RO-Crate: A framework for packaging research products into FAIR Research Objects

Carole Goble

The University of Manchester

ELIXIR-UK

https://orcid.org/0000-0003-1219-2137

@caroleannegoble

Stian Soiland-Reyes

The University of Manchester

BioExcel Centre of Excellence

The University of Amsterdam

https://orcid.org/0000-0001-9842-9718

@soilandreyes

Horizon2020
European Union Funding
for Research & Innovation

This work is licensed under a
Creative Commons Attribution 4.0 International License

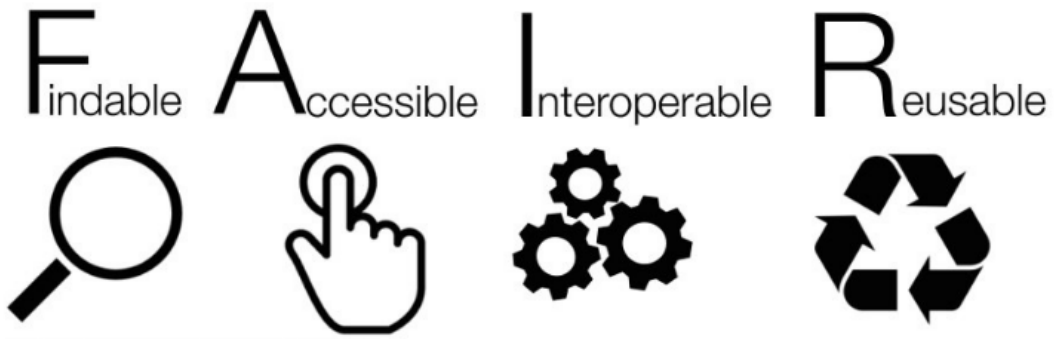
researchobject.org

tl;dr

Web standards-based metadata framework for bundling resources with their context into citable reproducible packages

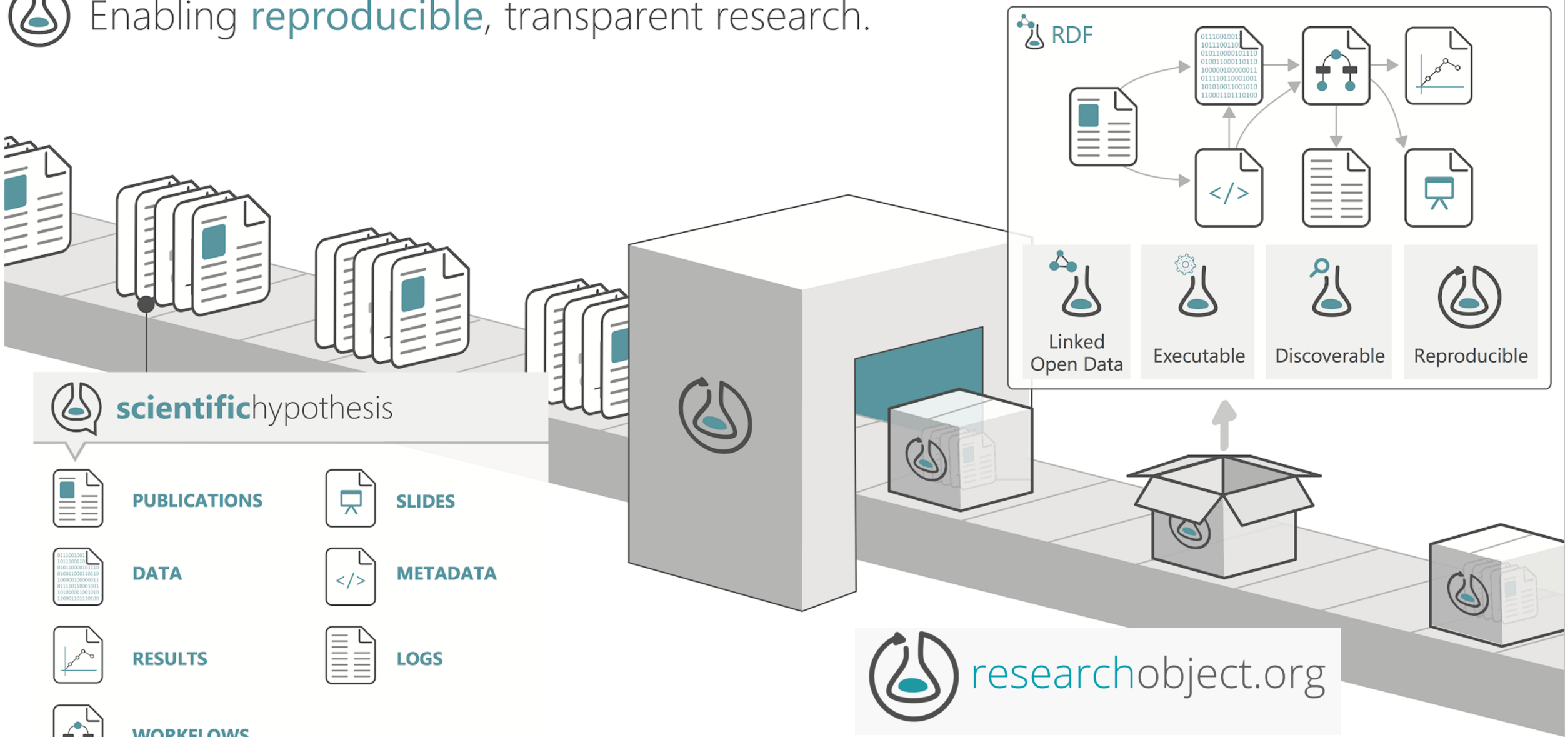
Machine actionable Metadata + Identifiers + Web protocols => FAIR

- What and Why
- Examples
- How and Tools
- Alignment with FDOF



Bechhofer et al (2013) Why linked data is not enough for scientists https://doi.org/10.1016/j.future.2011.08.004
Bechhofer et al (2010) Research Objects: Towards Exchange and Reuse of Digital Knowledge, https://eprints.soton.ac.uk/268555/

Enabling **reproducible**, transparent research.

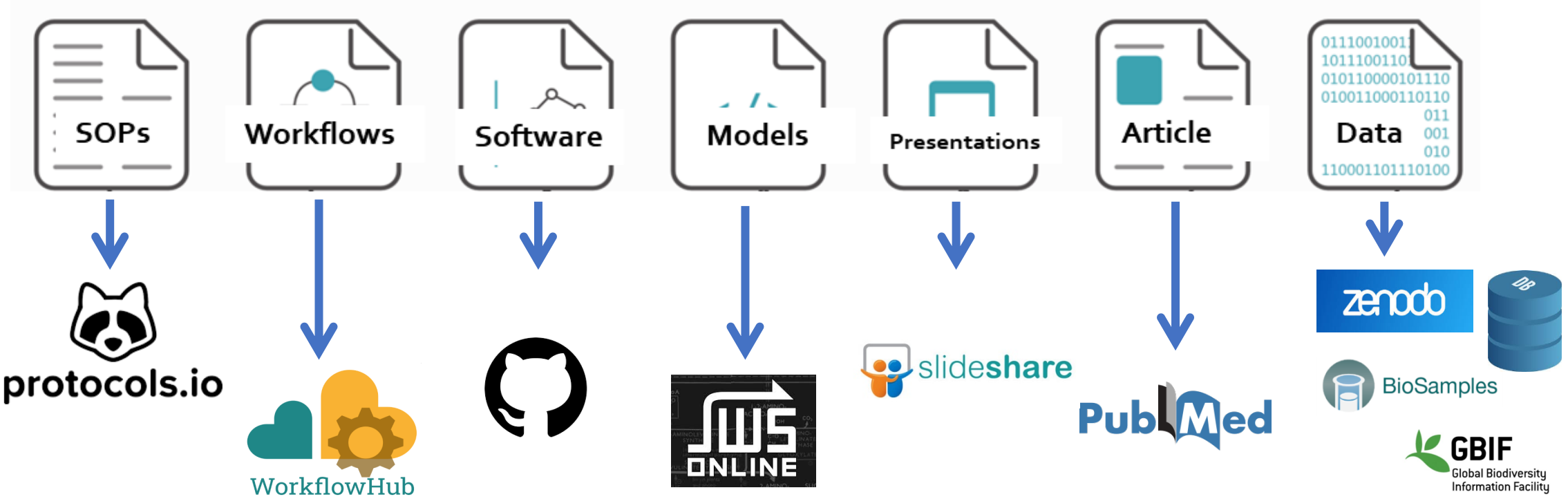


https://www.researchobject.org/

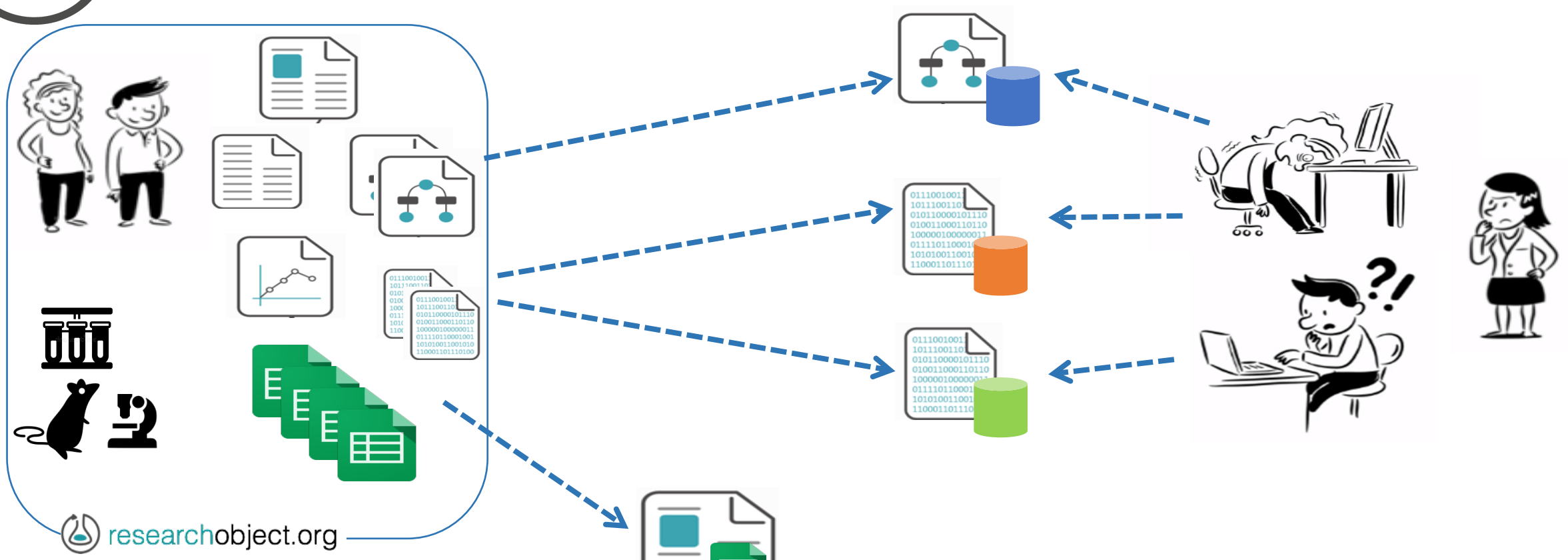
Many Objects are the Outcomes of Research

