NCBO BioPortal (2022). URL: https://bioportal.bioontology.org/ontologies (visited on 2022-05-26).

Neumann, Andy, Nuno Laranjeiro, and Jorge Bernardino (2021-07-01). "An Analysis of Public REST Web Service Apis". In: *IEEE Transactions on Services Computing* 14.4, pp. 957–970. ISSN: 1939-1374. https://doi.org/10.1109/TSC.2018.2847344.

Norris, Emma, Janna Hastings, Marta M. Marques, Ailbhe N. Finnerty Mutlu, Silje Zink, and Susan Michie (2021-03). "Why and how to engage expert stakeholders in ontology development: insights from social and behavioural sciences". In: *Journal of Biomedical Semantics* 12.1. https://doi.org/10.1186/s13326-021-0

Nottingham, Mark (2017-10). *Web Linking*. Request for Comments. RFC 8288. RFC Editor. https://doi.org/10.17487/rfc82

Nurdiati, Sri and Cornelis Hoede (2008). *25 years development of knowledge graph theory: the results and the challenge*. Memorandum. No. 2/1876. URL: https://purl.utwente.nl/publications/64931.

OCLC (2010-05). *"info" URI Registry (Frozen)*. OCLC. URL: http://info-uri.info/ (visited on 2023-01-24).

OGP (2022). *The Open Graph Protocol*. URL: https://ogp.me/ (visited on 2022-05-26).

OpenStand (2017-12-28). *The Modern Standards Paradigm - Five Key Principles*. OpenStand. URL: https://open-stand.org/about-us/principles/ (visited on 2023-01-24).

Page, Kevin R., David C. De Roure, and Kirk Martinez (2011). "REST and Linked Data". In: *Proceedings of the Second International Workshop on RESTful Design - WS-REST '11*. ACM Press. https://doi.org/10.1145/1967428.1967435. URL: http://eprints.soton.ac.uk/id/eprint/272098.

Pantos, Roger and William May (2017-08). *HTTP Live Streaming*. Request for Comments. RFC 8216. RFC Editor. https://doi.org/10.17487/rfc8216.

Parecki, Aaron (2017-05-23). *Micropub*. W3C Recommendation. W3C, Social Web Working Group. URL: https://www.w3.org/TR/2017/REC-micropub-20170523/.

Polleres, Axel, Maulik Rajendra Kamdar, Javier David Fernández, Tania Tudorache, and Mark Alan Musen (2020-01-31). "A More Decentralized Vision for Linked Data". In: *Satya Widya* 11.1, pp. 101–113. ISSN: 22104968. https://doi.org/10.3233/SW-190380.

Reilly, Sean (2009-11-12). *Digital Object Interface Protocol Version 1.0*. DONA Foundation. URL: https://www.dona.net/doipv1doc (visited on 2022-05-26).

Rescorla, Eric (2000-05). *HTTP Over TLS*. Request for Comments. RFC 2818. RFC Editor. https://doi.org/10.17487/rfc281

Riccardi, Demian et al. (2022-03). "Towards improved FAIRness of the ThermoML Archive". In: *Journal of Computational Chemistry* 43.12, pp. 879–887. https://doi.org/10.1002/jcc.26842.

Rice, Adam, Ian Hickson, Anne van Kesteren, and Yutaka Hirano (2022-10-25). *WebSockets Standard*. URL: https://websockets.spec.whatwg.org/ (visited on 2022-05-26).

Rocca-Serra, Philippe et al. (2023-05). "The FAIR Cookbook - the essential resource for and by FAIR doers". In: *Scientific Data* 10.1. https://doi.org/10.1038/s41597-023-02166-3.

Sandvine (2022). *Global Internet Phenomena Report 2022*. URL: https://www.sandvine.com/global-internet-phenomena-repo (visited on 2022-05-26).

Sauermann, Leo, Richard Cyganiak, Danny Ayers, and Max Völkel (2008-12). *Cool URIs for the Semantic Web*. Interest Group Note. W3C Interest Group Note. URL: http://www.w3.org/TR/cooluris/.

Schema.Org - Schema.Org (2022). URL: https://schema.org/ (visited on 2022-05-26).

Schema.Org Actions (2022). URL: https://schema.org/docs/actions.html (visited on 2022-05-26).

