The University of Manchester

University of Amsterdam Informatics Institute

researchobject.org

Sharing research artefacts as FAIR Digital Objects using RO-Crate

Carole Goble

The University of Manchester, UK

ELIXIR-UK Head of Node

carole.goble@manchester.ac.uk

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

Stian Soiland-Reyes

The University of Manchester & University of Amsterdam

RO-Crate co-lead

soiland-reyes@manchester.ac.uk

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

https://doi.org/10.5281/zenodo.7559339
derived from https://doi.org/10.5281/zenodo.7147703

Brookhaven National Laboratory
2023-01-23

Annual reminder on FAIR principles

scientific **data**

Explore our content **Journal information**

nature > scientific data > comment > article

Open Access | Published: 15 March 2016

The FAIR Guiding Principles for scientific data management and stewardship

Mark D. Wilkinson, Michel Dumontier, [...] Barend Mons

Scientific Data **3**, Article number: 160018 (2016) | Cite this article

137k Accesses | **1739** Citations | **1586** Altmetric | Metrics

https://doi.org/10.1038/sdata.2016.18

"FAIR Tulip" by Meznah Aloqalaa

Annual reminder on FAIR principles

https://doi.org/10.1038/sdata.2016.18

Findable

F1. (meta)data are assigned a globally unique and persistent **identifier**

F2. data are described with rich **metadata** (defined by R1 below)

F3. metadata clearly and explicitly **include the identifier** of the data it describes

F4. (meta)data are **registered** or **indexed** in a searchable resource

Interoperable

I1. (meta)data use a **formal**, accessible, shared, and broadly applicable **language for knowledge representation**.

I2. (meta)data use **vocabularies** that follow FAIR principles

I3. (meta)data include qualified **references** to other (meta)data

Accessible

A1. (meta)data are **retrievable** by their identifier using a standardized communications protocol

A1.1 the protocol is **open**, free, and universally implementable

A1.2 the protocol allows for an **authentication** and **authorization** procedure, *where necessary*

A2. **metadata** are accessible, even when the *data are no longer available*

Reusable

R1. meta(data) are richly described with a plurality of accurate and relevant **attributes**

R1.1. (meta)data are released with a clear and accessible **data usage license**

R1.2. (meta)data are associated with detailed **provenance**

R1.3. (meta)data meet domain-relevant community **standards**

tl;dr: **machine-readable metadata**

ARTICLE

OPEN

https://doi.org/10.1038/s41586-019-0965-1

A new genomic blueprint of the human gut microbiota

Alexandre Almeida[1,2*], Alex L. Mitchell[1], Miguel Boland[1], Samuel C. Forster[2,3,4], Gregory B. Gloor[5], Aleksandra Tarkowska[1], Trevor D. Lawley[2] & Robert D. Finn[1*]

The composition of the human gut microbiota is linked to health and disease, but knowledge of individual microbial species is needed to decipher their biological roles. Despite extensive culturing and sequencing efforts, the complete bacterial repertoire of the human gut microbiota remains undefined. Here we identify 1,952 uncultured candidate bacterial species by reconstructing 92,143 metagenome-assembled genomes from 11,850 human gut microbiomes. These uncultured genomes substantially expand the known species repertoire of the collective human gut microbiota, with a 281% increase in phylogenetic diversity. Although the newly identified species are less prevalent in well-studied populations compared to reference isolate genomes, they improve classification of understudied African and South American samples by more than 200%. These candidate species encode hundreds of newly identified biosynthetic gene clusters and possess a distinctive functional capacity that might explain their elusive nature. Our work expands the known diversity of uncultured gut bacteria, which provides unprecedented resolution for taxonomic and functional characterization of the intestinal microbiota.

For the past decade, studies of the human gut microbiota have shown that the interplay between microbes and host is associated with various phenotypes of medical importance[1,2]. Shotgun metagenomic analysis methods can infer both taxonomic and functional information from complex microbial communities, guiding phenotypic studies aimed at understanding their potential roles in human health and disease. However, various strategies used for analysis of metagenomic datasets rely on high-quality reference databases[3]. This highlights the need for extensive and well-characterized collections of reference genomes, such as those from the Human Microbiome Project (HMP)[4,5] and the Human Gastrointestinal Bacteria Genome Collection (HGG)[6–8]. Despite a new wave of culturing efforts, there is still a substantial but undetermined degree of unclassified microbial diversity within the gut ecosystem[6,8–11]. Whereas these unknown community members may have eluded current culturing strategies for a variety of reasons (for example, owing to lack of nutrients in growth media or their low abundance in the gut), they are likely to perform important biological roles that remain undiscovered. Thus, having access to a comprehensive catalogue of representative genomes and isolates from the intestinal microbiota is essential to gain new mechanistic insights.

Culture-independent and reference-free approaches have proved to be successful strategies for species discovery and characterization[12–16]. The most common approach is to perform de novo assembly of shotgun metagenomic reads into contig sequences and place them into different bins on the basis of sequence coverage and tetranucleotide frequency[15]—a process that enables the recovery of potential genomes, termed metagenome-assembled genomes (MAGs). Several studies have applied these methods to reconstruct large numbers of MAGs[13,17–19], one of the most prominent being the recovery of thousands of genomes revealing new insights into the tree of life[16].

Here we generated and classified a set of 92,143 MAGs from 11,850 human gut metagenome assemblies to expand our understanding of gut-associated microbiome diversity. We discovered 1,952 uncultured bacterial species and investigated their association with specific geographical backgrounds, as well as their unique functional capacity. This enabled new insights into which species and functions within this uncharacterized bacterial community might have underappreciated roles in the human gut environment.

Large-scale discovery of uncultured species

To perform a comprehensive characterization of the human gastrointestinal microbiota, we retrieved 13,133 human gut metagenomic datasets from 75 different studies (Supplementary Table 1 and Extended Data Fig. 1). Samples were collected mainly from North America ($n = 6,869$, 52%) or Europe ($n = 4,716$, 36%), reflecting a geographical bias in current human gut microbiome studies. The majority of datasets with available metadata were from diseased patients ($n = 4,323$, 33%) and adults ($n = 3,053$, 23%).

Following assembly with SPAdes[20,21], 11,850 of the 13,133 metagenome assemblies produced contigs that could undergo genomic binning by MetaBAT[15], generating a total of 242,836 bins. The quality of each bin was evaluated with CheckM[22] according to the level of genome completeness and contamination (Extended Data Fig. 2). On the basis of these metrics, 40,029 MAGs with more than 90% completeness and less than 5% contamination were obtained (hereafter referred to as 'near-complete'[16]). We also generated 65,671 medium-quality[23] MAGs (at least 50% completeness and less than 10% contamination), 52,347 of which had a quality score[16] (QS) above 50 (defined as completeness – (5 × contamination)). The robustness of our MAGs was evaluated with two independent assembly/binning methodologies[24,25] (see Supplementary Discussion and Extended Data Fig. 3), which showed the MAGs to be highly reproducible, independent of the method used for assembly or binning.

As CheckM is unable to evaluate non-prokaryotic genomes, we investigated separately how many of our bins represented known eukaryotes or viral sequences (see Supplementary Discussion and Supplementary Table 2). However, for the main set of analyses, we focused on the 39,891 near-complete MAGs that CheckM resolved to bacterial lineages

[1]European Bioinformatics Institute (EMBL-EBI), Wellcome Genome Campus, Hinxton, UK. [2]Wellcome Sanger Institute, Wellcome Genome Campus, Hinxton, UK. [3]Centre for Innate Immunity and Infectious Diseases, Hudson Institute of Medical Research, Clayton, Victoria, Australia. [4]Department of Molecular and Translational Sciences, Monash University, Clayton, Victoria, Australia. [5]Department of Biochemistry, University of Western Ontario, London, Ontario, Canada. *e-mail: aalmeida@ebi.ac.uk; rdf@ebi.ac.uk

Use case: Metagenomics pipeline

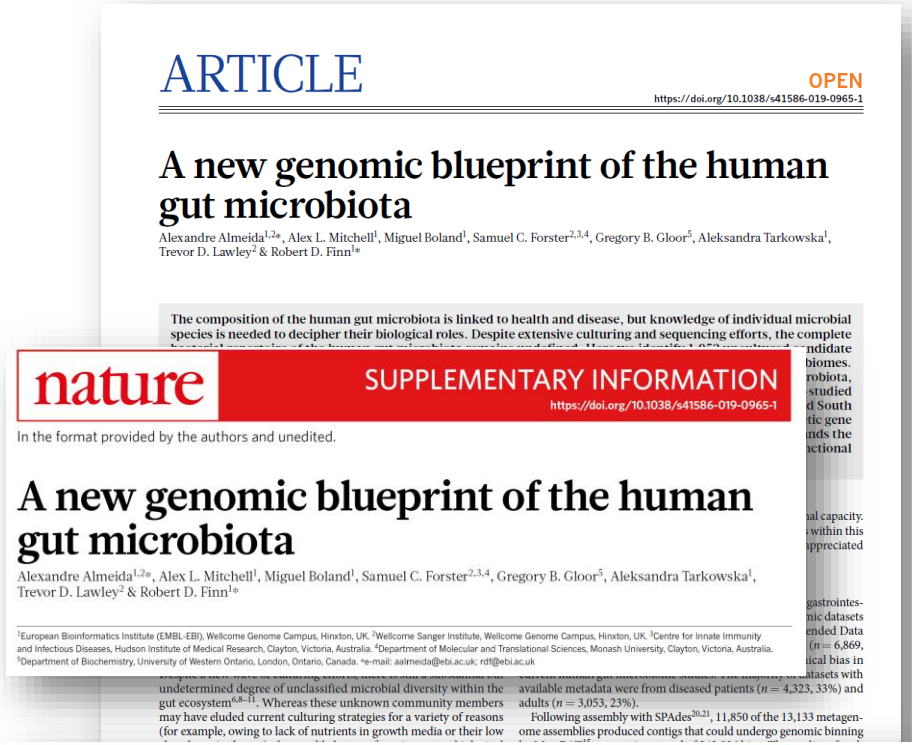
Identifies 1952 bacterial species from the human gut

Computationally intense pipeline to assembles the sequencing reads

Genomics:
Plethora of open source tools for different steps

Almeida, A., Mitchell, A.L., Boland, M. *et al.* A new genomic blueprint of the human gut microbiota. *Nature* **568**, 499–504 (2019). https://doi.org/10.1038/s41586-019-0965-1

FAIR Mixed and Multi Object



ARTICLE

OPEN

https://doi.org/10.1038/s41586-019-0965-1

A new genomic blueprint of the human gut microbiota

Alexandre Almeida[1,2*], Alex L. Mitchell[1], Miguel Boland[1], Samuel C. Forster[2,3,4], Gregory B. Gloor[5], Aleksandra Tarkowska[1], Trevor D. Lawley[2] & Robert D. Finn[1*]

The composition of the human gut microbiota is linked to health and disease, but knowledge of individual microbial species is needed to decipher their biological roles. Despite extensive culturing and sequencing efforts, the complete

nature SUPPLEMENTARY INFORMATION

https://doi.org/10.1038/s41586-019-0965-1

In the format provided by the authors and unedited.