All are first class citizens and are required to make research FAIR+R

Each object has its own metadata and repositories



Scattered Reporting and Reading



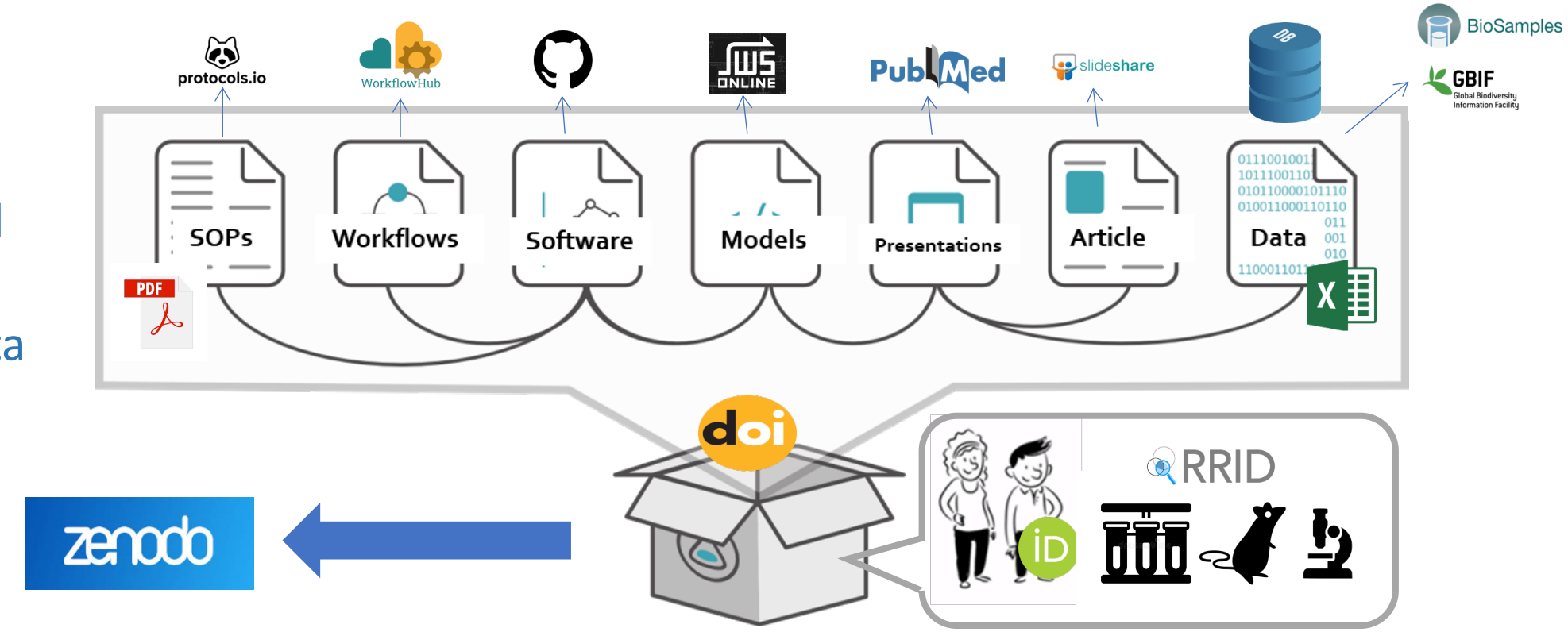
Contextualised
Active, Unified
Semantic linking

Buried in a PDF
figure

De-contextualised
Static, Fragmented
Lost Semantic linking
Rebuild to be reproducible

Encapsulated content and references to external resources

integrated view over fragmented resources using PIDs and metadata



The RO package has its own metadata, can be registered and deposited in its own right, unpackaged and accessed, activated and reproduced if appropriate

self-describing, chiefly **metadata**, objects

RO Metadata file

Structured metadata about the RO and content

type
id
description
datePublished
…

author

organisation

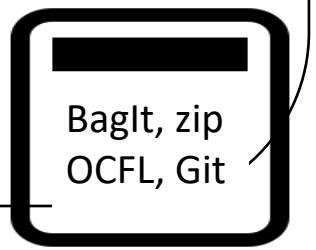
license

RO Content

files

directories

links to web resources



BagIt, zip
OCFL, Git

Archive file format / packaging system

self-describing, chiefly **metadata**, objects

PID

PID

RO Metadata file

Structured metadata about the RO and content

type
id
description
datePublished
…

author

organisation

license

Linked Data approach

RO Content

PID

type, id
description
datePublished
creator
size
format …

image file

PID

PID

directory of data

links to web resources

PDF

https://zenodo.org/record/3541888

https://github.com/o/script

BagIt, zip
OCFL, Git

Archive file format / packaging system

Self-describing, chiefly **metadata**, objects
Strict Structure, Open ended content

RO-Crate

http://www.researchobject.org/ro-crate/

Standard Web Mark-up

JSON-LD

schema.org

How do we describe the metadata?

- PIDs + JSON-LD + Schema.org descriptors
- Opinionated profile of schema.org
- Linked Data by Stealth: JSON with gradual path to extensibility with LD – e.g. ad-hoc terms
- Example-driven documentation

How can I add additional metadata?

- Schema.org, domain ontologies

How do I define a checklist of what is expected to be in a type of RO?

- RO-Crate Profiles

Adding new or ad hoc vocabulary terms

Context terms must ultimately map to HTTP(s) URIs which poses challenges for crate-authors wishing to use their own vocabularies.

RO-Crate provides some strategies to add a new term (a Class or Property) that is not in Schema.org or another published vocabulary, so that there is a stable URI that can be added to the @context.

Choosing URLs for ad hoc terms

For projects that have their own web-presence, URLs MAY be defined there and SHOULD resolve to useful content. For example for a project with web page https://criminalcharacters.com/ the property `education` could have a URL: https://criminalcharacters.com/vocab#education which resolves to an HTML page that explains the term using HTML anchors:

```
<div id="education">
  <h1>Property: education</h1>
  <p>Literacy of prisoner. Prison authorities would record the prisoner's statement as to whether they could
  </p>
</div>
```

Tip

Ensure you have a consistent use of `http` or `https` (preferring https) as well as consistent path `/vocab` vs `/vocab/` vs `/vocab/index.html` (preferring the shortest that is also visible in browser).

For ad hoc terms where the crate author does not have the resources to create and maintain an HTML page, authors may use the RO-Crate public namespace (`https://w3id.org/ro/terms/`) to reserve their terms. For example, an ad-hoc URL MAY be used in the form `https://w3id.org/ro/terms/criminalcharacters#education` where `criminalcharacters` is acting as a *namespace* for one or more related terms like `education` . Ad-hoc namespaces under `https://w3id.org/ro/terms/` are available on first-come-first-serve basis; to avoid clashes, namespaces SHOULD be registered by submitting terms and definitions to the RO-Crate terms project.

RO-Crate Structure

https://w3id.org/ro/crate/1.1

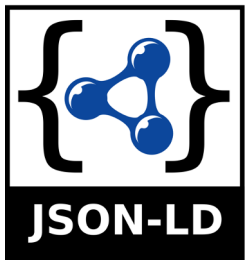
The structure an *RO-Crate* MUST follow is:

```
<RO-Crate root directory>/
|   ro-crate-metadata.json     # RO-Crate Metadata File MUST be present
|   ro-crate-preview.html      # RO-Crate Website homepage MAY be present
|   ro-crate-preview_files/    # MAY be present
|    | [other RO-Crate Website files]
|   [payload files and directories]  # 0 or more
```

JSON with a flat list of:

- **Data** Entities (e.g. files, dirs, DBs)
- **Contextual** Entities (e.g. people)

Objects reference each other by `@id`



```
{
  "@type": "Dataset",
  "@id": "./",
  "publisher": {"@id": "https://ror.org/03f0f6041"},
  "author": {"@id": "https://orcid.org/0000-0002-3545-944X"}
},
{
  "@id": "https://ror.org/03f0f6041",
  "@type": "Organization",
  "name": "University of Technology Sydney"
},
{
  "@id": "https://orcid.org/0000-0002-3545-944X",
  "@type": "Person",
  "affiliation": {"@id": "https://ror.org/03f0f6041"},
  "email": "peter.sefton@uts.edu.au",
  "name": "Peter Sefton"
}
```

RO-CRATE 1.1

1. About this document
2. Introduction
3. Terminology
4. RO-Crate Structure
5. Metadata of the RO-Crate
6. Root Data Entity
7. Data Entities
8. Contextual Entities
   - Contextual vs Data entities
   - Identifiers for contextual entities
   - People
   - Organizations as values
   - Contact information
   - Publications via citation property
   - Publisher
   - Funding and grants
   - Licensing, Access control and copyright
   - Extra metadata such as Exif
   - Places
   - Subjects & keywords
   - Time
   - Thumbnails
9. Provenance of entities
10. Workflows and scripts

Appendix

Summary: RO-Crate in a nutshell

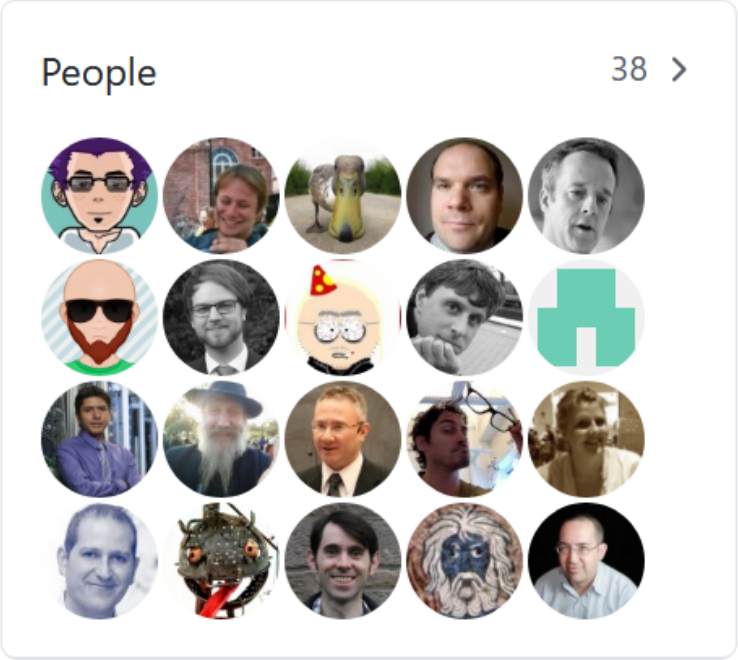
Practical lightweight approach to packaging research **data entities (any object)** with metadata

Aggregate **files** and/or **any URI-addressable content**, with contextual information to aid decisions about re-use: Who What When Where Why How.

Web Native Machine readable. Human readable. Search engine friendly. Familiar.

Extensible and Incremental: add additional metadata; nested and typed by their profile.

Open Community effort



http://www.researchobject.org/ro-crate/

Tooling

- Describo interactive **desktop application** to create, update and export RO-Crates for different profiles. (~ *beta*)
  - CalcyteJS is a **command-line** tool to help create RO-Crates and HTML-readable rendering (~ *beta*)
  - ro-crate - **JavaScript**/NodeJS library for RO-Crate rendering as HTML. (~ *beta*)
  - ro-crate-js - utility to render HTML from RO-Crate (~ *alpha*)
  - ro-crate-ruby **Ruby** library to consume/produce RO-Crates (~ *alpha*)
  - ro-crate-py **Python library** to consume/produce RO-Crates (~ *planning*)

These applications use or expose RO-Crates:

- Workflow Hub imports and exports Workflow RO-Crates
- OCFL-indexer NodeJS application that walks the Oxford Common File Layout on the file system, validate RO-Crate Metadata Files and parse into objects registered in Elasticsearch. (~ *alpha*)
- ONI indexer
- ocfl-tools
- ocfl-viewer
- Research Object Composer is a REST API for gradually building and depositing Research Objects according to a pre-defined profile. (RO-Crate support *alpha*)

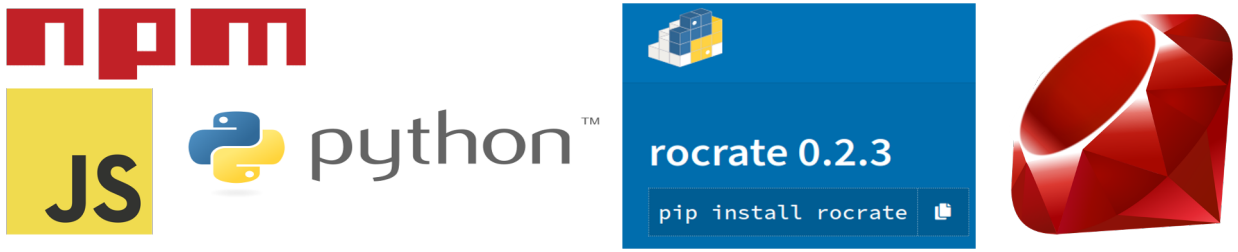
https://www.researchobject.org/ro-crate/implementations.html