Schreiber, Guus and Yves Raimond (2014-06). *RDF 1.1 Primer*. W3C Note. URL: http://www.w3.org/TR/2014/NOTE-rdf1

Schultes, Erik, Barbara Magagna, Kristina Maria Hettne, Robert Pergl, Marek Suchánek, and Tobias Kuhn (2020). "Reusable FAIR Implementation Profiles as Accelerators of FAIR Convergence". In: *Lecture Notes in Computer Science*, pp. 138–147. https://doi.org/10.1007/978-3-030-65847-2_13. doi: 10.31219/osf.io/2p85g.

Schultes, Erik and Peter Wittenburg (2019). "FAIR Principles and Digital Objects: Accelerating Convergence on a Data Infrastructure". In: *Data Analytics and Management in Data Intensive Domains: 20th International Conference, DAMDID/RCDL 2018, Moscow, Russia, October 9–12, 2018, Re-*

vised Selected Papers. Ed. by Yannis Manolopoulos and Sergey Stupnikov. Vol. 1003. Communications in Computer and Information Science, pp. 3–16. https://doi.org/10.1007/978-3-030-23584-0_1.
Schwardmann, Ulrich and Tibor Kálmán (2022-10-12). "Two Examples on How FDO Types Can Support Machine and Human Readability". In: *Research Ideas and Outcomes* 8. https://doi.org/10.3897/rio.8.e96014.
Schwardmann, Ulrich, George Strawn, Robert Quick, and Peter Wittenburg (2022-10-17). *DOIP Endorsement Request*. Enforsement Request PED-DOIPEndorsement-1.1-20221017. https://doi.org/10.5281/zenodo.782
Semmler, Tido et al. (2022). *IPCC DDC: AWI AWI-CM1.1MR model output prepared for CMIP6 CMIP historical*. World Data Center for Climate (WDCC) at DKRZ. HDL: 21.14100/2fcf49d3-0608-3373-a47f-0e721b7eaa87 URL: https://www.wdc-climate.de/ui/entry?acronym=C6CMAWAWMhi.
Singhal, Amit (2012-05-16). *Introducing the Knowledge Graph: things, not strings*. Google. URL: https://blog.google/proc (visited on 2023-05-18).
Smith, Barry et al. (2007-11). "The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration". In: *Nature Biotechnology* 25.11, pp. 1251–1255. https://doi.org/10.1038/nbt1346.
Soiland-Reyes, Stian, Leyla Jael Castro, Daniel Garijo, Marc Portier, Carole Goble, and Paul Groth (2022-10). "Updating Linked Data practices for FAIR Digital Object principles". In: *Research Ideas and Outcomes* 8. https://doi.org/10.3897/rio.8.e94501.
Soiland-Reyes, Stian, Peter Sefton, et al. (2022-10). "Creating lightweight FAIR Digital Objects with RO-Crate". In: *Research Ideas and Outcomes* 8. https://doi.org/10.3897/rio.8.e93937.
SPARQL WG (2013-03). *SPARQL 1.1 Overview*. W3C Recommendation. The W3C SPARQL Working Group. URL: https://www.w3.org/TR/2013/REC-sparql11-overview-20130321/ (visited on 2022-05-26).
Speicher, Steve, John Arwe, and Ashok Malhotra, eds. (2015-02-26). *Linked Data Platform 1.0*. W3C Recommendation. W3C Linked Data Platform Working Group. URL: https://www.w3.org/TR/2015/REC-ldp-20150226/
Sporny, Manu and Amy Guy (2023-01-01). *Media Types with Multiple Suffixes*. Internet-Draft draft-ietf-mediaman-suffixes-03. Work in Progress. Internet Engineering Task Force. URL: https://datatracker.ietf.org/doc/draft-ie
Sporny, Manu, Ivan Herman, Ben Adida, and Mark Birbeck (2015-03). *RDFa 1.1 Primer - Third Edition*. W3C Note. URL: https://www.w3.org/TR/2015/NOTE-rdfa-primer-20150317/.
Sporny, Manu, Dave Longley, Gregg Kellogg, Markus Lanthaler, Pierre-Antoine Champin, and Niklas Lindström (2020-07-16). *JSON-LD 1.1*. Ed. by Gregg Kellogg, Pierre-Antoine Champin, and Dave Longley. W3C Recommendation. URL: https://www.w3.org/TR/2020/REC-json-ld11-20200716/.
Stallings, William (1990). *Handbook of Computer-Communications Standards: The Open Systems (OSI) Model and OSI-related Standards*. 2nd ed. Carmel, IN, USA: Sams. ISBN: 978-0-672-22697-7.
Stanczyk, Stefan K (1987). "Process Modelling for Information System Description". In: *The Open University*. https://doi.org/10.21954/ou.ro.0000f821.
Stefi, Anisa (2015). *Do Developers Make Unbiased Decisions? - The Effect of Mindfulness and Not-Invented-Here Bias on the Adoption of Software Components*. https://doi.org/10.18151/7217489.
Stefi, Anisa and Thomas Hess (2015). "To Develop or to Reuse? Two Perspectives on External Reuse in Software Projects". In: *Software Business*. Ed. by João M. Fernandes, Ricardo J. Machado, and Krzysztof Wnuk. Vol. 210. Lecture Notes in Business Information Processing, pp. 192–206. https://doi.org/10.1007/97
Sun, Sam, Larry Lannom, and Brian P. Boesch (2003-11). *Handle System Overview*. Request for Comments. RFC 3650. RFC Editor. https://doi.org/10.17487/rfc3650.
Sun, Sam, Sean Reilly, Larry Lannom, and Jason Petrone (2003-11). *Handle System Protocol (Ver 2.1) Specification*. Request for Comments. RFC 3652. RFC Editor. https://doi.org/10.17487/rfc3652.
Thompson, Henry, Sandy Gao, David Beech, Murray Maloney, Noah Mendelsohn, and Michael Sperberg-McQueen (2012-04). *W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures*. W3C Recommendation. URL: https://www.w3.org/TR/2012/REC-xmlschema11-1-20120405/.
Thornton, Katherine, Harold Solbrig, Gregory S. Stupp, Jose Emilio Labra Gayo, Daniel Mietchen, Eric Prud, and Andra Waagmeester (2019). "Using Shape Expressions (ShEx) to Share RDF Data Models and to Guide Curation with Rigorous Validation". In: *The Semantic Web: 16th International Conference, ESWC 2019, Portorož, Slovenia, June 2–6, 2019, Proceedings*. Ed. by Pascal Hitzler et al. Vol. 11503. Lecture Notes in Computer Science, pp. 606–620. https://doi.org/10.1007/978-3-030-21348-0_39.
Tirmizi, Syed, Stuart Aitken, Dilvan A Moreira, Chris Mungall, Juan Sequeda, Nigam H Shah, and Daniel P Miranker (2011). "Mapping between the OBO and OWL Ontology Languages". In: *Journal of Biomedical Semantics* 2 (Suppl 1), S3. https://doi.org/10.1186/2041-1480-2-s1-s3.