A new genomic blueprint of the human gut microbiota

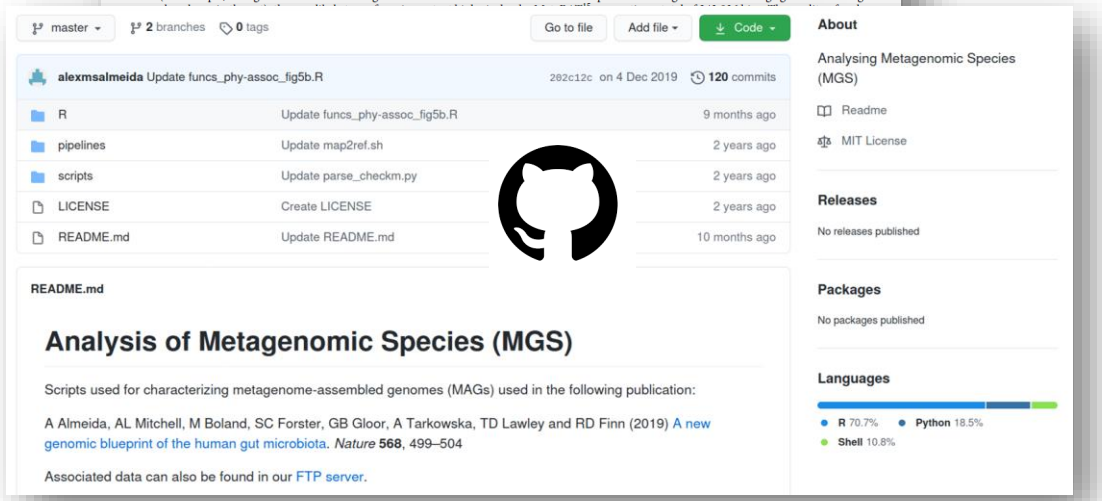
Alexandre Almeida[1,2*], Alex L. Mitchell[1], Miguel Boland[1], Samuel C. Forster[2,3,4], Gregory B. Gloor[5], Aleksandra Tarkowska[1], Trevor D. Lawley[2] & Robert D. Finn[1*]

[1]European Bioinformatics Institute (EMBL-EBI), Wellcome Genome Campus, Hinxton, UK. [2]Wellcome Sanger Institute, Wellcome Genome Campus, Hinxton, UK. [3]Centre for Innate Immunity and Infectious Diseases, Hudson Institute of Medical Research, Clayton, Victoria, Australia. [4]Department of Molecular and Translational Sciences, Monash University, Clayton, Victoria, Australia. [5]Department of Biochemistry, University of Western Ontario, London, Ontario, Canada. *e-mail: aalmeida@ebi.ac.uk; rdf@ebi.ac.uk

undetermined degree of unclassified microbial diversity within the gut ecosystem[6–11]. Whereas these unknown community members may have eluded current culturing strategies for a variety of reasons (for example, owing to lack of nutrients in growth media or their low

available metadata were from diseased patients ($n = 4,323$, 33%) and adults ($n = 3,053$, 23%).

Following assembly with SPAdes[20,21], 11,850 of the 13,133 metagenome assemblies produced contigs that could undergo genomic binning



master · 2 branches · 0 tags · Go to file · Add file · Code

alexmsalmeida Update funcs_phy-assoc_fig5b.R — 202c12c on 4 Dec 2019 — 120 commits

R	Update funcs_phy-assoc_fig5b.R	9 months ago
pipelines	Update map2ref.sh	2 years ago
scripts	Update parse_checkm.py	2 years ago
LICENSE	Create LICENSE	2 years ago
README.md	Update README.md	10 months ago

README.md

Analysis of Metagenomic Species (MGS)

Scripts used for characterizing metagenome-assembled genomes (MAGs) used in the following publication:

A Almeida, AL Mitchell, M Boland, SC Forster, GB Gloor, A Tarkowska, TD Lawley and RD Finn (2019) A new genomic blueprint of the human gut microbiota. *Nature* **568**, 499–504

Associated data can also be found in our FTP server.

About

Analysing Metagenomic Species (MGS)

Readme

MIT License

Releases

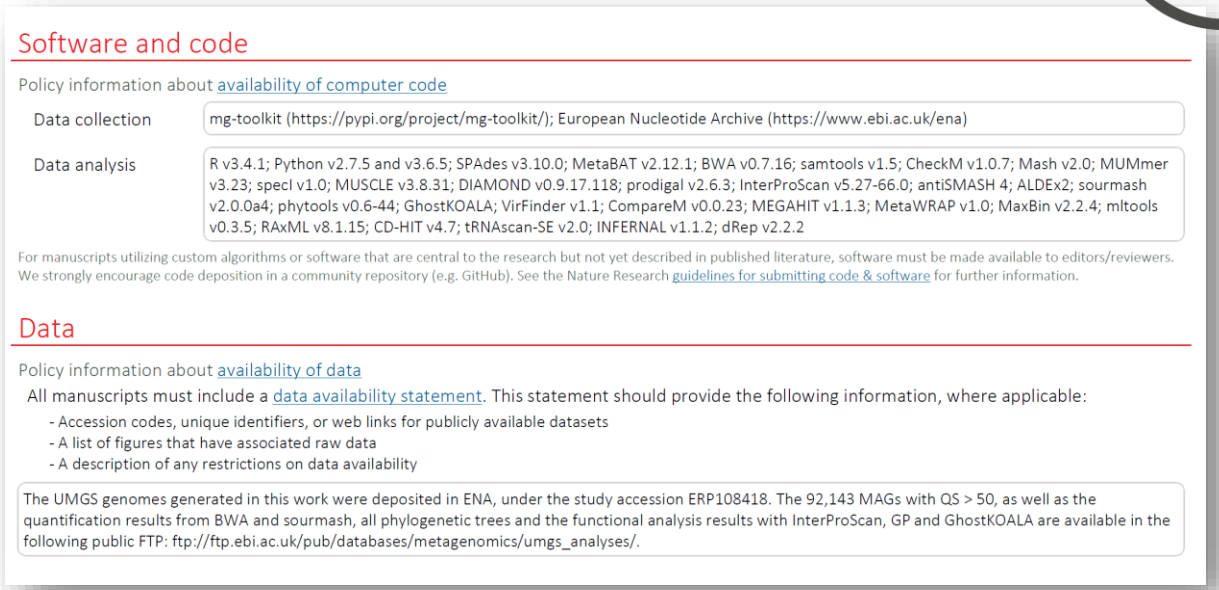
No releases published

Packages

No packages published

Languages

R 70.7% · Python 18.5% · Shell 10.8%



Software and code

Policy information about availability of computer code

Data collection	mg-toolkit (https://pypi.org/project/mg-toolkit/); European Nucleotide Archive (https://www.ebi.ac.uk/ena)
Data analysis	R v3.4.1; Python v2.7.5 and v3.6.5; SPAdes v3.10.0; MetaBAT v2.12.1; BWA v0.7.16; samtools v1.5; CheckM v1.0.7; Mash v2.0; MUMmer v3.23; specI v1.0; MUSCLE v3.8.31; DIAMOND v0.9.17.118; prodigal v2.6.3; InterProScan v5.27-66.0; antiSMASH 4; ALDEx2; sourmash v2.0.0a4; phytools v0.6-44; GhostKOALA; VirFinder v1.1; CompareM v0.0.23; MEGAHIT v1.1.3; MetaWRAP v1.0; MaxBin v2.2.4; mltools v0.3.5; RAxML v8.1.15; CD-HIT v4.7; tRNAscan-SE v2.0; INFERNAL v1.1.2; dRep v2.2.2