User facing

https://uts-eresearch.github.io/describo/

Infrastructure facing

Software libraries



https://www.npmjs.com/package/ro-crate
https://github.com/ResearchObject/ro-crate-ruby
https://pypi.org/project/rocrate/

[Marco La Rosa, Peter Sefton]

PARADISEC

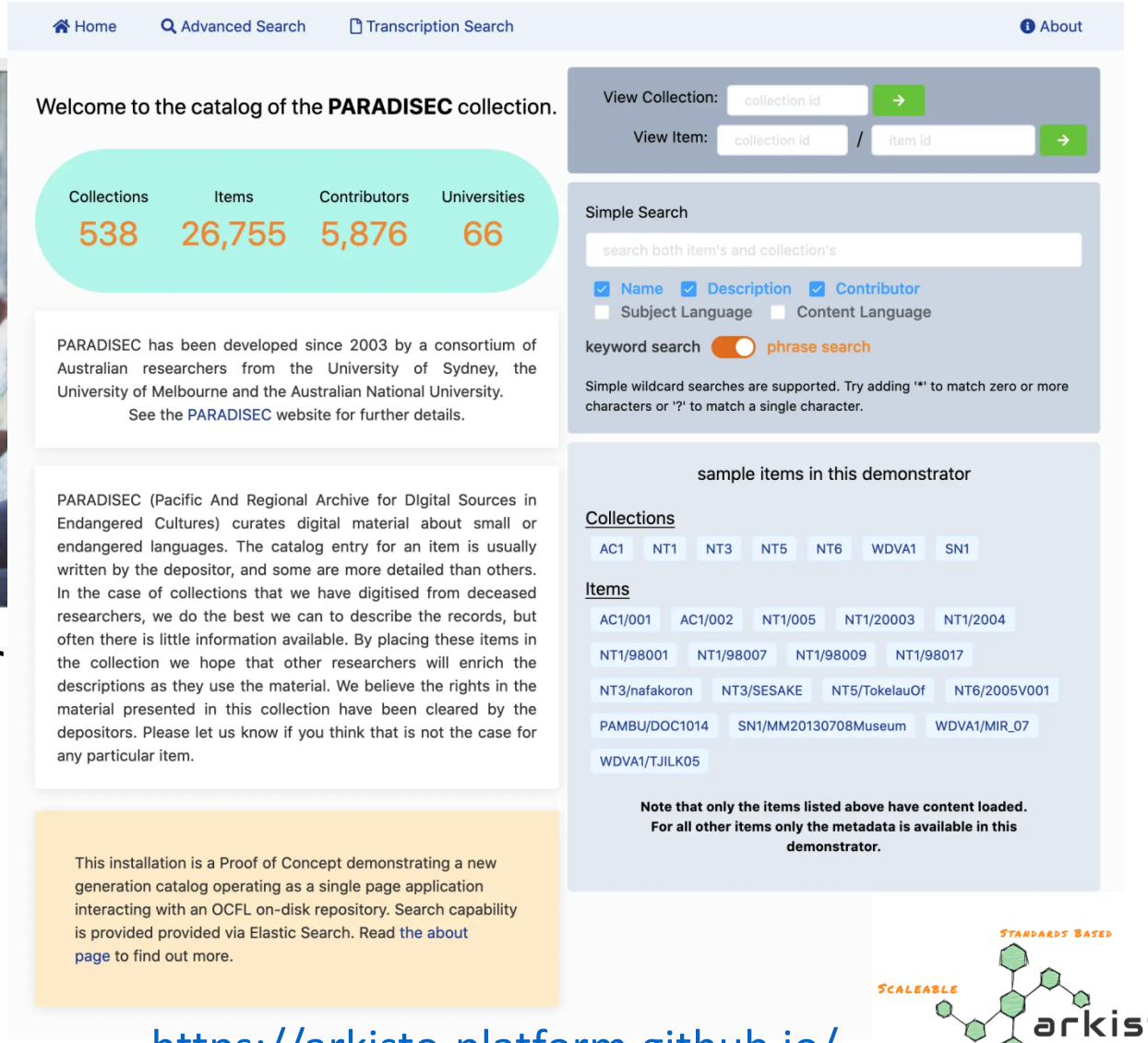
Home About us Our Collections Deposit Resources Support Us Donate

Pacific and Regional Archive for Digital Sources in Endangered Cultures

A race against time to digitise analog records of materials from endangered cultures from all over the world.

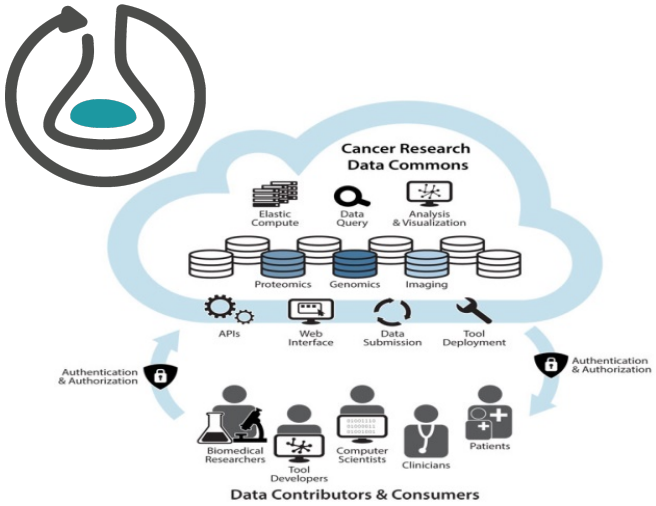
Cultural Heritage: A data curation service for endangered languages: 500,000 files in 28,624 items and 574 collections

long term preservation and accessibility of research data objects



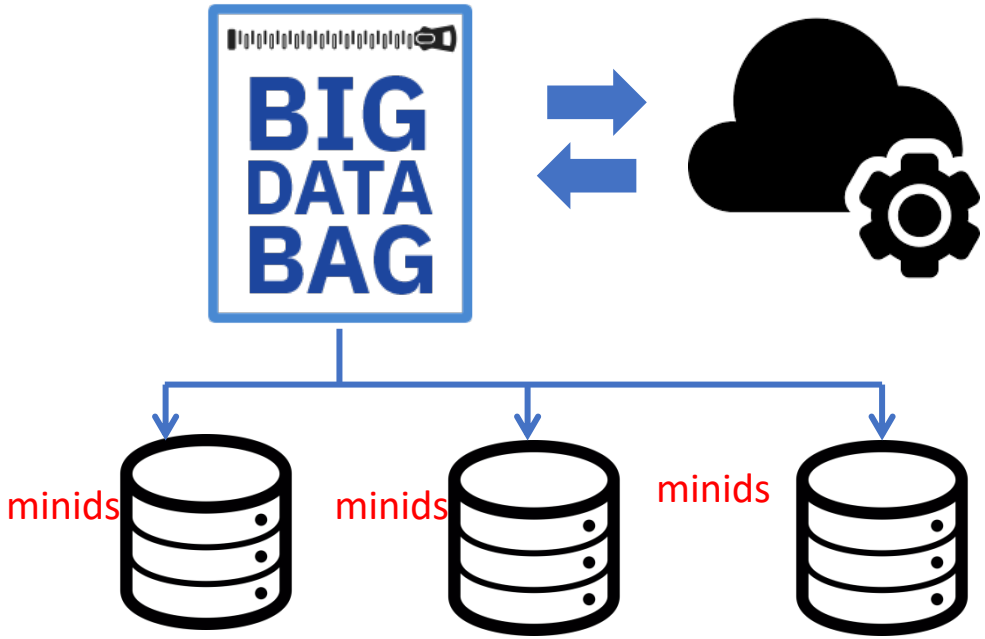
https://arkisto-platform.github.io/

STANDARDS BASED SCALEABLE arkisto SUSTAINABLE



NIH Data Commons

Processing big genomic & clinical data distributed over multiple locations



Scalable verified collections of references

Retain and archive processed datasets
Reference and transfer large data on demand
Controlled access to sensitive data

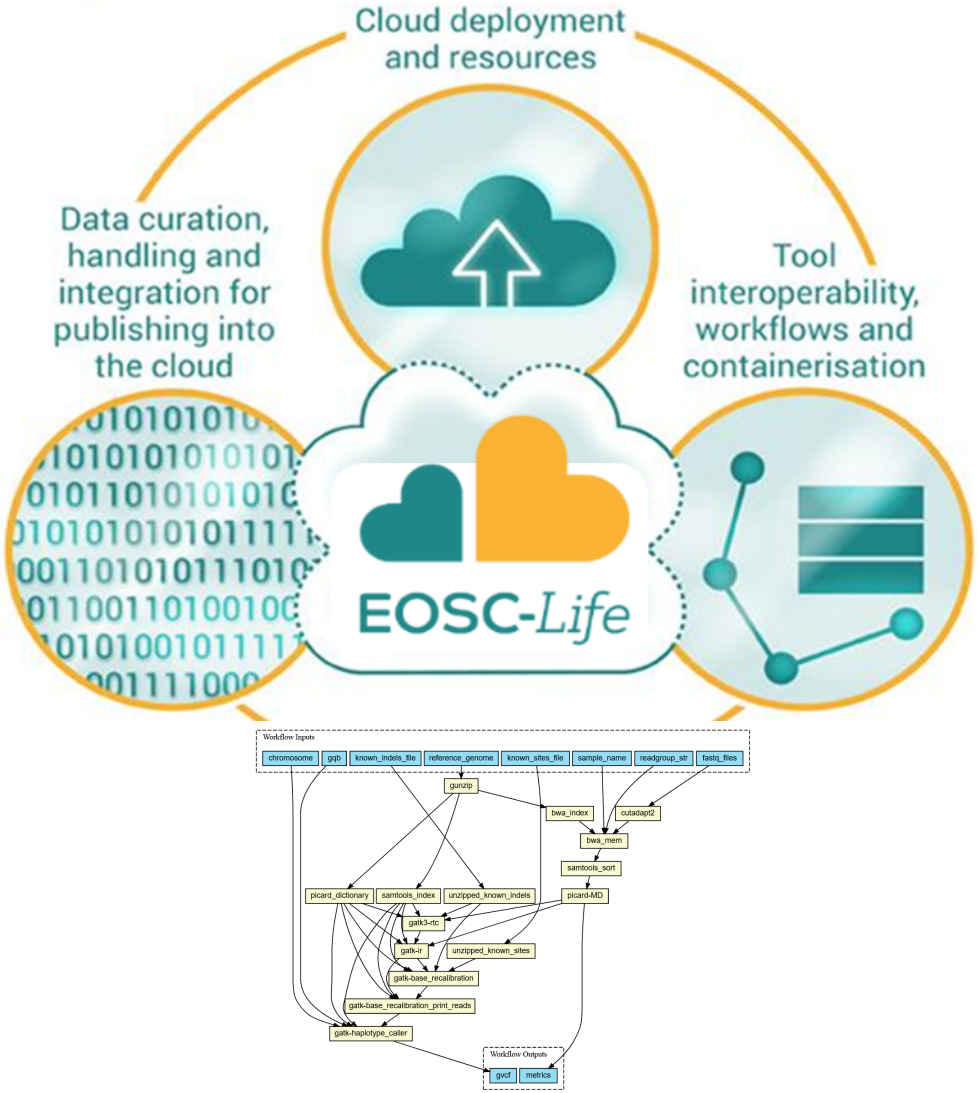
[Kesselman, Foster]