Tupelo-Schneck, Robert and Larry Lannom (2022-03-22). "Brief Introduction to Cordra & DOIP". RDA FAIR DO Fabric. URL: https://www.rd-alliance.org/sites/default/files/Cordra.2022.pdf (visited on 2023-01-25).

Turcoane, Ovidiu (2014-09-30). "Linked Data, JSON-LD and the Semantics of Cultural and Scientific Heritage". In: *Digital Presentation and Preservation of Cultural and Scientific Heritage* 4, pp. 95–105. ISSN: 2535-0366. https://doi.org/10.55630/dipp.2014.4.11.

Van de Sompel, Herbert, Martin Klein, et al. (2022-07-27). *FAIR Signposting Profile*. URL: https://signposting.org/FAIR/ (visited on 2023-01-05).

Van de Sompel, Herbert, Michael Nelson, and Robert Sanderson (2013-12). *HTTP Framework for Time-Based Access to Resource States – Memento*. Request for Comments. RFC 7089. RFC Editor. https://doi.org/10.17487/rfc7089.

Verborgh, Ruben (2018-12-28). *Designing a Linked Data Developer Experience*. URL: https://ruben.verborgh.org/blog/201 (visited on 2022-05-26).

Verborgh, Ruben and Miel Vander Sande (2020-01-31). "The Semantic Web Identity Crisis: In Search of the Trivialities That Never Were". In: *Satya Widya* 11.1, pp. 19–27. ISSN: 22104968. https://doi.org/10.3233/SW-19037

Verburg, Maaike, Robert Huber, Clement Jonquet, and Daniel Garijo (2023-04). *FAIR-IMPACT project response to "FAIR Assessment Tools: Towards an "Apples to Apples" Comparisons"*. Version 1.0. https://doi.org/10.5281/zenodo.7848102.