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

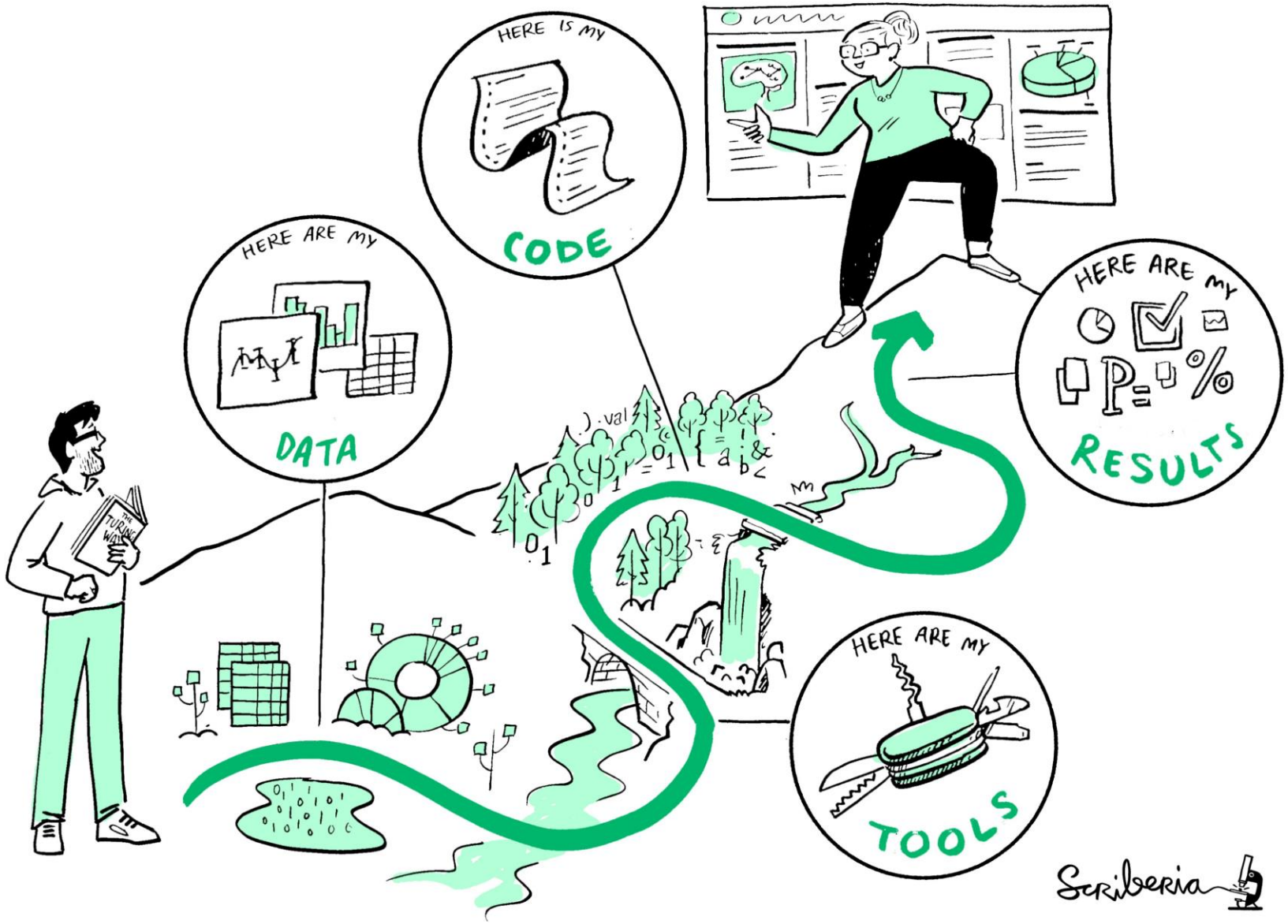
The UMGS genomes generated in this work were deposited in ENA, under the study accession ERP108418. The 92,143 MAGs with QS > 50, as well as the quantification results from BWA and sourmash, all phylogenetic trees and the functional analysis results with InterProScan, GP and GhostKOALA are available in the following public FTP: ftp://ftp.ebi.ac.uk/pub/databases/metagenomics/umgs_analyses/.

Source data and results

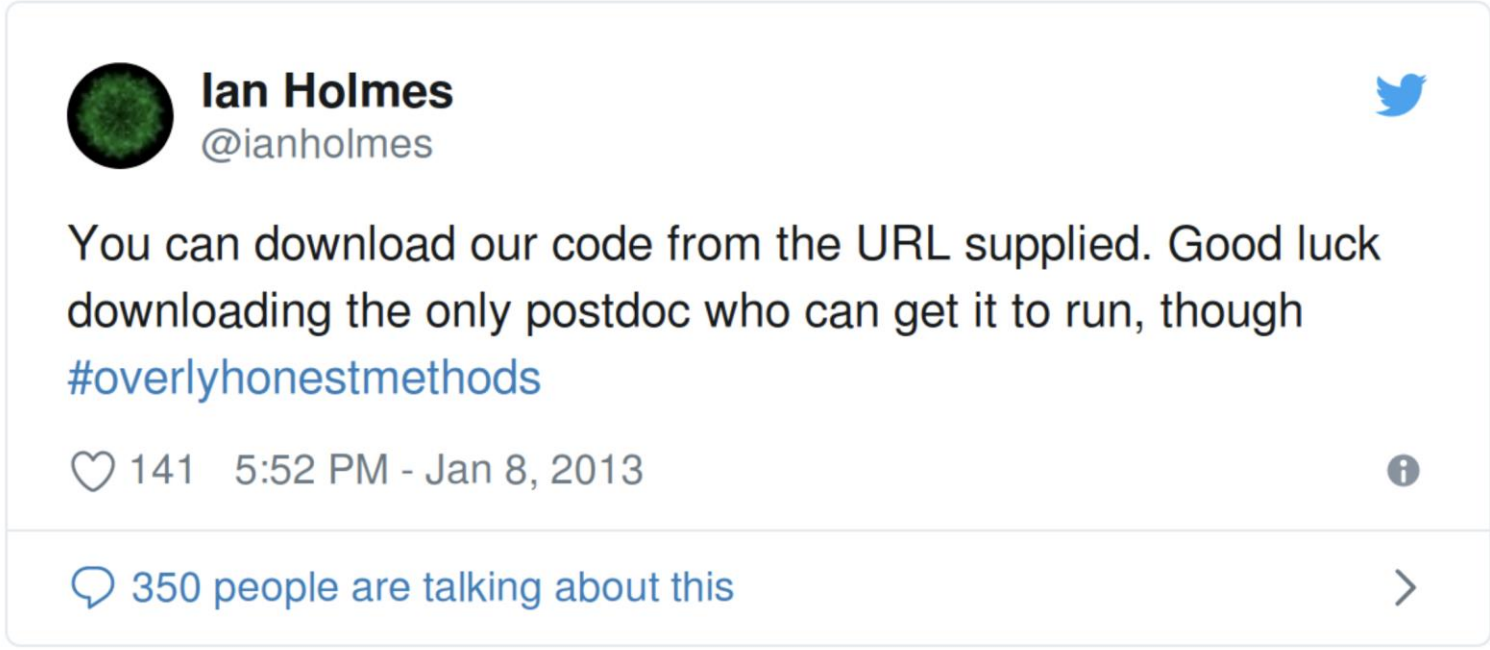
Instruments, software, workflows, scripts…

Different data types…

Public archives, spreadsheets, project ftp servers…



This image was created by Scriberia for The Turing Way community and is used under a CC-BY licence.



https://twitter.com/ianholmes/status/288689712636493824

Why use workflows?

Automation

- Automate computational aspects
- Repetitive pipelines, sweep campaigns

Scaling—compute cycles

- Make use of computational infrastructure
- Handle large data

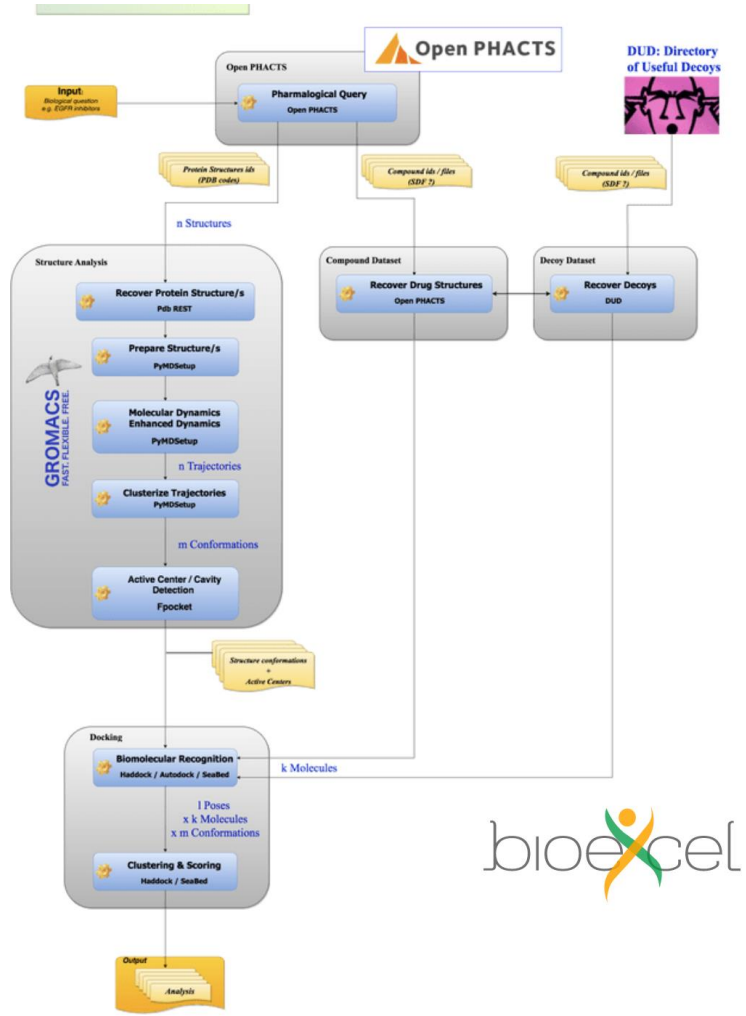
Abstraction—people cycles

- Shield complexity and incompatibilities
- Report, re-usue, evolve, share, compare
- Repeat—Tweak—Repeat
- First-class commodities

Provenance—reporting

- Capture, report and utilize log and data lineage
- Auto-documentation
- Tracable evolution, audit, transparency
- Reproducible science

Findable
Accessible
Interoperable
Reusable
(Reproducible)



Adapted from Bertram Ludäscher (2015)

https://www.slideshare.net/ludaesch/works-2015provenancemileage

https://doi.org/10.1007/s13222-012-0100-z

Existing Workflow systems

Michael R. Crusoe edited this page on Sep 13, 2022 · 329 revisions

Permalink: https://s.apache.org/existing-workflow-systems

Cite as (update dates):

Peter Amstutz, Maxim Mikheev, Michael R. Crusoe, Nebojša Tijanić, Samuel Lampa, et al. (2022): **Existing Workflow systems.** *Common Workflow Language wiki*, GitHub. https://s.apache.org/existing-workflow-systems updated 2022-08-30, accessed 2022-08-30.

Computational Data Analysis Workflow Systems

An incomplete list

Please add new entries at the bottom.