[Chard, et al 2016]

https://doi.org/10.1109/BigData.2016.7840618

Data and Method Thematic Commons

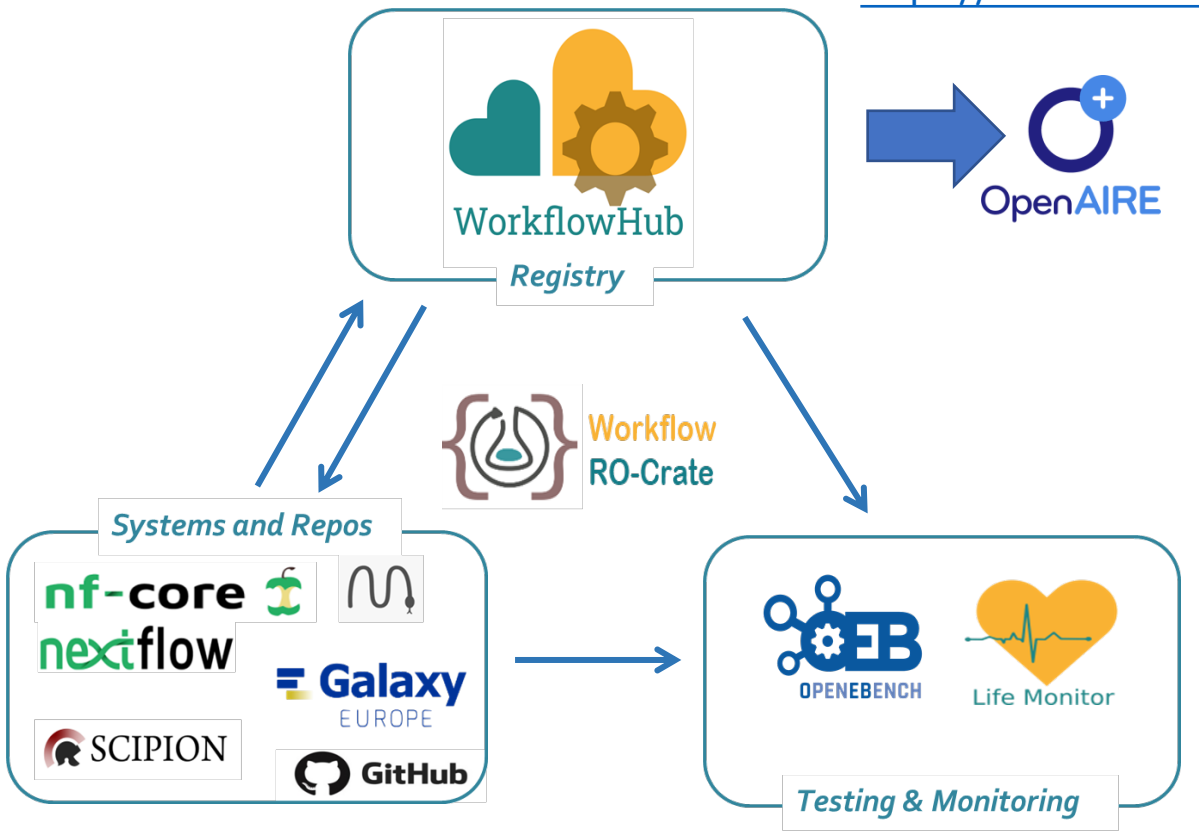
Sharing data, tools and workflows in the cloud



13 EU Life Science Research Infrastructures

EOSC-Life

https://www.eosc-life.eu/



Workflow + data + provenance interchange, stewardship, recording dependencies -> portability & reuse/reproducibility

RO-Crate as interchange format

WorkflowHub

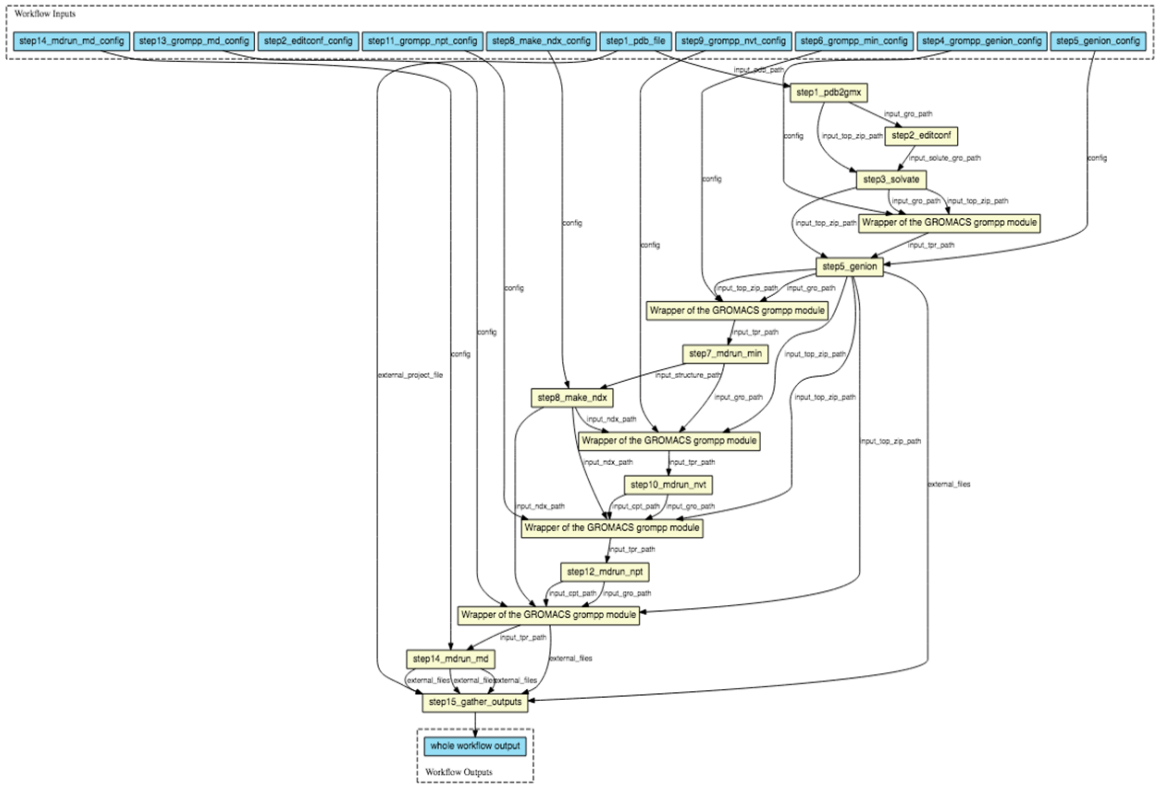
Download RO Crate

Example of setting up a simulation system Version 2

View on GitHub | Request Contact | Unsubscribe | Download RO Crate | Actions

Work-in-progress

CWL version of the md_list.cwl workflow for HPC.



SEEK ID: https://workflowhub.eu/workflows/98?version=2

Creators and Submitter

Creators

Not specified

Submitter

Douglas Lowe

License

Apache Software License 2.0

Activity

Views: 40 **Downloads:** 0

Created: 29th Jan 2021 at 16:37
Last updated: 29th Jan 2021 at 16:56
Last used: 4th Feb 2021 at 08:12

Tags

This item has not yet been tagged.

Add your tags

Attributions

None

- graph.svg
- md_list.cwl
- ro-crate-metadata.json
- ro-crate-preview.html


```
  "license": "Apache-2.0",
  "identifier": "https://workflowhub.eu/workflows/98?version=2",
  "url": "https://workflowhub.eu/workflows/98/ro_crate?version=2"
  "isBasedOn": "https://github.com/douglowe/biobb_hpc_cwl_md_list
  "sdPublisher": {
    "@id": "https://orcid.org/0000-0002-1248-3594"
  },
  "sdDatePublished": "2021-02-04 11:33:41 +0000",
  "creativeWorkStatus": "Work-in-progress"
},
{
  "@id": "md_list.cwl",
  "@type": [
    "File",
    "SoftwareSourceCode",
    "ComputationalWorkflow"
  ],
  "programmingLanguage": {
    "@id": "#cwl"
  },
  "url": "https://github.com/douglowe/biobb_hpc_cwl_md_list/blob/
  "image": {
    "@id": "graph.svg"
  },
  "contentSize": 10444
```


https://workflowhub.eu/workflows/98

WorkflowHub Browse Help Search here... Search

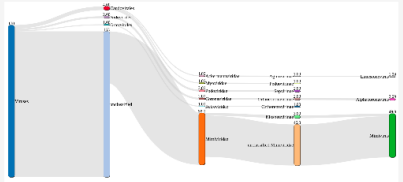
VIRify Version 1

Stable

Download RO Crate

CWL 1.2.0-dev2 | nextflow 20.01.0 | uses docker | uses conda | build passing

VIRify

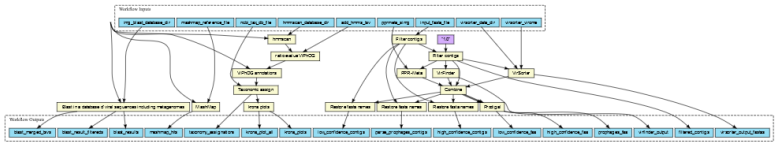


VIRify is a recently developed pipeline for the detection, annotation, and taxonomic classification of viral contigs in metagenomic and metatranscriptomic assemblies. The pipeline is part of the repertoire of analysis services offered by MGnify. VIRify's taxonomic classification relies on the detection of taxon-specific profile hidden Markov models (HMMs), built upon a set of 22,014 orthologous protein domains and referred to as ViPhOGs.

VIRify was implemented in CWL.

What do I need?

The current implementation uses CWL version 1.2 dev+2. It was tested using Toil version 4.10 as the workflow engine and conda to manage the software dependencies.



SEEK ID: https://workflowhub.eu/workflows/26?version=1

Creators

Martin Beracochea, Martín Beracochea, Martin Hölzer, Alexandre Almeida, Guillermo Rangel-Pineros and Ekaterina Sakharova

Submitter

Laura Rodriguez-Navas

License

Apache Software License 2.0

Activity

Views: 585 **Downloads:** 24

Created: 8th Jun 2020 at 11:21
Last updated: 8th Jun 2020 at 13:56
Last used: 27th Nov 2020 at 10:46

Tags

covid-19

Attributions

None

COMMON WORKFLOW LANGUAGE

Inputs

ID	Name	Description	Type
input_fasta_file	*n/a*	*n/a*	File
virsorter_virome	*n/a*	Set this parameter if the input fasta is mostly viral. See: https://github.com/simroux/VirSorter/issues/50	boolean
virsorter_data_dir	*n/a*	VirSorter supporting database files.	Directory
add_hmms_tsv	*n/a*	Additonal metadata tsv	File
hmmscan_database_dir	*n/a*	HMMScan Viral HMM (databases/vpHMM/vpHMM_database). NOTE: it needs to be a full path.	Directory
ncbi_tax_db_file	*n/a*	ete3 NCBITaxa db https://github.com/etetoolkit/ete/blob/master/ete3/ncbi_taxonomy/ncbiquery.py http://etetoolkit.org/docs/latest/tutorial/tutorial_ncbitaxonomy.html This file was manually built and placed in the corresponding path (on databases)	File
img_blast_database_dir	*n/a*	Downloaded from: https://genome.jgi.doe.gov/portal/IMG_VR/IMG_VR.home.html	Directory
mashmap_reference_file	*n/a*	MashMap Reference file. Use MashMap to	File?
pprmeta_simg	*n/a*	PPR-Meta singularity simg file	File

Steps

ID	Name
fasta_rename	Filter contigs
length_filter	Filter contigs
virfinder	VirFinder

Same conceptual workflow; multiple executable flavours for different workflow engines and specific use-cases

https://workflowhub.eu/

Concepts

This section uses terminology from the RO Crate 1.0 specification.

Main Workflow

The *Crate* MUST contain a data entity of type `["File", "SoftwareSourceCode", "Workflow"]` as the *Main Workflow*.

The *Crate* MUST refer to the *Main Workflow* via `mainEntity`.

The *Main Workflow* MUST refer to its type via `pro`

Main Workflow CWL Description

The *Crate* COULD contain a data entity of type `[ "SoftwareSourceCode", "Workflow"]` as th *Description*.

If present the *Main Workflow* MUST refer to the *M Description* via `subjectOf`.

Main Workflow Diagram

The *Crate* COULD contain a *Main Workflow Diagra* entity of type `["File", "ImageObject", "Wo`

If *Main Workflow Diagram* is present, the *Main W* `image`.

