

Practical web-based FDOs with RO-Crate and FAIR Signposting

This agenda/notes: <https://bit.ly/fdo-signposting>

Tutorial at FDO Summit 2024

- **Instructors:** Leyla Jael Castro, Stian Soiland-Reyes, Jonas Grieb & Claus Weiland
- **When:** Tue 2024-03-19 13:30 – 15:00 CET
- **Where:** ST3 DIN (see  Training Day at the FDO Conference-web-final.pdf)
- **Material/slides:**  2024-03-19 FDO2024 training RO-Crate Signposting training

Research Object Crate ([RO-Crate](#)) is a lightweight method to package research outputs along with their metadata. [Signposting](#) provides a simple yet powerful approach to navigate FAIR aspects of the scholarly objects on the Web.

Combining these technologies form a "webby" implementation of the [FAIR Digital Object principles](#) which is suitable for retrofitting to existing data infrastructures or even for ad-hoc research objects using regular Web hosting platforms.

In this tutorial we will give a brief introduction to the FDO principles, and show how they have been realised in deployments using HTTP, HTML and JSON. We demonstrate building FDO metadata using RO-Crate editors, briefly introducing the Web standard JSON-LD. Finally we publish the FDO using Signposting on the Web and assign a persistent identifier.

This session is organised as an interactive walk-through, with opportunity to discuss use cases and challenges. It aims to give the participants an overview of the technologies in order to continue training and experimentation at their own pace following the summit. Brief knowledge of Web technology (HTTP, HTML, JSON) is an advantage, but not a requirement.

Learning objectives

1. Knowledge of FDO principles and how they can be practically applied
2. Ability to create research objects FDOs with brief metadata
3. Understanding of considerations for FDO deployment on the Web

Agenda

Time	
13:30	Welcome - FDO flavours [CW]
13:35	Brief introduction to JSON-LD and "just enough Linked Data" [LJC]

13:45	HTTP signposting as a lightweight FDO overlay [SSR] Tutorial: https://github.com/stain/signposting-tutorial
14:00	Packaging data with their metadata in RO-Crate [SSR] Tutorial: https://www.researchobject.org/packaging_data_with_ro-crate/
14:10	Q & A
14:20	Demonstration of RO-Crate and Signposting in biodiversity [JG]
14:30	Demo: Adding metadata using the Crate-O graphical editor [SSR]
14:40	Discussion
15:00	End of session (Coffee Break before ST4)

Agenda is subject to change.

Additional material

- **Demonstration of RO-Crate and Signposting in biodiversity**
 - Retrieve Signposting headers:
 - `curl -I "https://wildlive.senckenberg.de/captureevent/wildlive/7df91e6d148a386cc674"`
- **Signposting Python library**
 - <https://pypi.org/project/signposting/>
-

Notes

Please feel free to help take notes during the session!

Q: How do different JSON-LD @context avoid collision? Namespace differences?

A: It's important to note that the @context do not define the namespace, but defines a JSON-LD Context document which can map in the namespaces.

It's possible to use multiple contexts, or one context can inherit from another. They should ideally not be colliding in terms, if not the latest one decides. To avoid collisions you can either map explicitly terms to an alias in the context (As RO-Crate does for a couple of imported terms), or you can use namespace prefixes defined by the context, like "schema:TrainingMaterial" – similar to in other Linked Data formats.

Q: Why do link relations rel=describedby etc not match the FDO terms?

A: The link relations are all defined by IANA RFC standards and predate FDO. The full list is at <https://www.iana.org/assignments/link-relations> where each have a corresponding standard.

You can register additional link relations or use own by using URIs `rel="<http://example.com/soup>"` but this is probably a sign that you really need to make a metadata resource e.g. in JSON-LD – Signposting is mainly navigational and semantics should be in metadata.

Signposting to operations: <https://signposting.org/FAIRiCat/>

Link relations `rel="api-catalog"` or `rel="service-doc"`, `rel="service-desc"`, `rel="service-meta"`