In addition to this list, actively developed free/open-source systems should be registered at https://workflows.community/systems

See also: https://github.com/pditommaso/awesome-pipeline

1. Arvados - CWL-based distributed computing platform for data analysis on massive data sets. https://arvados.org/ https://github.com/arvados/arvados
2. Apache Taverna http://www.taverna.org.uk/ https://taverna.incubator.apache.org/
3. Galaxy http://galaxyproject.org/
4. SHIWA https://www.shiwa-workflow.eu/
5. Apache Oozie https://oozie.apache.org/
6. DNANexus https://wiki.dnanexus.com/API-Specification-v1.0.0/IO-and-Run-Specifications https://wiki.dnanexus.com/API-Specification-v1.0.0/Workflows-and-Analyses
7. ~~BioDT http://www.biodatomics.com/~ archived at https://web.archive.org/web/20180609011656/http:

https://s.apache.org/existing-workflow-systems

302. redun (yet another redundant workflow engine) https://github.com/insitro/
303. pyiron (The materials science IDE) https://pyiron.org/
304. looper (pipeline submitting engine) https://github.com/pepkit/looper
305. dagster (Python based data orchestration platform) https://dagster.io/
306. StackStorm (Devops automation engine) https://stackstorm.com/
307. Geoweaver (compose and execute full-stack deep learning workflows) http
308. Popper: Container-native task automation engine https://github.com/getpo
309. Cloud Build: Build, test, and deploy on our serverless CI/CD platform https:/
310. Task/Taskfile: A task runner / simpler Make alternative written in Go https://
311. pypyr: task runner for automation pipelines script sequential task workflow error handling & retries https://pypyr.io/
312. SimTool/Sim2Ls: Jupyter notebook-based pipelines of Simulation Tools for https://github.com/hubzero/simtool https://simtool.readthedocs.io/ https:/
313. SideIO: A Side I/O system framework for hybrid scientific workflow (*no proj* /10.1016/j.jpdc.2016.07.001
314. Flyte https://flyte.org/
315. StreamFlow https://streamflow.di.unito.it/
316. Jupyter Workflow https://jupyter-workflow.di.unito.it/
317. Nnodes: a simple workflow manager for Python functions and command li
318. Orchest: A GUI for developing, running and managing container workflows
319. Wasmflow: platform for building applications out of WebAssembly code co /wasmflow https://wasmflow.com/
320. HyperShell: cross-platform, high-performance computing utility for process asynchronous queue. https://doi.org/10.1145/3491418.3535138 https://gitl
321. Covalent: Pythonic distributed workflow tool used to prototype and run hig https://github.com/AgnostiqHQ/covalent
322. Icolos: workflow manager for structure-based workflows in computational /Icolos https://doi.org/10.26434/chemrxiv-2022-vqbxg
323. dwork Task graph scheduler with a minimalistic API. https://github.com/fro
324. pmake parallel make developed for use within batch jobs https://docs.olcf. /pmake.html#workflows-pmake

https://www.commonwl.org/

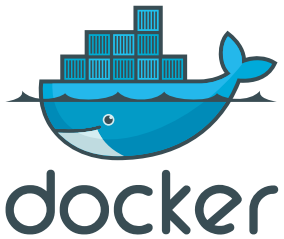
COMMON WORKFLOW LANGUAGE

OPEN AND FREE	COMMUNITY FIRST	INTEROPERABILITY AND PORTABILITY	VENDOR NEUTRALITY
Free and open standards	Community is a core principle of the CWL Project	Portable and interoperable across a variety of software and deployment environments	Developed by a multi-vendor working group of organizations and individuals/contributors

REUSABILITY AND REPRODUCIBILITY	PARALLELIZATION AND SCALE	ECOSYSTEM SUPPORT	TRANSPARENT GOVERNANCE
Enables scientists to reuse and reproduce their data analysis workflows	Scalable from workstations to cluster, cloud, and high performance computing (HPC environments	Supported by an ecosystem of tools, libraries, and editor plugins	Designed with an open and transparent governance

One standard, many CWL implementations

- **commercially** supported systems that support CWL from vendors such as **Curii** (Arvados), DNAnexus, **IBM** (IBM® Spectrum LSF), Illumina (Illumina Connected Analytics), and **Seven Bridges Genomics**/Velsera.
- **Open source** implementations: **cwltool** (ref engine), **Arvados** (*AWS, GCP, Azure, Slurm, LSF*), **Toil** (*AWS, Azure, GCP, Grid Engine, HTCondor, LSF, Mesos, OpenStack, Slurm, PBS/Torque*), **CWL-Airflow** (*Apache Airflow*), **StreamFlow** (*Kubernetes, HPC with Singularity, PBS, Slurm*), **Occam** (*multi-node SSH, local, Docker, Singularity*)
- Partial implementations (ep3, REANA, Xenon, Galaxy, cwl-tes, Calrissian, Pegasus)



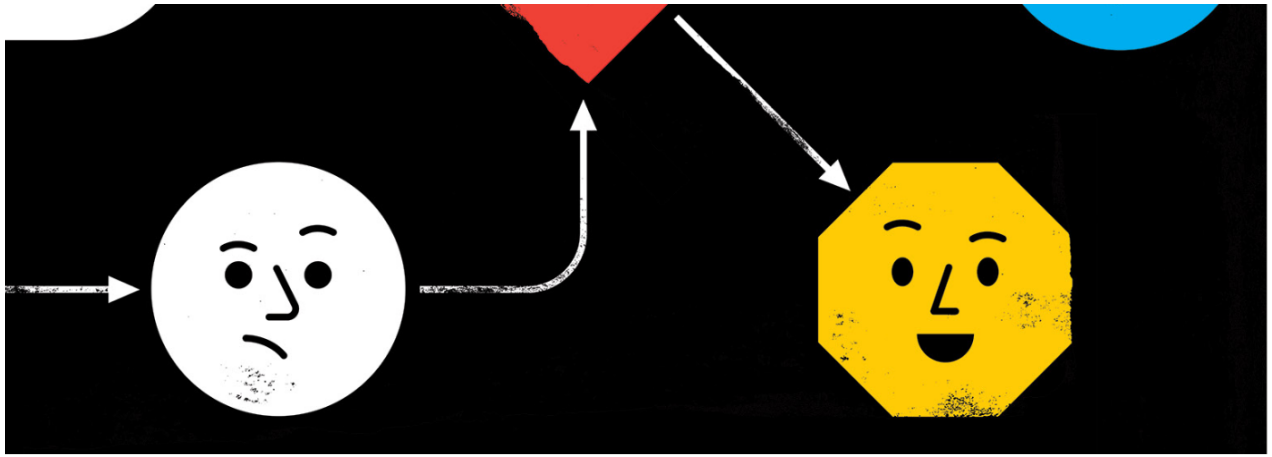

```yaml
cwlVersion: v1.0
class: Workflow
inputs:
  inp: File
  ex: string

outputs:
  classout:
    type: File
    outputSource: compile/classfile

steps:
  untar:
    run: tar-param.cwl
    in:
      tarfile: inp
      extractfile: ex
    out: [example_out]

  compile:
    run: arguments.cwl
    in:
      src: untar/example_out
    out: [classfile]
```

TOOLBOX | 02 September 2019

Workflow systems turn raw data into scientific knowledge

How workflow tools can make your computational methods portable, maintainable, reproducible and shareable.

Jeffrey M. Perkel



BY JEFFREY M. PERKEL

Reinventing the wheel is pointless, but for computational biologists it's sometimes unavoidable. So when Rob Finn and Folker Meyer realized how much their work overlapped, they decided to try something different.

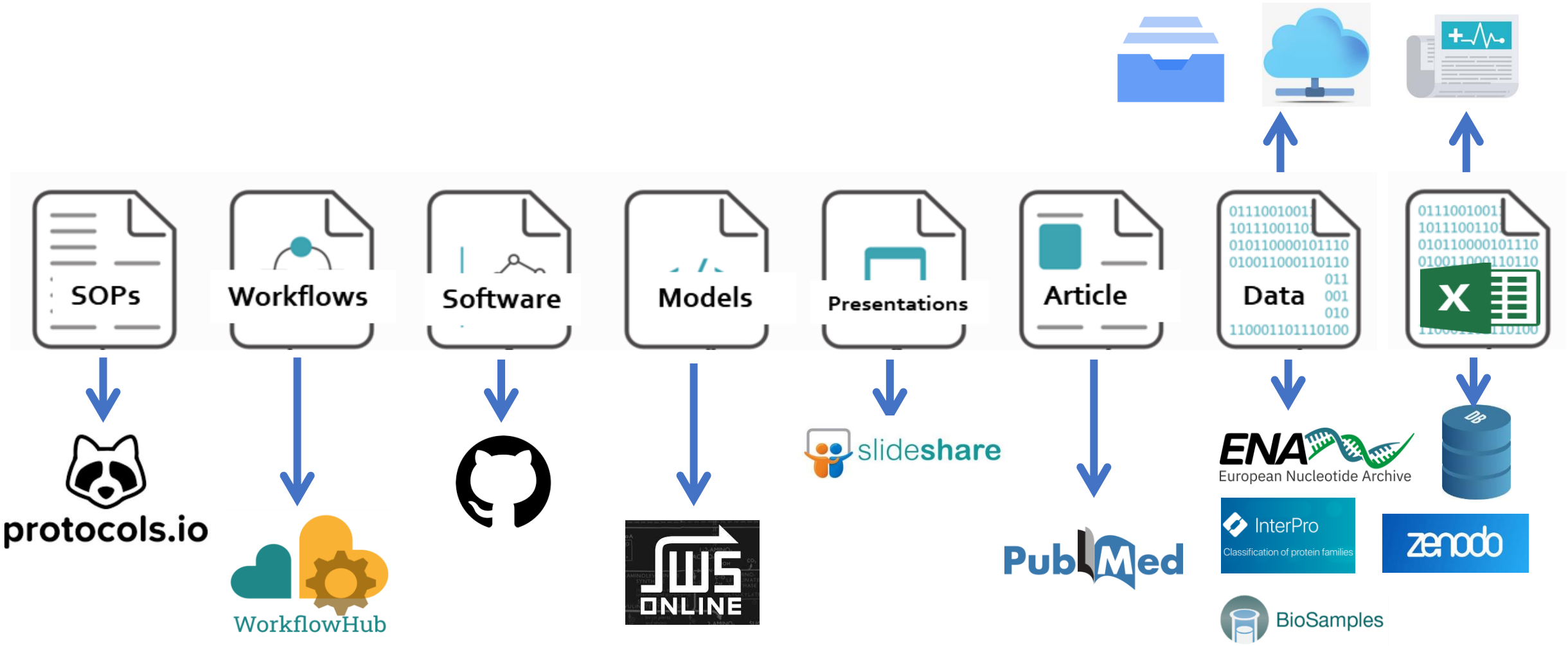
Finn is head of the sequence-families team at the European Bioinformatics Institute (EBI) in Hinxton, UK; Meyer is a computer scientist at Argonne National Laboratory in Lemont, Illinois. Both run facilities that let researchers perform a computationally intensive process called metagenomic analysis, which allows microbial communities to be reconstructed from shards of DNA. It would be helpful, they realized, if they could try each other's code. The problem was that their analytical 'pipelines' — the carefully choreographed computational steps required to turn raw data into scientific knowledge — were written in different languages. Meyer's team was using an in-house system called AWE, whereas Finn was working with nearly 9,500 lines of Python code.