Crate

The *Crate* MUST specify a `license`.

The *Crate* SHOULD contain README.md at the r

The *Crate* COULD contain a Dataset (directory) d `["Dataset"]` named "test" to hold tests.

The *Crate* COULD contain a Dataset (directory) d `["Dataset"]` named "examples" to hold examples.

Workflow RO-Crate

Profile

Bioschemas

Home | Getting Started | Specifications | Deploy & Develop | Community | About

ComputationalWorkflow DRAFT Profile

Version: 0.5-DRAFT-2020_07_21 (22 July 2020)

Bioschemas specification for describing a computational workflow in the Life Sciences

If you spot any errors or omissions with this type, please file an issue in our GitHub.

Description | Contributors | Links

Schema.org hierarchy

This Profile fits into the schema.org hierarchy as follows:

Thing > CreativeWork > SoftwareSourceCode > ComputationalWorkflow

Description

A computational workflow consists of an orchestrated and repeatable pattern of activity enabled by the systematic organization of resources into processes that transform materials, provide services, or process information (source Wikipedia.org). It is executed by a computational process and is thus distinct from laboratory or business workflows.

Latest profiles

Latest release: none

Latest draft: 0.5-DRAFT-2020_07_21

Bioschemas.org

EOSC-Life

elixir

Embedding into Infrastructures & Standards

EGI-ACE data spaces for
Earth Science researchers

Reliance

Research Lifecycle Management technologies for Earth Science Communities and Copernicus users in EOSC

ROHUB

zenodo

OpenAIRE

CS³MESH⁴EOSC
Connecting European Data

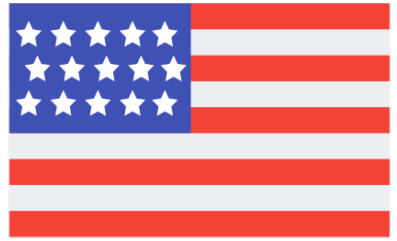
FAIR and GDPR compliant data storage and sharing fabric Science Mesh using Cloud Services for Synchronization and Sharing

STANDARDS BASED
SCALEABLE
arkisto
SUSTAINABLE

MENDELEY DATA

Seven Bridges

exchange between genomics platform and repository.



BioCompute Objects

FDA

standardise and share analyses generated from genome sequencing.

IEEE 2791-2020

Supporting the Research Life Cycle

Release the objects as they are created, updated and used

- ✓ Exchange & Import/Export
- ✓ Report & Archive
- ✓ Share & Access
- ✓ Reuse & Reproduce
- ✓ Living Objects

Describe the objects as they are being created using open standards and tools

Figure: RDMKit, https://rdmkit.elixir-europe.org/

Science 2.0 Repositories: Time for a Change in Scholarly Communication Assante, Candela, Castelli, Manghi, Pagano, D-Lib 2015

Machine-processable

Standards

EXAMPLES

LOW TECH

Incremental

MULTI-PLATFORM

Graceful degradation

Commodity tooling

Technology Independent

Keep it simple

Developer friendliness

Just enough complexity / standards

- sufficient extra benefits from what already exists…without compromising developer entry-level experience so they do their own thing

Just Enough Linked Data Just In Time

- simplifications instead of generalizations

Retain Linked Data benefits

- querying, graph stores, vocabularies, clickable URIs)

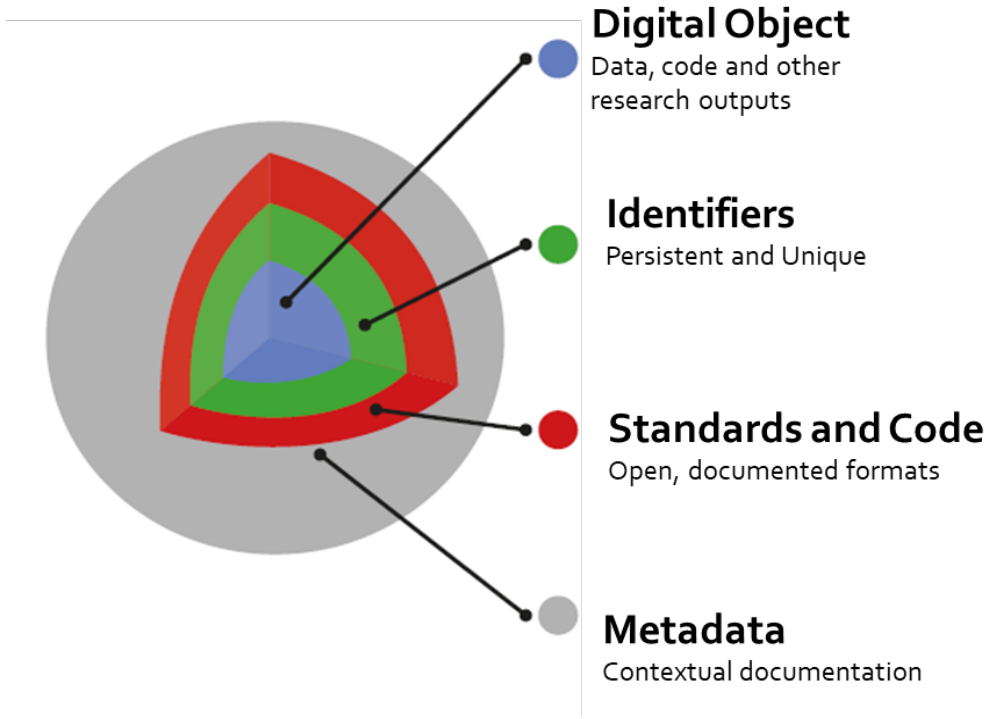
Plus the developer needs

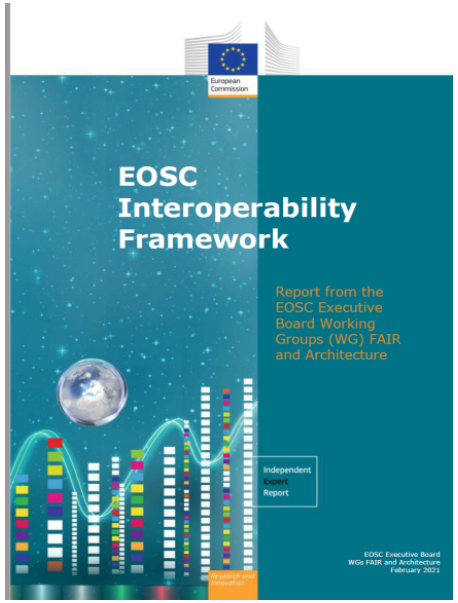
- documentation, examples, libraries, tools, community …