W3C OWL Working Group, ed. (2012-12). *OWL 2 Web Ontology Language Document Overview (Second Edition)*. W3C Recommendation. URL: https://www.w3.org/TR/2012/REC-owl2-overview-20121211/.

W3Techs (2023). *Usage Statistics of JSON-LD for Websites*. 2023-05. W3Techs - World Wide Web Technology Surveys. URL: https://w3techs.com/technologies/details/da-jsonld (visited on 2023-05-18).

Weigel, Tobias et al. (2018). *RDA Recommendation on PID Kernel Information*. https://doi.org/10.15497/rda00031.

Weiland, C. et al. (2022-01-29). *FDO Forum Document Standards*. Working Draft WD-DocProcessStd-1.1-20220129. (internal draft). URL: https://drive.google.com/file/d/1lPNBBROjEoZ6fTfrtdqcMa3Q2G27PoC_ (visited on 2022-11-30).

Weiland, Claus, Sharif Islam, Daan Broder, Ivonne Anders, and Peter Wittenburg (2022-11-19). *FDO Machine Actionability*. Version 2.2. Proposed Recommendation PR-MachineActionDef-2.2-20221119. https://doi.org/10.5281/zenodo.7825650.

Wieczorek, John et al. (2012-01-06). "Darwin Core: An Evolving Community-Developed Biodiversity Data Standard". In: *PLOS ONE* 7.1. Ed. by Indra Neil Sarkar, e29715. https://doi.org/10.1371/journal.pone.0029715.

Wilde, Erik (2013-03). *The 'profile' Link Relation Type*. Request for Comments. RFC 6906. RFC Editor. https://doi.org/10.17487/rfc6906.

Wilkinson, Mark D, Susanna-Assunta Sansone, Grootveld Marjan, Josefine Nordling, Richard Dennis, and David Hecker (2022-12). *FAIR Assessment Tools: Towards an "Apples to Apples" Comparisons*. https://doi.org/10.5281/zenodo.7463421.

Wilkinson, Mark D. et al. (2016-03-15). "The FAIR Guiding Principles for Scientific Data Management and Stewardship". In: *Scientific Data* 3.1. https://doi.org/10.1038/sdata.2016.18.

Wilkinson, Sean R., Greg Eisenhauer, Anuj J. Kapadia, Kathryn Knight, Jeremy Logan, Patrick Widener, and Matthew Wolf (2022). "F*** Workflows: When Parts of FAIR Are Missing". In: *arXiv*. https://doi.org/10.48550/ar

Williams, Antony J. et al. (2012-11). "Open PHACTS: Semantic Interoperability for Drug Discovery". In: *Drug Discovery Today* 17.21-22, pp. 1188–1198. https://doi.org/10.1016/j.drudis.2012.05.016.

Wittenburg, Peter, Ivonne Anders, et al. (2022). "FAIR Digital Object Demonstrators 2021". In: *Zenodo*. https://doi.org/10.5281/zenodo.5872645.

Wittenburg, Peter, George Strawn, Barend Mons, Luiz Bonino, and Erik Schultes (2019-01-06). "Digital Objects as Drivers towards Convergence in Data Infrastructures". In: *B2Share*. https://doi.org/10.23728/b2share.b605d8

Wolstencroft, Katherine et al. (2013-05-02). "The Taverna Workflow Suite: Designing and Executing Workflows of Web Services on the Desktop, Web or in the Cloud". In: *Nucleic Acids Research* 41.W1, W557–W561. https://doi.org/10.1093/nar/gkt328.

Wolstencroft, Katy et al. (2011-05-26). "RightField: Embedding Ontology Annotation in Spreadsheets". In: *Bioinformatics* 27.14, pp. 2021–2022. https://doi.org/10.1093/bioinformatics/btr312.

Wood, David, Richard Cyganiak, and Markus Lanthaler (2014-02-25). *RDF 1.1 Concepts and Abstract Syntax*. W3C Recommendation. URL: https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/.

Wright, Austin, Henry Andrews, Ben Hutton, and Greg Dennis (2022-06-10). *JSON Schema: A Media Type for Describing JSON Documents*. Internet-Draft. Work in Progress. Internet Engineering Task Force. URL: https://datatracker.ietf.org/doc/draft-bhutton-json-schema/01/.

Zarras, Apostolos (2004). "A Comparison Framework for Middleware Infrastructures." In: *The Journal of Object Technology* 3.5, p. 103. https://doi.org/10.5381/jot.2004.3.5.a2.