"It was a horrible Python code base," says Finn — complicated, and difficult to maintain. "Bits had been bolted on in an ad hoc fashion over seven years by at least four different developers." And it was "heavily tied to the compute infrastructure", he says, meaning it was written for specific computational resources and a particular way of organizing files, and thus essentially unusable outside the EBI. Because the EBI wasn't using AWE, the reverse was also true. Then Finn and Meyer learnt about the Common Workflow Language (CWL).

CWL is a way of describing analytical pipelines and computational tools — one of more than 250 systems now available, including such popular options as Snakemake, Nextflow and Galaxy. Although they speak different languages and support different features, these systems have a common aim: to make computational methods reproducible, portable, maintainable and shareable. CWL is essentially an exchange language that researchers can use to share pipelines for whichever system. For Finn, that

https://doi.org/10.1038/d41586-019-02619-z

Scattered and diverse metadata
Multiple platforms and repositories

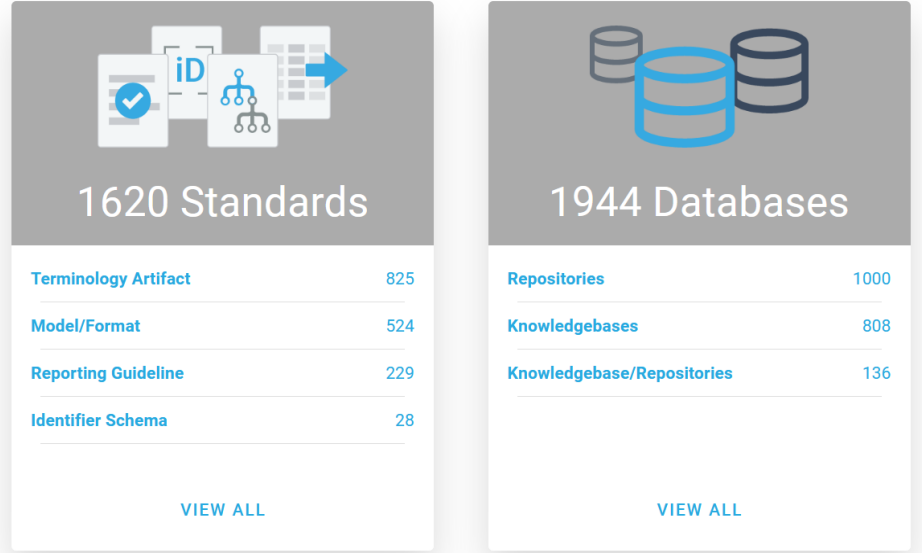


Is it FAIR to use all these repositories?

re3data.org
REGISTRY OF RESEARCH DATA REPOSITORIES

https://www.re3data.org/

https://faircookbook.elixir-europe.org/



FAIRsharing.org
standards, databases, policies

https://fairsharing.org/

Researchers are asked to make their research outputs FAIR – **where** to publish?

Thousands of public, institutional and domain-specific repositories

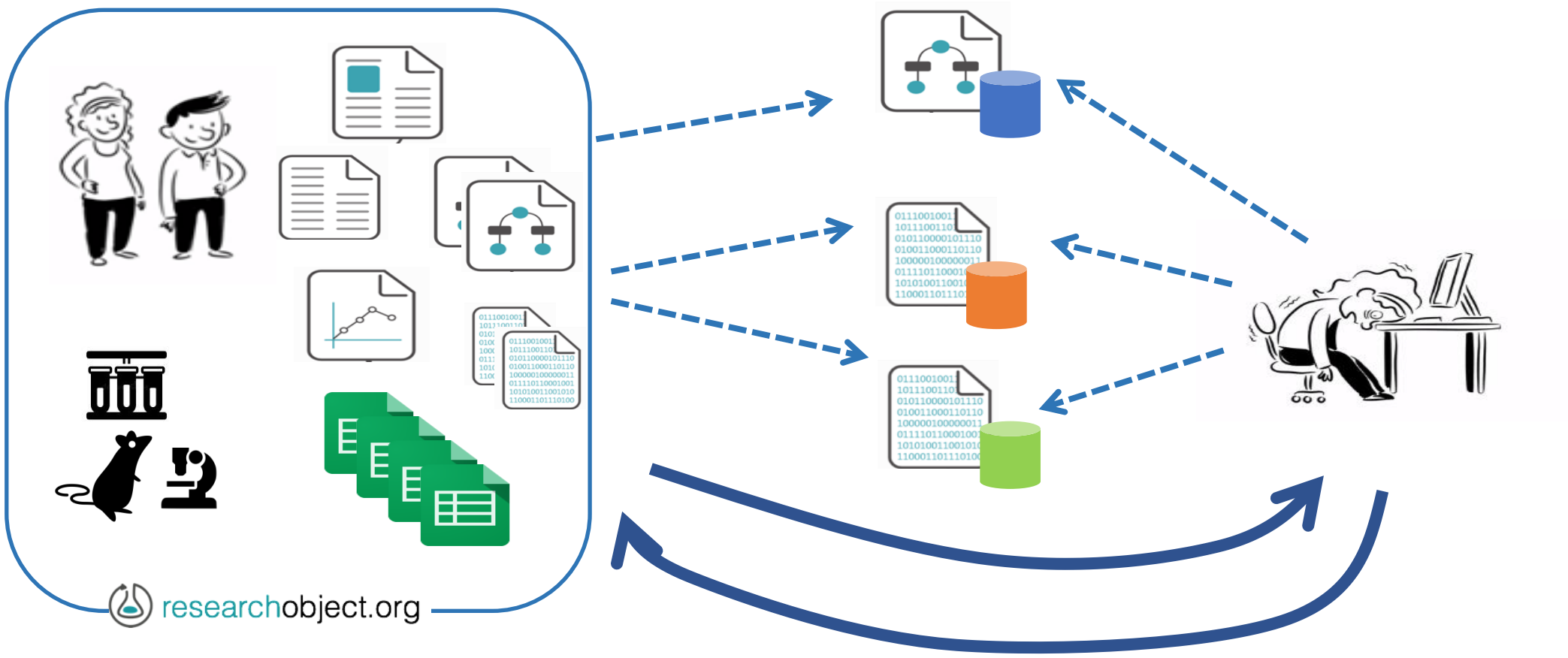
Help from guidance and **catalogues** (FAIRsharing, re3data, FAIR Cookbook)

..but how to gather and reference outputs **across multiple repositories**?

What about **contextual** information?

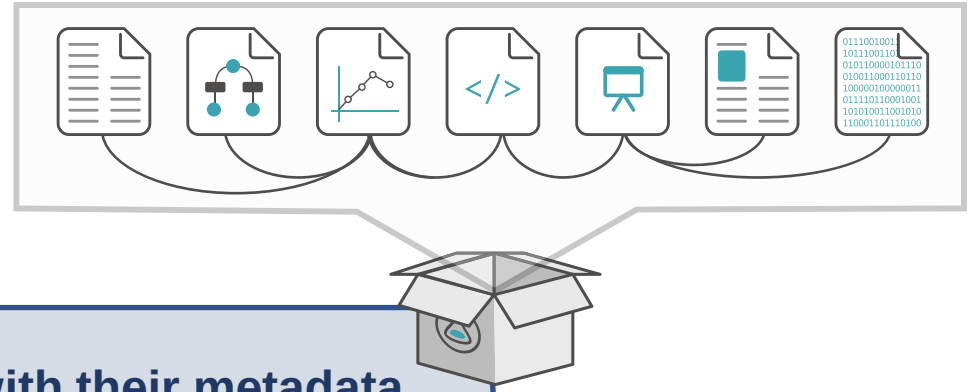
Metadata love letter delivery : Research Objects

Each object in its own repository or platform with its own metadata



https://www.researchobject.org/

Aims of FAIR Research Objects



Describe and **package** data collections, datasets, software etc. **with their metadata**

Platform-independent object exchange between repositories and services

Support **reproducibility** and **analysis**: link data with codes and workflows

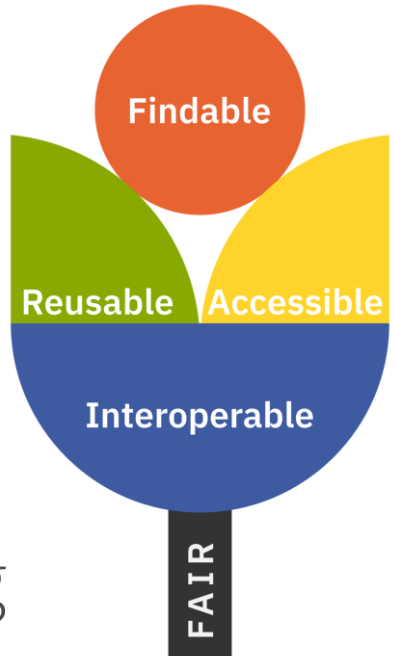
Transfer of **sensitive/large** distributed datasets with persistent identifiers

Aggregate **citations** and **persistent identifiers**

Propagate **provenance** and **existing metadata**

Publish and archive **mixed objects** and references

Reuse existing **standards**, but hide their complexity



research object.org

A little bit of packaging goes a long way

- Practical lightweight packaging
- Aggregate files/directories
  + any content with URI
- Embed contextual information
- Archive with rich structured metadata