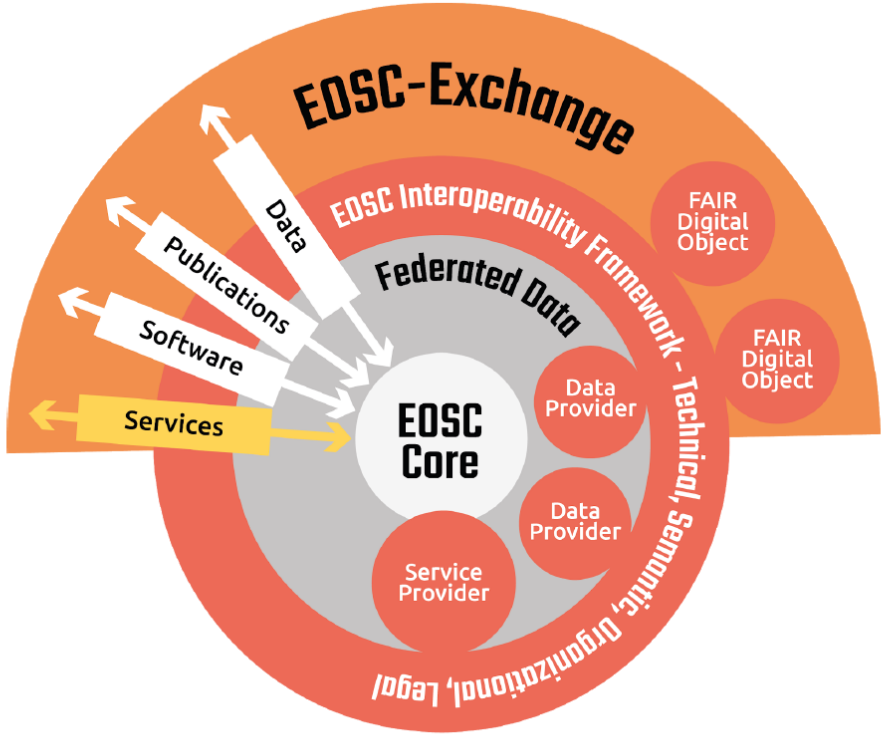
Limited flexibility frees up developers

Familiarity is important for uptake

From FAIR data to FAIR Digital Objects







GEDE RDA RESEARCH DATA ALLIANCE
Group of European Data Experts in RDA

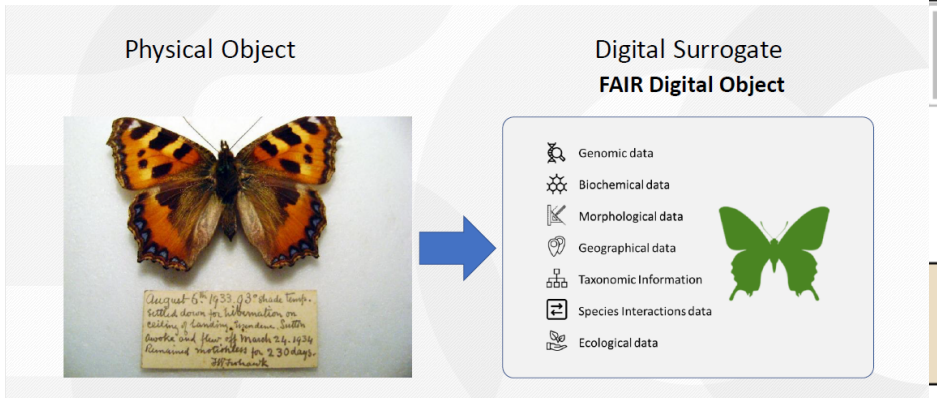
EOSCFAIR
Executive Board Working Group

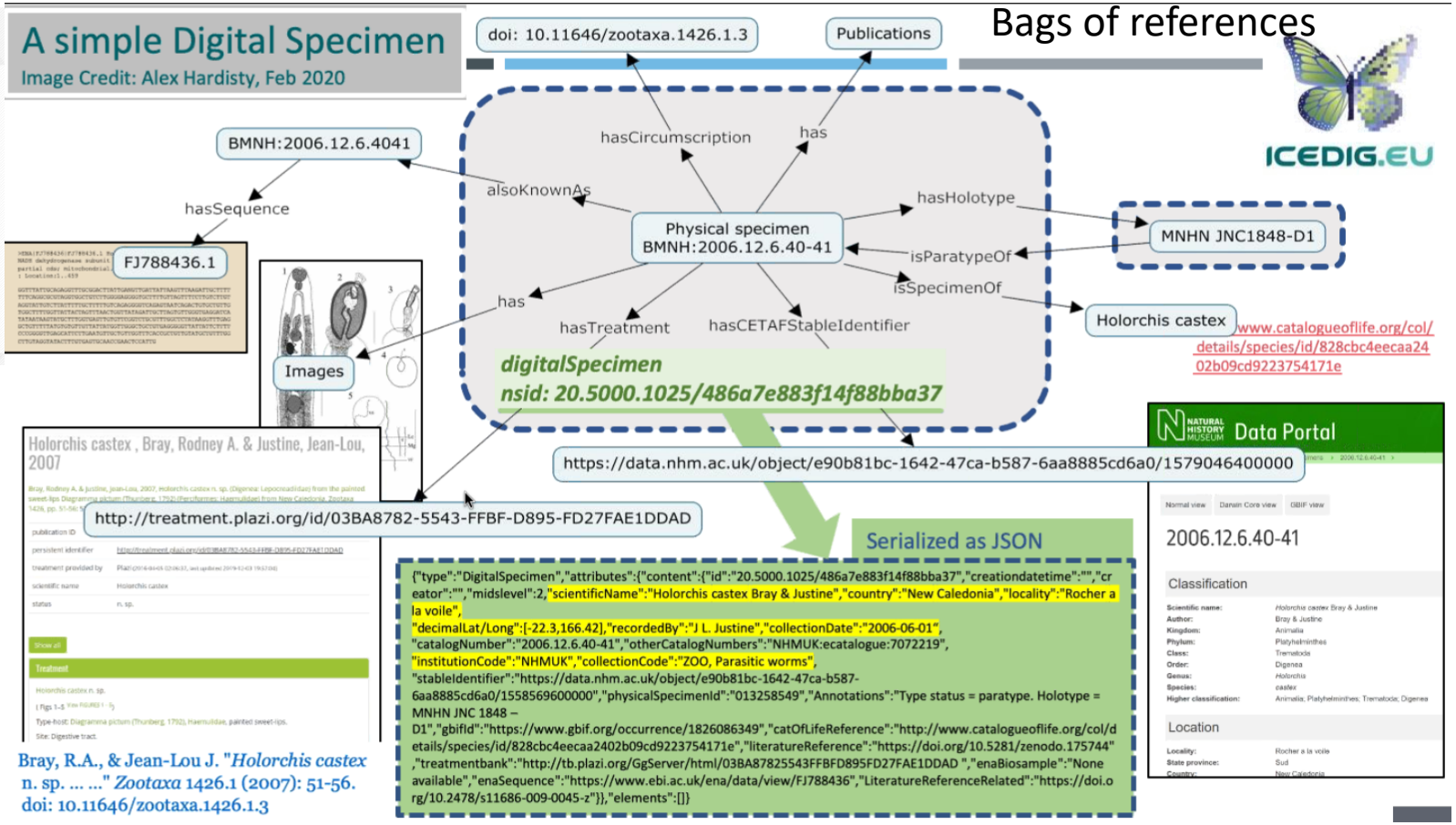
FAIR Digital Objects for Science: From Data Pieces to Actionable Knowledge Units: https://doi.org/10.3390/publications8020021

https://op.europa.eu/en/publication-detail/-/publication/d787ea54-6a87-11eb-aeb5-01aa75ed71a1/language-en/format-PDF/source-190308283

FAIR Digital Objects

Actionable knowledge unit
Digital butterfly – digital twins





courtesy Dimitris Koureas
Coordinator DiSSCo EU Research Infrastructure

Specimen object image courtesy of Alex Hardisty

[Hardisty et al, 2020]

Specimen Data Refinery

Workflows to Digitise Natural History Specimens

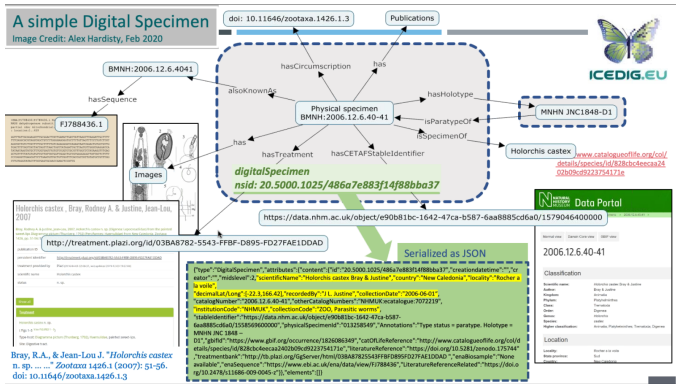
FAIR Digital Objects -> Packaged + Actionable

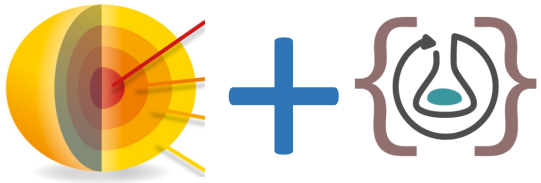
SYNTHESYS+ Synthesis of Systematic Resources a DiSSCo project

DiSSCo Distributed System of Scientific Collections

FAIR Digital Object Framework

Open Digital Specimen





+

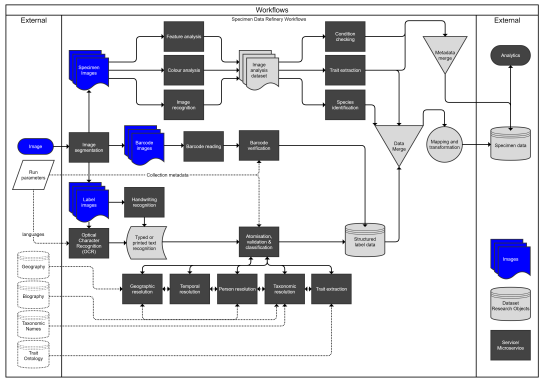
EOSC-Life

researchobject

elixir

Workflow Infrastructure

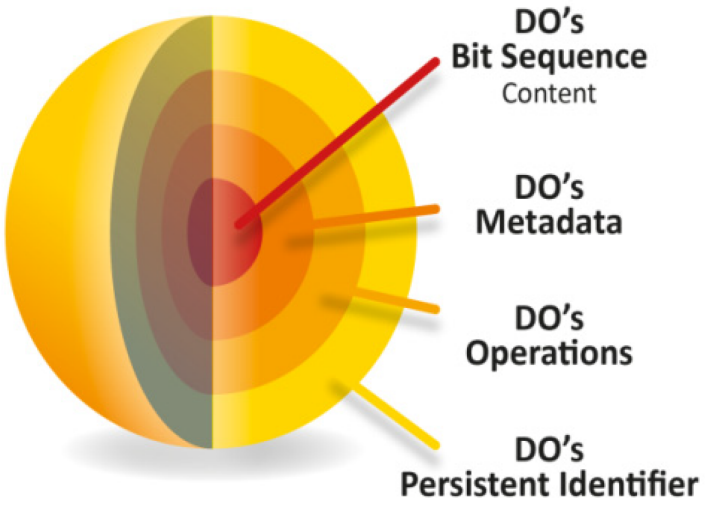
RO-Crate

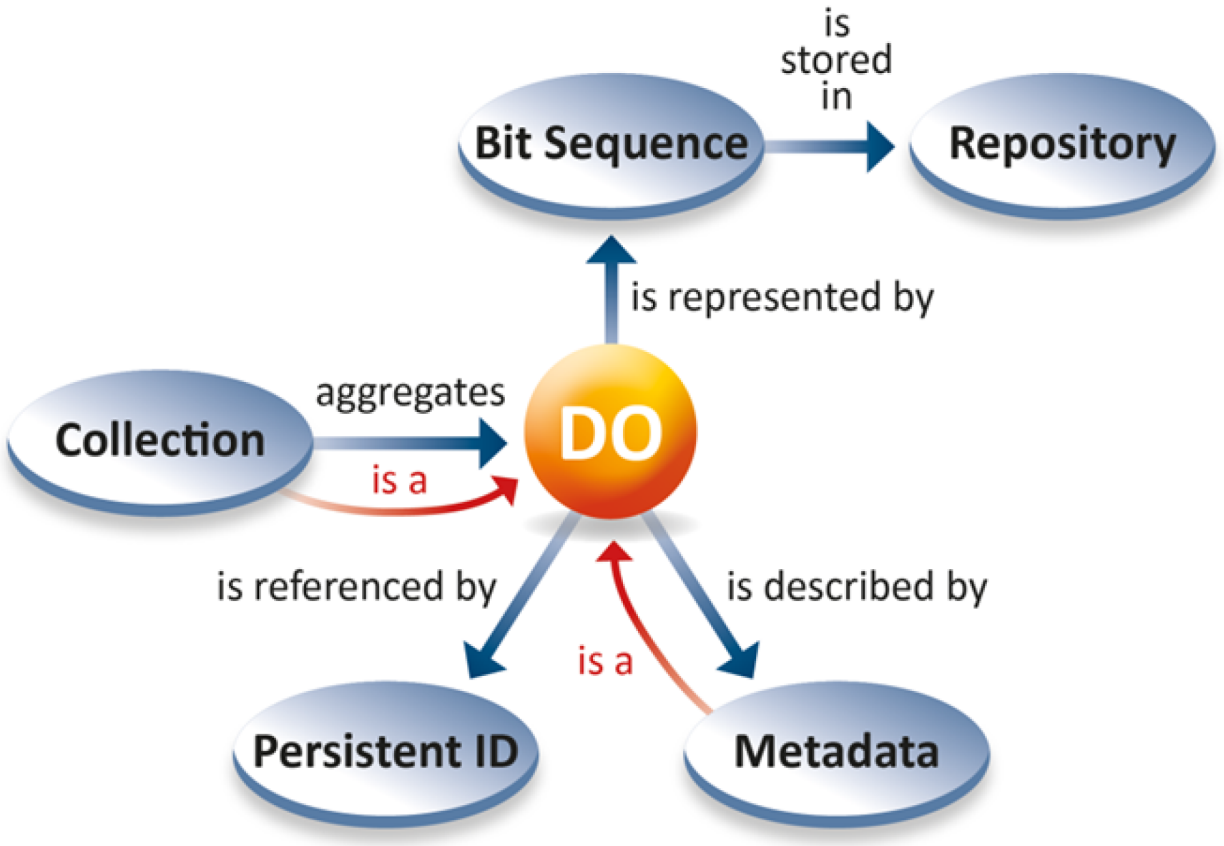


Same conceptual workflow; multiple executable flavours

Workflow RO-Crate

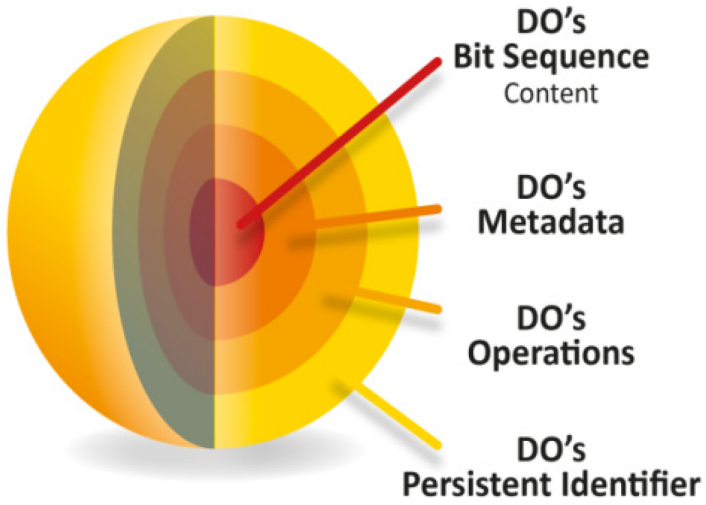
FAIR Digital Object

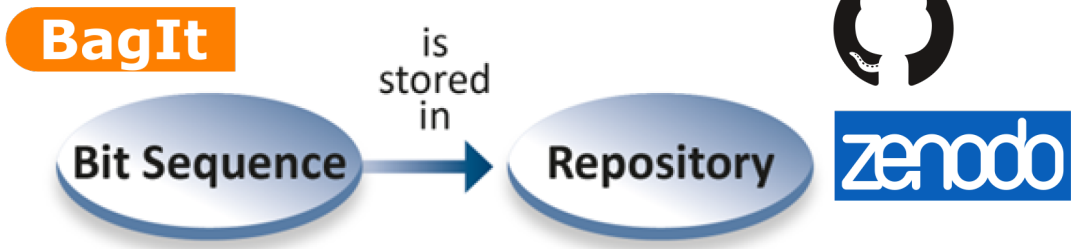




Adapted from
https://doi.org/10.3390/publications8020021

RO-Crate as FAIR Digital Object





is represented by

RO-Crate

http://schema.org/Dataset

Data Entity
Contextual Entity

is referenced by

is described by

is a

https://w3id.org/

RO-Crate
metadata file

Adapted from
https://doi.org/10.3390/publications8020021

RO-Crate model as UML

https://www.researchobject.org/ro-crate/1.1/structure.html

«schema.org/Dataset»