FDO Specifications

FDO-Overview-PEN-2.0 Ivonne Anders, Christophe Blanchi, Daan Broder, Maggie Hellström, Sharif Islam, Thomas Jejkal, Larry Lannom, Karsten Peters-von Gehlen, et al. (2023-01-18). *FAIR Digital Object Technical Overview*. Version PEN 2.0. Proposed Recommendation Full FDO Overview PEN-2.0-v2. https://doi.org/10.5281/zenodo.7824714.

PED-DOIPEndorsement-1.1 Ulrich Schwardmann, George Strawn, Robert Quick, and Peter Wittenburg (2022-10-17). *DOIP Endorsement Request*. Enforsement Request PED-DOIPEndorsement-1.1-20221017. https://doi.org/10.5281/zenodo.7824796

PEN-FDO-Upload Christophe Blanchi, Daan Broeder, Thomas Jejkal, Islam Sharif, Alexander Schlemmer, Dieter van Uytvanck, and Peter Wittenburg (2022-10-17). *FDO – Upload of FDO*. Proposed Endorsement Note PEN-FDO-Upload-1.1-20221017. https://doi.org/10.5281/zenodo.7825549.

PR-ConfigurationTypes-2.1 Larry Lannom, Karsten Peters-von Gehlen, Ivonne Anders, Andreas Pfeil, Alexander Schlemmer, Zach Trautt, and Peter Wittenburg (2022-10-17). *FDO Configuration Types*. Proposed Recommendation PR-ConfigurationTypes-2.1-20221017. https://doi.org/10.5281/zenodo.7825703.

PR-Granularity-2.2 Maggie Hellström, Carlo Zwölf, and Peter Wittenburg (2022-10-17). *FDO – Granularity, Versioning, Mutability*. Proposed Recommendation PR-Granularity-2.2-20221017. https://doi.org/10.5281/zenodo.7825686.

PR-KernelAtributues-2.0 Daan Broeder, Peter Wittenburg, Ivonne Anders, and Karsten Peters-von Gehlen (2022-10-17). *FDO – Kernel Attributes & Metadata*. Proposed Recommendation PR-FDO-KernelAttributesAndMetadata-2.0-20221017. https://doi.org/10.5281/zenodo.7825693.

PR-MachineActionDef-2.2 Claus Weiland, Sharif Islam, Daan Broder, Ivonne Anders, and Peter Wittenburg (2022-11-19). *FDO Machine Actionability*. Version 2.2. Proposed Recommendation PR-MachineActionDef-2.2-20221119. https://doi.org/10

PR-PIDProfileAttributes-2.1 Ivonne Anders, Maggie Hellström, Sharif Islam, Thomas Jejkal, Larry Lannom, Ulrich Schwardmann, and Peter Wittenburg (2022-10-17). *FDO PID Profiles & Attributes*. Proposed Recommendation PR-PIDProfileAttribute 2.1-20221017. https://doi.org/10.5281/zenodo.7825630.

PR-RequirementSpec-3.0 Ivonne Anders, Christophe Blanchi, Daan Broder, Maggie Hellström, Sharif Islam, Thomas Jejkal, Larry Lannom, Karsten Peters-von Gehlen, et al. (2023-01-12). *FDO Forum FDO Requirement Specifications*. Version 3.0. Proposed Recommendation PR-RequirementSpec-3.0. https://doi.org/10.5281/zenodo.7782262.

PR-TypingFDOs-2.0 Larry Lannom, Ulrich Schwardmann, Cristophe Blanchi, and Peter Wittenburg (2022-06-08). *Typing FAIR Digital Objects*. Proposed Recommendation PR-TypingFDOs-2.0-20220608. https://doi.org/10.5281/zenodo.782

WD-DocProcessStd-1.1 C. Weiland et al. (2022-01-29). *FDO Forum Document Standards*. Working Draft WD-DocProcessStd-1.1-20220129. (internal draft). URL: https://drive.google.com/file/d/1lPNBBROjEoZ6fTfrtdqcMa3Q2G27PoC_ (visited on 2022-11-30).

WD-ImplAttributesTypesProfiles Christophe Blanchi, Maggie Hellström, Larry Lannom, Andreas Pfeil, Ulrich Schwardmann, and Peter Wittenburg (2023-03-14). *Implementation of Attributes, Types, Profiles and Registries*. Working Draft WD-Implementation-of-Attributes-0.4-20230314. https://doi.org/10.5281/zenodo.7825