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

A little bit of packaging goes a long way

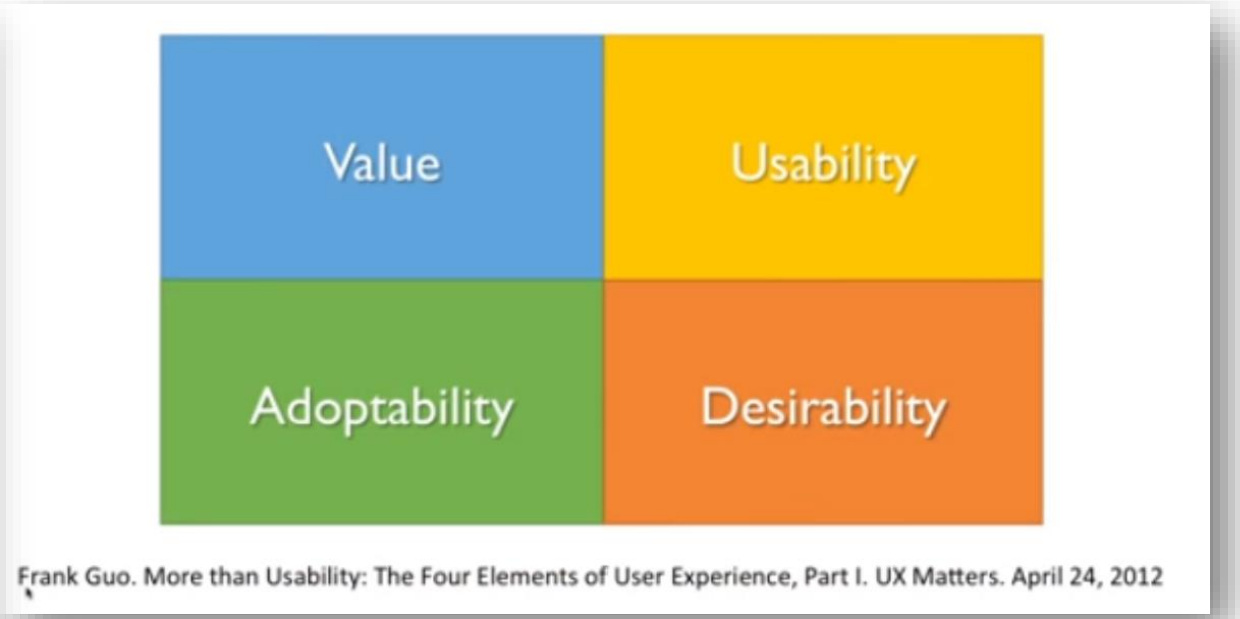
Infrastructure independent to overcome repository and service silos: Practical, lightweight, robust.

Familiar, developer friendly Lo-Tek - web native, off-the-shelf, machine and human readable, search engine accessible: PIDs + JSON-LD + Schema.org + BagIT/Zip/OCDL.

One size does not fit all - embrace diversity, legacy, unknowns – open-ended, multi-interpretation, self-describing. Extensible metadata + pre-existing ontologies: Duck type profiling.

Developers Matter – this is Middleware!

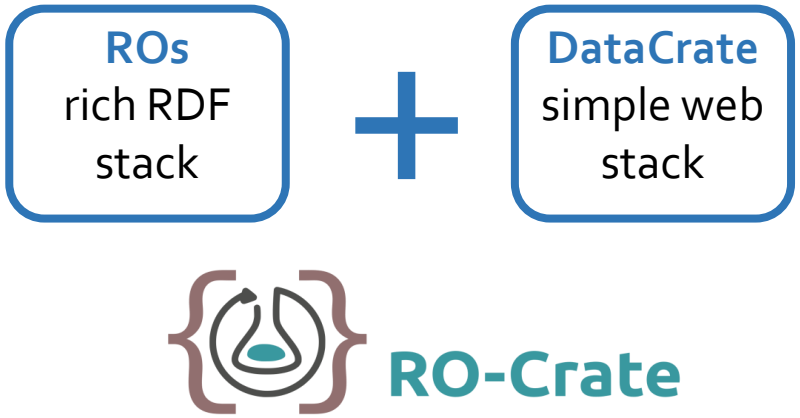
RSECon 2022 – Research Software Engineers!



Frank Guo. More than Usability: The Four Elements of User Experience, Part I. UX Matters. April 24, 2012

https://society-rse.org/

Developer Friendly, Problem Driven



Adoptability!!!!

simplifications rather than generalisations
fewer features, more directed

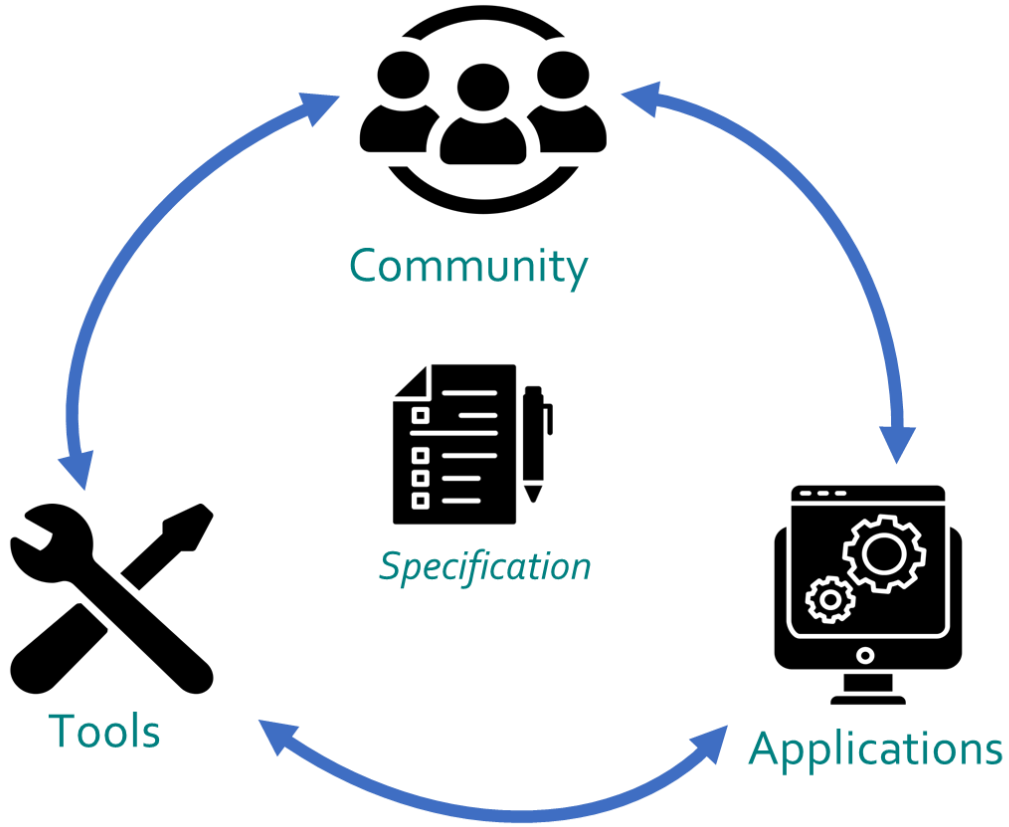
easier to understand, conceptually simpler
opinionated guide to current best practices

constrained and predictable but not too cumbersome to work with

retain "just enough" linked data for benefits
querying, vocabularies, clickable URIs, knowledge graphs

with all the stuff developers need
documentation, examples, libraries, tools

Developer Friendly, Tool development



Infrastructure facing

Software libraries

npm Java JS python™

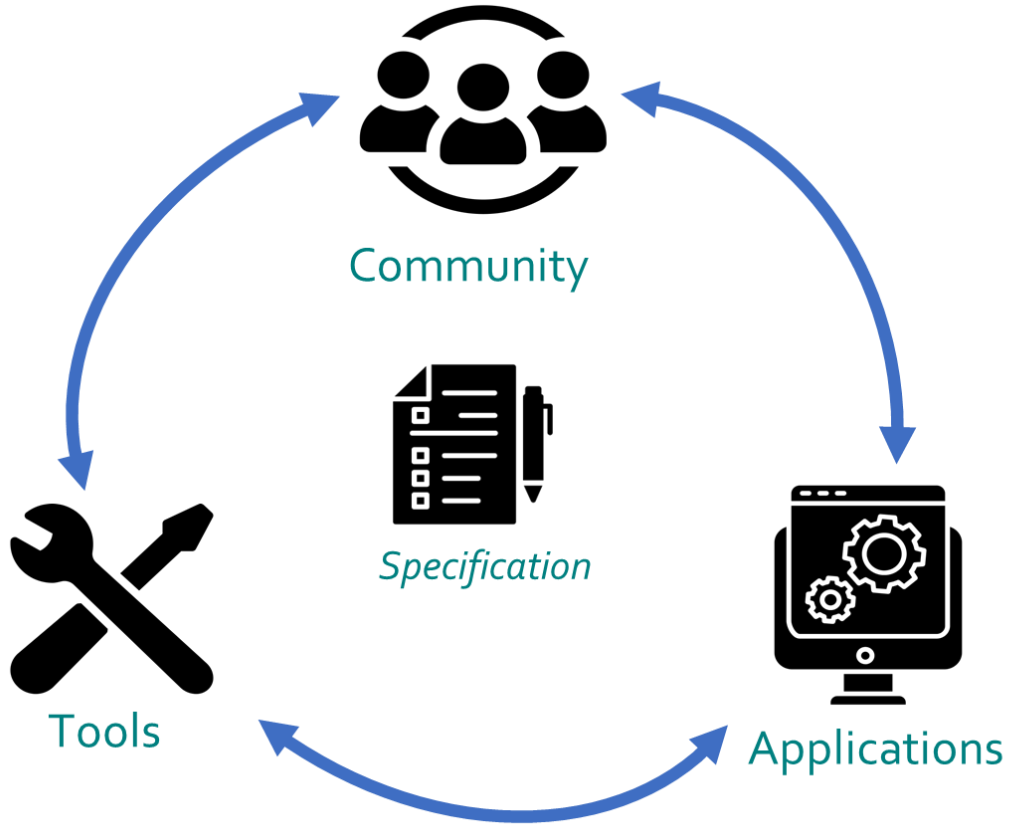
Contact: andreas.pfeil@kit.edu

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

Packaging research artefacts with RO-Crate. *Data Science*
https://doi.org/10.3233/DS-210053