RO-Crate Root Data Entity

«schema.org/CreativeWork»

RO-Crate Metadata File

schema.org/about

```
"conformsTo": {"@id": "https://w3id.org/ro/crate/1.1"},
```

A valid *RO-Crate JSON-LD* graph MUST describe:
1. The RO-Crate Metadata File Descriptor
2. The Root Data Entity
3. Zero or more Data Entities
4. Zero or more Contextual Entities

schema.org/hasPart

schema.org/mentions

(is-a)

«schema.org/CreativeWork»

Data Entities

(describes)

«schema.org/Thing»

Contextual Entities

(describes)

«schema.org/Place»

subClassOf

subClassOf

subClassOf

«schema.org/Dataset»

«schema.org/MediaObject»

«schema.org/ImageObject»

«bioschemas.org/ComputationalWorkflow»

«schema.org/Person»

«schema.org/Organization»

«schema.org/IndividualProduct»

```
{ "@context": "https://w3id.org/ro/crate/1.1/context",
  "@graph": [
```

JSON-LD preamble

```
    { "@type": "CreativeWork",
      "@id": "ro-crate-metadata.json",
      "conformsTo": {"@id": "https://w3id.org/ro/crate/1.1"},
      "about": { "@id": "./" }
    }
```

RO-Crate **metadata file** descriptor

Collection

```
    { "@id": "./",
      "identifier": "https://doi.org/10.5281/zenodo.1009240",
      "@type": "Dataset",

      "hasPart": [
        { "@id": "cp7glop.ai" },
        { "@id": "lots_of_little_files/" },
        { "@id": "communities-2018.csv" },
        { "@id": "https://doi.org/10.4225/59/59672c09f4a4b" },
        { "@id": "SciDataCon-Presentations/AAA_Pilot_Abstract.html"}
       ],

      "author": { "@id": "https://orcid.org/0000-0002-8367-6908" },
      "publisher": { "@id": "https://ror.org/03f0f6041" },
      "citation": { "@id": "https://doi.org/10.1109/TCYB.2014.2386282"},
      "name": "Presentation of user survey 2018"
    },
```

RO-Crate **root** dataset

..aggregates **Data entities**

..described w/ **contextual entities**

```
    { "@id": "cp7glop.ai",
      "@type": "File",
      "name": "Diagram showing trend to increase",
      …
    },
    …
```

Flat list of metadata per entity

Metadata

```
{
 "@id": "figure.png",
 "@type": ["File", "ImageObject"],
 "name": "XXL-CT-scan of an XXL Tyrannosaurus rex skull",
 "identifier": "https://doi.org/10.5281/zenodo.3479743",
 "author": {"@id": "https://orcid.org/0000-0002-8367-6908"},
 "encodingFormat": "image/png"
}
```

```
{
 "@id": "https://orcid.org/0000-0002-8367-6908",
 "@type": "Person",
 "affiliation": { "@id": "https://ror.org/03f0f6041" },
 "name": "J. Xuan"
}
```

```
{
 "@id": "https://ror.org/03f0f6041",
 "@type": "Organization",
 "name": "University of Technology Sydney",
 "url": "https://www.uts.edu.au/"
}
```

Data and **Contextual** entities
described *within* RO-Crate Metadata File

Base vocabulary & types: **schema.org**

Cross-references to further contextual entities

RO-Crate **principle**:
Reuse existing PIDs and URLs

..but always **describe entities** which lack a human-readable resolution

Persistent IDs
Gradual ascent towards FAIR

Base line: **Relative paths** from *RO-Crate Metadata File*

Use cases: Describing dataset on **desktop**
Ad-hoc **web-hosting** (e.g. GitHub pages)

Institutional **archives** (e.g. Oxford Common File Layout)

Reuse **existing PIDs** and URLs

Use cases: Large data, not a file (e.g. database), reference datasets

Cite/reference existing resources (e.g. via identifiers.org)

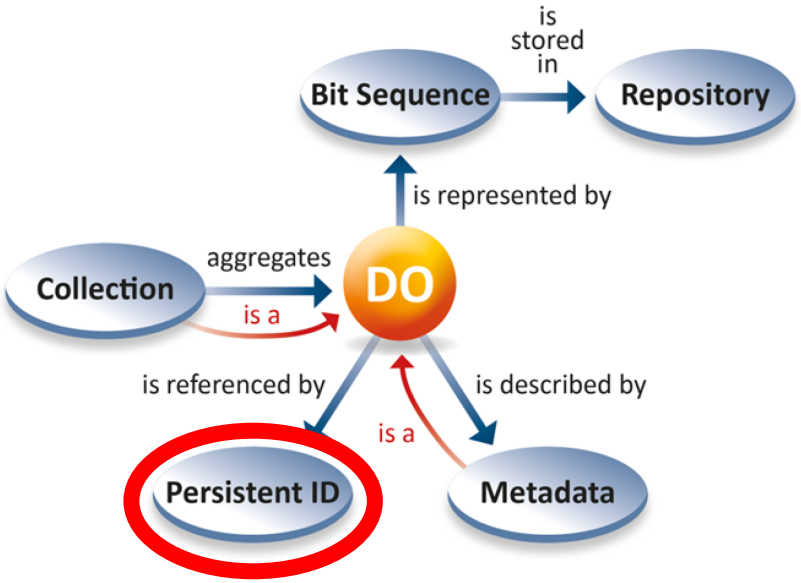
Distinguish and crosslink contextual entities

Make paths absolute, using **location-independent PIDs**
(UUIDs, *Naming Things with Hashes*, ARCP)

Use cases: Found RO-Crate "in the wild", ZIP archives, workflow outputs

Assign PIDs to RO-Crate (and its entities)

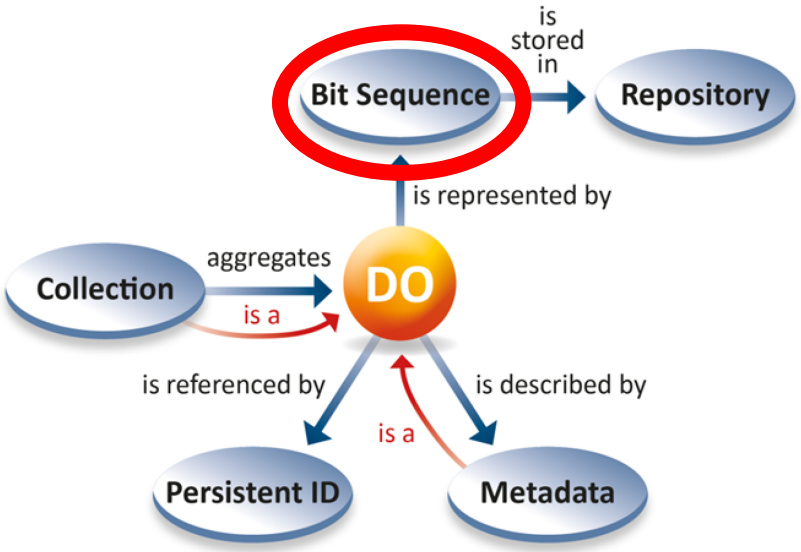
Use cases: Long-term availability, citations, permalinks



Bit Sequences

Base line: Files on disk within **RO-Crate Root** folder + **URLs**

Downloadable **Web resources** listed w/ *content size* and *access time*.



Packaging/**archiving RO-Crate Root** folder:

- BagIt (RFC8493): **Manifest** of all (local) files and their checksums
  Use case: Ensure all files are transferred/archived
- BDBag: Include external files by ARK/MinID PIDs
  Use case: "thin" RO-Crate, **Big Data**, shared immutable files
- OCFL: **Archival** storage with **revision** tracking, infrequent changes
- Git LFS: Tracked **frequent changes**, collaborative editing

Repository

Base line: zip/tar download

- e.g. “supplementary data”, GitHub release

Export from web platforms

- e.g. workflowhub.eu, Galaxy workflow system

Deposit in **general/institutional repositories**

- e.g. Zenodo, Mendeley Data

Deposit in **domain-specific** repositories

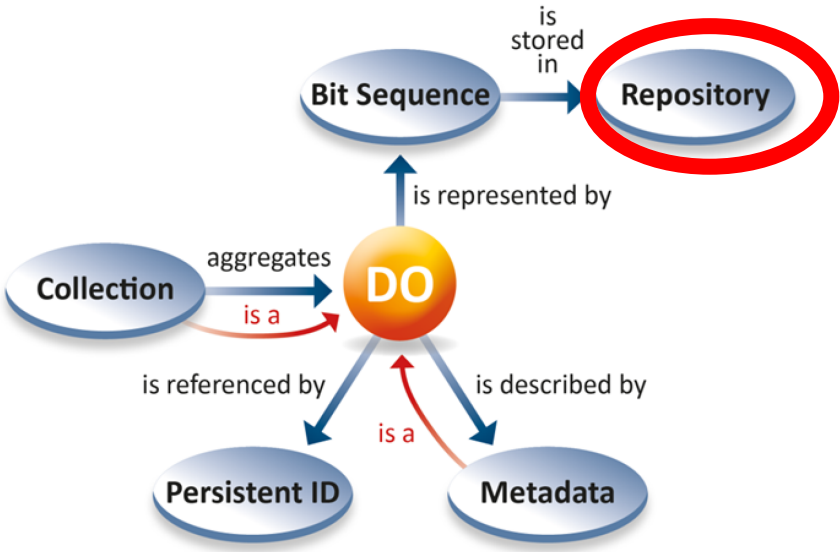
- e.g. GBIF

Deposit **individual** files

- e.g. ARK, S3, B2SHARE

Deposit **metadata**

- e.g. WikiData, nanopubs



What can FDO learn from RO-Crate?

Data and **Contextual** entities are equally important

Use wheels already invented

- Fit into researchers' existing working practices and familiar technologies.
  - Aim for gradual improvements.
- Reuse existing technology
  - .. But only when not too complex
- Reuse existing PID infrastructure, including URLs
  - Human consumers recognize and click hyperlinks
- Build on existing metadata standards
  - Which is both simple and extendable

Remember people … especially Developers

- Keep human consumption in focus
  - Ensure metadata is easily rendered and edited
- Provide Best Practice guidance
  - Firm, but not too restrictive
- Developer producer and consumer friendly
  - Rather than academically elegant

Not everything is known before hand

- Existing PID metadata not always relevant
  - Allow contextualized metadata
- Types not always known in advance
  - Allow "casting" or reinterpretation
- Operations not always known in advance
  - Allow *open-ended* generation and use

What can RO-Crate learn from FDO?

Provide stronger guidance on **PID** and availability

- Recommend deposit infrastructure for end-users and framework developers
- Tools to assist – e.g. generate Zenodo Datacite metadata from RO-Crate

Provide stronger **typing** of RO-Crates and its content

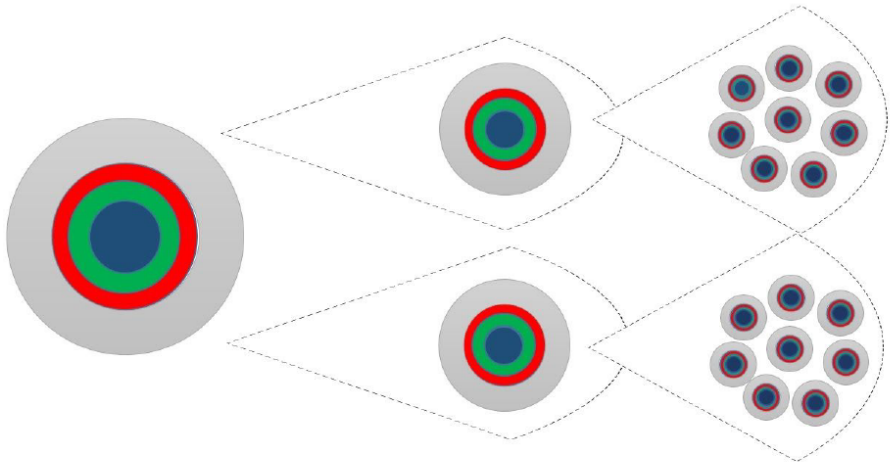
- Profiles as first approach

Expose potential **operations** on an RO-Crate

- Build general RO-Crate services, e.g. index

Turtles all the way down!

- Document better how RO-Crates can/should be nested
- How to choose granularity of RO-Crates?
- Tools to liberate/reuse an entity from a single RO-Crate

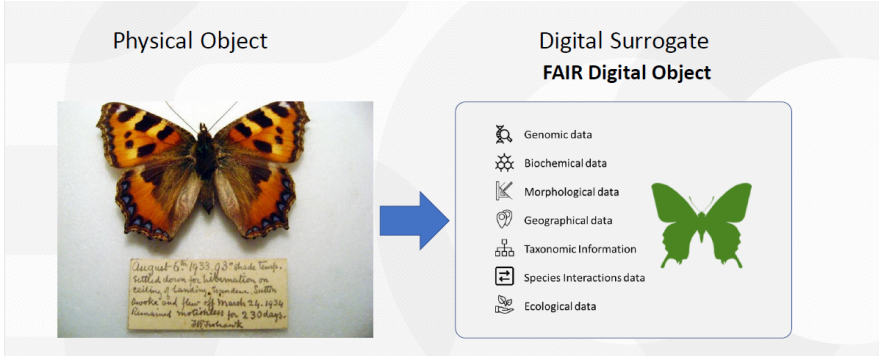


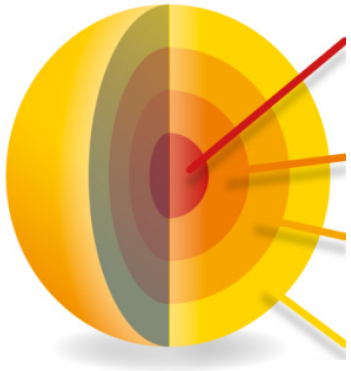
RO-Crate as part of the FDO ecosystem

RO-Crate

A bag of references + metadata

A metadata framework for FDO
Developer friendly practical and web-native implementations
Community
Infrastructure applications





A bag of references + metadata

Framework for actionable objects

RO-Crates enable active operations for FDOs
FDO offers additional infrastructure and practice

Synthesys+ Project gives a concrete use case for practically join the two up.

SYNTHESYS+ Synthesis of Systematic Resources a DiSSCo project

DiSSCo Distributed System of Scientific Collections

elixir

RO-Crate Community

https://www.researchobject.org/ro-crate/

Team

RO-Crate

The *RO-Crate* team is:

- Peter Sefton https://orcid.org/0000-0002-3545-944X (co-chair)
- Stian Soiland-Reyes https://orcid.org/0000-0001-9842-9718 (co-chair)
- Eoghan Ó Carragáin https://orcid.org/0000-0001-8131-2150 (emeritus ch
- Oscar Corcho https://orcid.org/0000-0002-9260-0753
- Daniel Garijo https://orcid.org/0000-0003-0454-7145
- Raul Palma https://orcid.org/0000-0003-4289-4922
- Frederik Coppens https://orcid.org/0000-0001-6565-5145
- Carole Goble https://orcid.org/0000-0003-1219-2137
- José María Fernández https://orcid.org/0000-0002-4806-5140
- Kyle Chard https://orcid.org/0000-0002-7370-4805
- Jose Manuel Gomez-Perez https://orcid.org/0000-0002-5491-6431
- Michael R Crusoe https://orcid.org/0000-0002-2961-9670
- Ignacio Eguinoa https://orcid.org/0000-0002-6190-122X
- Nick Juty https://orcid.org/0000-0002-2036-8350
- Kristi Holmes https://orcid.org/0000-0001-8420-5254
- Jason A. Clark https://orcid.org/0000-0002-3588-6257
- Salvador Capella-Gutierrez https://orcid.org/0000-0002-0309-604X
- Alasdair J. G. Gray https://orcid.org/0000-0002-5711-4872
- Stuart Owen https://orcid.org/0000-0003-2130-0865
- Alan R Williams https://orcid.org/0000-0003-3156-2105
- Giacomo Tartari https://orcid.org/0000-0003-1130-2154
- Finn Bacall https://orcid.org/0000-0002-0048-3300
- Thomas Thelen https://orcid.org/0000-0002-1756-2128
- Hervé Ménager https://orcid.org/0000-0002-7552-1009
- Laura Rodríguez Navas https://orcid.org/0000-0003-4929-1219
- Paul Walk https://orcid.org/0000-0003-1541-5631
- brandon whitehead https://orcid.org/0000-0002-0337-8610
- Mark Wilkinson https://orcid.org/0000-0001-6960-357X
- Paul Groth https://orcid.org/0000-0003-0183-6910
- Erich Bremer https://orcid.org/0000-0003-0223-1059
- LJ Garcia Castro https://orcid.org/0000-0003-3986-0510
- Karl Sebby https://orcid.org/0000-0001-6022-9825
- Alexander Kanitz https://orcid.org/0000-0002-3468-0652
- Ana Trisovic https://orcid.org/0000-0003-1991-0533
- Gavin Kennedy https://orcid.org/0000-0003-3910-0474
- Mark Graves https://orcid.org/0000-0003-3486-8193
- Jasper Koehorst https://orcid.org/0000-0001-8172-8981
- Simone Leo https://orcid.org/0000-0001-8271-5429

Join us!

Bi-weekly calls & GitHub

WorkflowHub Club

The weekly WorkflowHub Club is chaired by **Frederik Coppens**.

Participants include:

- Alan R Williams (The University of Manchester)
- Alexander Vasilenko (VKM IBPM RAS, MIRRI)
- Alexander Kanitz
- Alban Gaignard
- Ambarish Kumar (Jawaharlal Nehru University, New Delhi, India)
- Antonio Rosato
- Bert Droesbeke (ELIXIR-BE, VIB-UGent Center for Plant Systems Biology)
- Björn Grüning (University of Freiburg, ELIXIR-DE, Galaxy Project)
- Carole Goble (The University of Manchester, ELIXIR-UK)
- Carlos Oscar Sorzano (CNB CSIC)
- Castrense Savojardo (ELIXIR-IT)
- Dan Fornika
- Decruw Cedric
- Djura Smits (Netherlands eScience Center)
- Emidio Capriotti (UNIBO, Italy)
- Emmy Tsang
- Finn Bacall (The University of Manchester, ELIXIR-UK)
- Flora D'Anna
- Frédéric Lemoine (Institut Pasteur, Paris)
- Frederik Coppens (ELIXIR-BE, VIB-UGent Center for Plant Systems Biology, EOSC-Life)
- Georg Peiter
- Giacomo Tartari (ELIXIR-IT)
- Hervé Ménager (Institut Pasteur, ELIXIR-FR, bio.tools)
- Ignacio Eguinoa (ELIXIR-BE, VIB-UGent Center for Plant Systems Biology)
- Jennifer Harrow (ELIXIR-Hub, tools platform coordinator)
- Jon Ison (Institut Pasteur, ELIXIR-FR, bio.tools)
- José Mª Fernández (BSC, ELIXIR-ES)
- Kiran K Telukunta
- Lars Ridder (Netherlands eScience Center)
- Laura del Caño (CNB-CSIC, INSTRUCT, EOSC-Life)
- Laura Rodriguez-Navas (BSC, ELIXIR-ES)
- Leyla Garcia
- Luca Pireddu (CRS4/BBMRI)

https://about.workflowhub.eu/

Bi-weekly calls & Slack

- Magnus Palmblad (LUMC, ELIXIR-NL)
- Marco Tangaro (ELIXIR-IT)
- Michael R. Crusoe (ELIXIR-NL, CWL, standards developer, interop cheerleader)
- Miguel Vazquez (BSC)
- Miriam Payá Milans (CBGP, Madrid)
- Munazah Andrabi
- Nick Juty (The University of Manchester, ELIXIR-UK)
- Paolo Romano
- Pier Luigi Martelli (ELIXIR-IT)
- Philipp G.
- Rena Bakhshi (Netherlands eScience Center)
- Rob Hooft
- Robin Richardson (Netherlands eScience Center)
- Romain Dallet (EMBRC)
- Salvador Capella-Gutierrez (BSC, ELIXIR-ES)
- Simone Leo (CRS4/BBMRI)
- Sirarat Sarntivijai (ELIXIR-Hub)
- Stian Soiland-Reyes (The University of Manchester, ELIXIR-UK, BioExcel, CWL)
- Stuart Owen (The University of Manchester, FAIRDOM, ELIXIR-UK)
- Vahid Kiani
- Vincenzo Laveglia (CIRMMP Florence, EOSC-Life)
- Wolfgang Müller (HITS GmbH)
- Xiaoming Hu (HITS GmbH)

Acknowledgements

- RO-Crate Community
  https://www.researchobject.org/ro-crate/community
- Workflow Hub Club
  https://about.workflowhub.eu/acknowledgements/

- Funding:
  - H2020-INFRAEOSC-2018-2 824087
  - H2020-INFRAEDI-2018-1 823830
  - H2020-INFRAIA-2017-1 730976
  - H2020-INFRADEV-2019-2 871118
  - H2020-INFRAIA-2018-1 823827

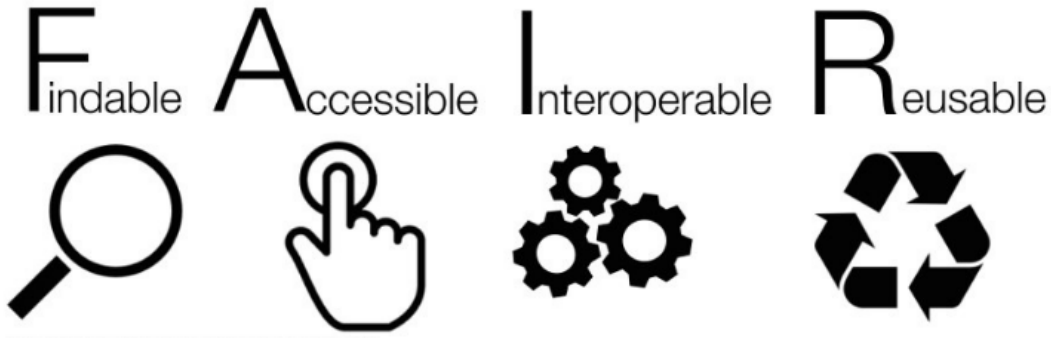
FAIR Principles for Digital Research Objects

FAIR **all the way down**

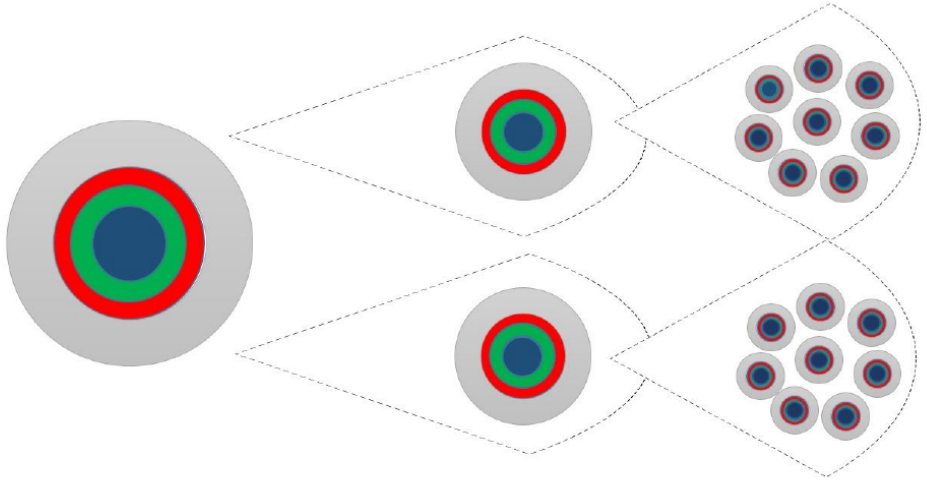
Unbounded FAIR

Distributed FAIR

Living FAIR



Analogous to FAIR Software



FAIR RO-Crate is a practical start

Fig from EOSC Interoperability Framework