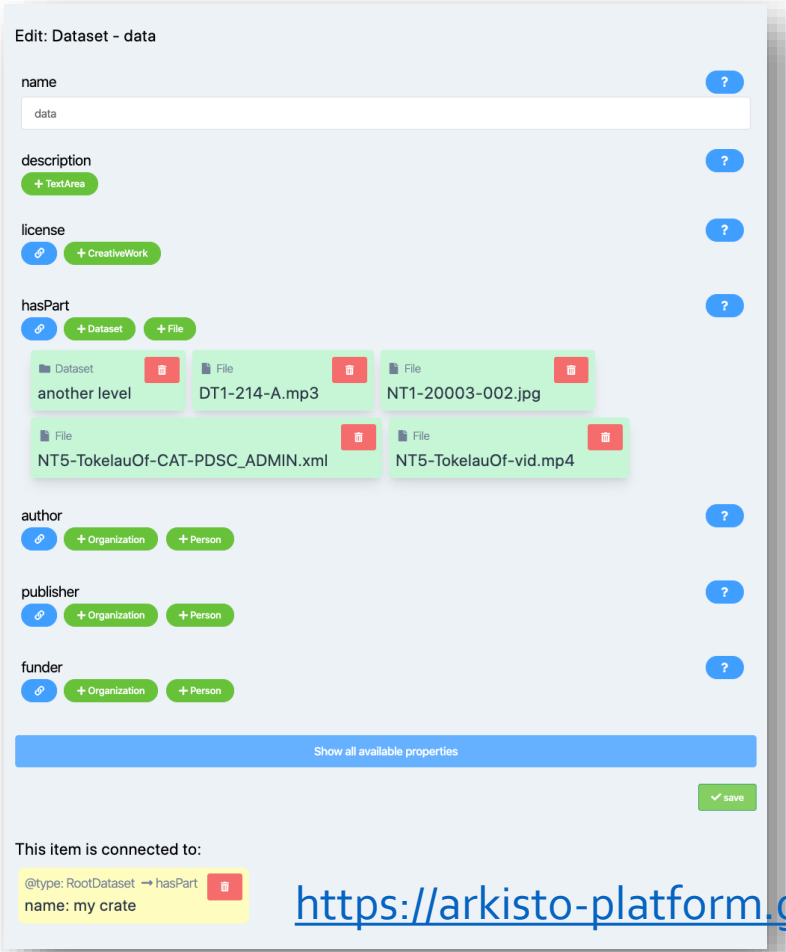
RO-Crate Specification 1.1
https://w3id.org/ro/crate/1.1

Developer Friendly, Tool development



User Facing

Describo



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

Packaging research artefacts with RO-Crate. *Data Science*
https://doi.org/10.3233/DS-210053

Structured self-describing, machine readable, metadata objects

PID

RO-Crate Metadata file

type
id
description
datePublished
...

author

organisation

license

Structured metadata about the RO-Crate and content

Linked Data

JSON-LD Schema.org

RO-Crate Content

PID

type, id
description
datePublished
creator
size
format ...

files

PID

PID

directories

links to web resources

PDF

https://zenodo.org/record/3541888

https://github.com/o/script

Standard Packaging

BagIT, Zip

Archive file format / packaging system

Techie deep-dive!

Warning: JSON ahead

JSON for Linking Data

Data is messy and disconnected. JSON-LD organizes and connects it, creating a better Web.

https://json-ld.org/

⇄ Linked Data

Linked Data empowers people that publish and use information on the Web. It is a way to create a network of standards-based, machine-readable data across Web sites. It allows an application to start at one piece of Linked Data, and follow embedded links to other pieces of Linked Data that are hosted on different sites across the Web.

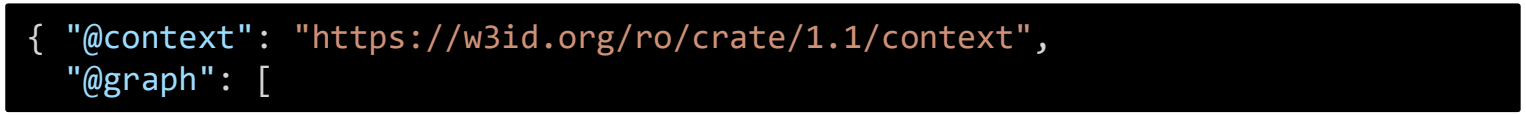
A Simple Example

```
{
  "@context": "http://json-ld.org/contexts/person.jsonld",
  "@id": "http://dbpedia.org/resource/John_Lennon",
  "name": "John Lennon",
  "born": "1940-10-09",
  "spouse": "http://dbpedia.org/resource/Cynthia_Lennon"
}
```

{} JSON-LD

JSON-LD is a lightweight Linked Data format. It is easy for humans to read and write. It is based on the already successful JSON format and provides a way to help JSON data interoperate at Web-scale. JSON-LD is an ideal data format for programming environments, REST Web services, and unstructured databases such as CouchDB and MongoDB.

https://www.w3.org/TR/json-ld/



```
{ "@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 entitie

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

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

Dataset

A Schema.org Type

Thing > CreativeWork > Dataset

[more...]

A body of structured information describing some topic(s) of interest.

Property	Expected Type	Description
Properties from Dataset		
distribution	DataDownload	A downloadable form of this dataset, at a specific location, in a specific format. This property can be repeated if different variations are available. There is no expectation that different downloadable distributions must contain exactly equivalent information (see also DCAT on this point). Different distributions might include or exclude different subsets of the entire dataset, for example.
includedInDataCatalog	DataCatalog	A data catalog which contains this dataset. Supersedes catalog, includedDataCatalog. Inverse property: dataset
issn	Text	The International Standard Serial Number (ISSN) that identifies this serial publication. You can repeat this property to identify different formats of, or the linking ISSN (ISSN-L) for, this serial publication.
measurementTechnique	Text or URL	A technique or technology used in a Dataset (or DataDownload, DataCatalog), corresponding to the method used for measuring the corresponding variable(s) (described using variableMeasured). This is oriented towards scientific and scholarly dataset publication but may have broader applicability; it is not intended as a full representation of measurement, but rather as a high level summary for dataset discovery. For example, if variableMeasured is: molecule concentration, measurementTechnique could be: "mass spectrometry" or "nmr spectroscopy" or "colorimetry" or "immunofluorescence". If the variableMeasured is "depression rating", the measurementTechnique could be "Zung Scale" or "HAM-D" or "Beck Depression Inventory". If there are several variableMeasured properties recorded for some given data object, use a PropertyValue for each variableMeasured and attach the corresponding measurementTechnique.
variableMeasured	PropertyValue or Text	The variableMeasured property can indicate (repeated as necessary) the variables that are measured in some dataset, either described as text or as pairs of identifier and description using PropertyValue.
Properties from CreativeWork		
about	Thing	The subject matter of the content. Inverse property: subjectOf
abstract	Text	An abstract is a short description that summarizes a CreativeWork.
accessMode	Text	The human sensory perceptual system or cognitive faculty through which a person may process or perceive information. Values should be drawn from the approved vocabulary.
accessModeSufficient	ItemList	A list of single or combined accessModes that are sufficient to understand all the intellectual

Suggested new FAIR principle:

A3+ Metadata is not just machine-readable!

> *“ One of the grand challenges of data-intensive science, therefore, is to improve knowledge discovery through assisting both **humans**, and their **computational agents**, in the discovery of, **access** to, and integration and analysis of, task-appropriate scientific data and other scholarly digital objects.*

https://doi.org/10.1038/sdata.2016.18

https://doi.org/10.25504/FAIRsharing.WWI10U

https://doi.org/10.4225/59/5a4d9b76d79f4

https://data.research.uts.edu.au/examples/ro-crate/0.2/Victoria_Arch_pub/

Dataset: Survey of Victoria Arch, Wombeyan Caves NSW

Download this Dataset

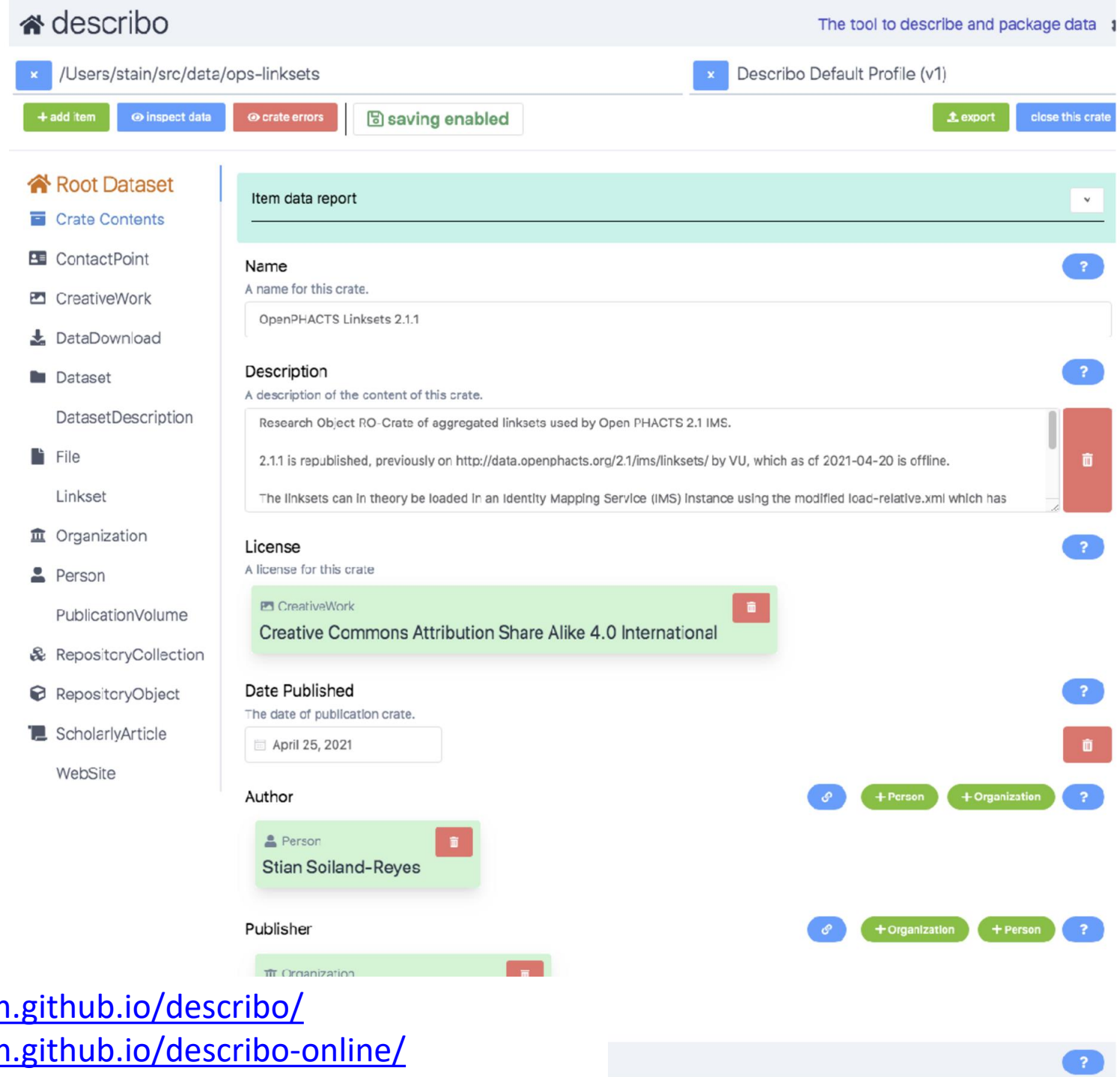
Download all the metadata for Survey of Victoria Arch, Wombeyan Caves NSW in JSON-LD format

Check this crate

FAIR is not just machine-readable!

@id	https://dx.doi.org/10.4225/59/5a4d9b76d79f4
name	Survey of Victoria Arch, Wombeyan Caves NSW
@type	Dataset
description	This data is part of a project by Michael Lake and supported by the Australian Speleological Federation. Data was acquired at Wombeyan Caves by Robert Zlot in January 2014 using the Zebedee 3D Mapping System developed by CSIRO.
datePublished	2017-10-01
creator	• Robert Zlot • Mike Lake • Lukas Kaul
path	./
contactPoint	Contact Mike Lake

https://www.npmjs.com/package/ro-crate-html

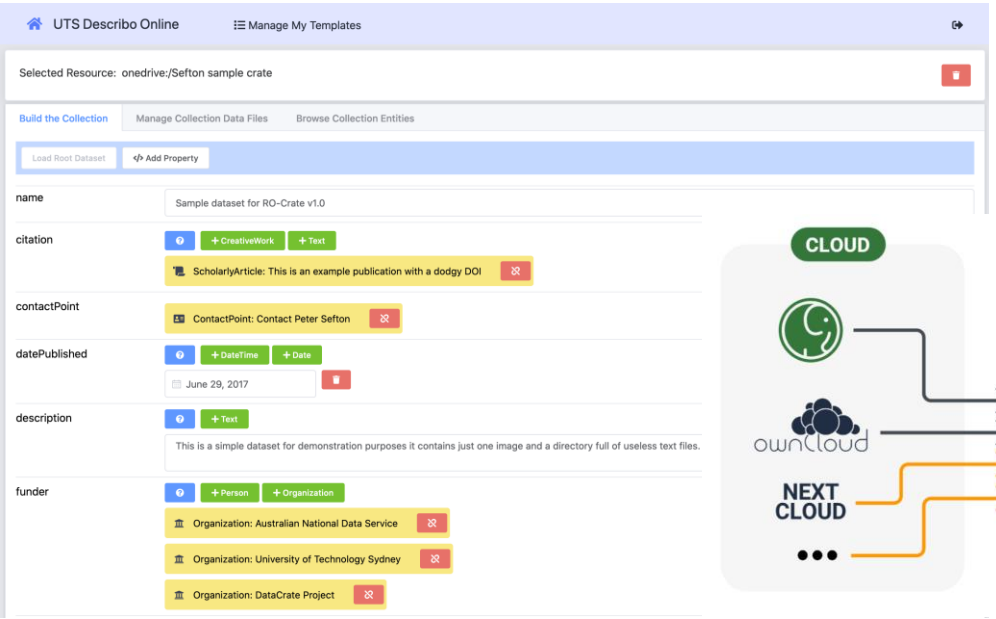


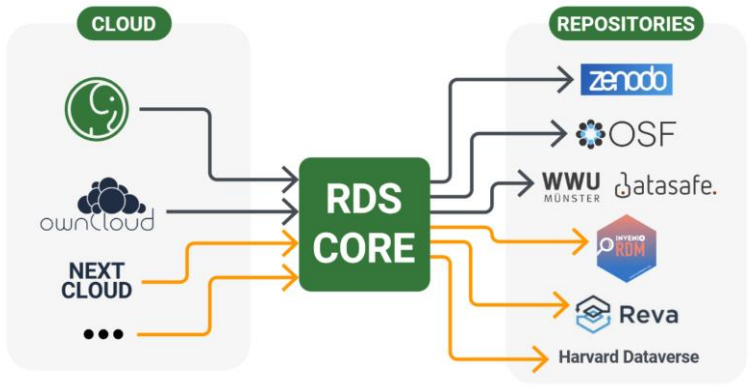
https://arkisto-platform.github.io/describo/
https://arkisto-platform.github.io/describo-online/

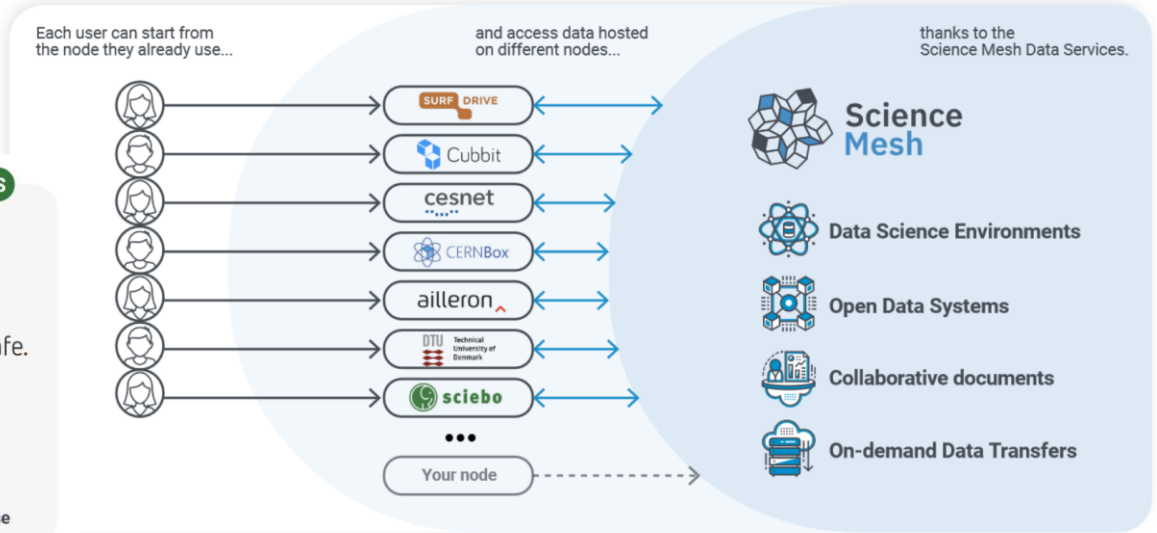
How is RO-Crate being used?

Adding rich metadata to existing data platforms

CS³MESH⁴EOSC
Connecting European Data







The **CS3MESH4EOSC** project combines major data services into the federated **ScienceMesh**

Users can collaborate across established data **repositories** and data science **services**.

FAIR Description Service (based on **Describo Online**) to annotate data using RO-Crate

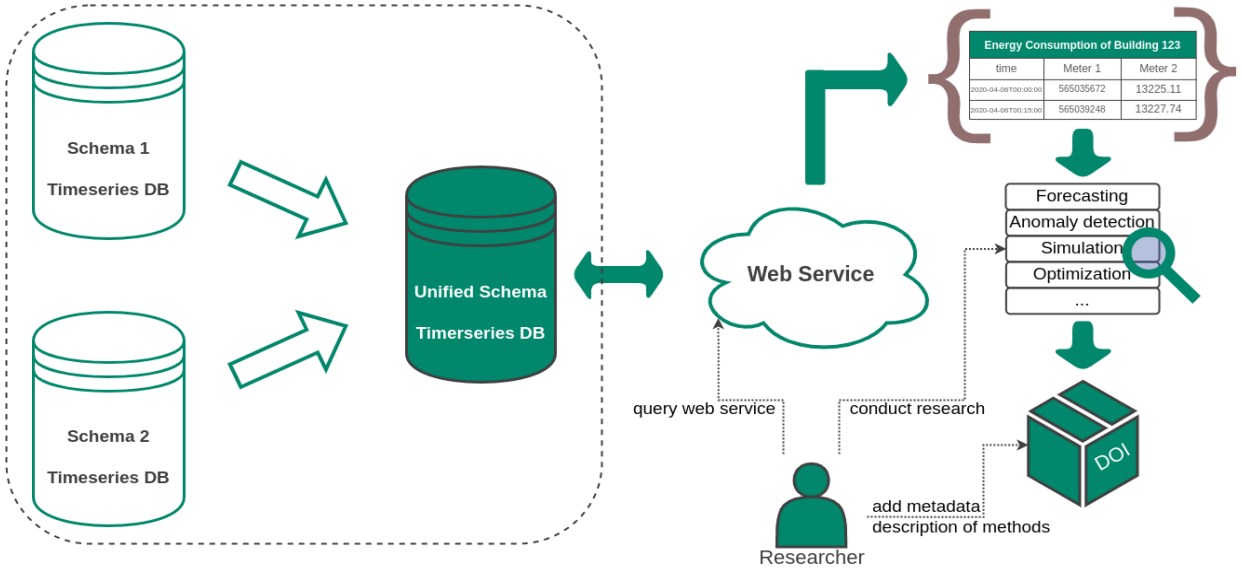
Domain-specific **profiles** for additional metadata requirements

https://doi.org/10.5281/zenodo.7310739
https://doi.org/10.3897/rio.8.e95972
https://cs3mesh4eosc.eu/
https://arkisto-platform.github.io/tools/description/describo-online/

Exporting data using an interchange format

HMC Hub Energy

Time series data from different databases exported with metadata description of their structure and content into a single web service

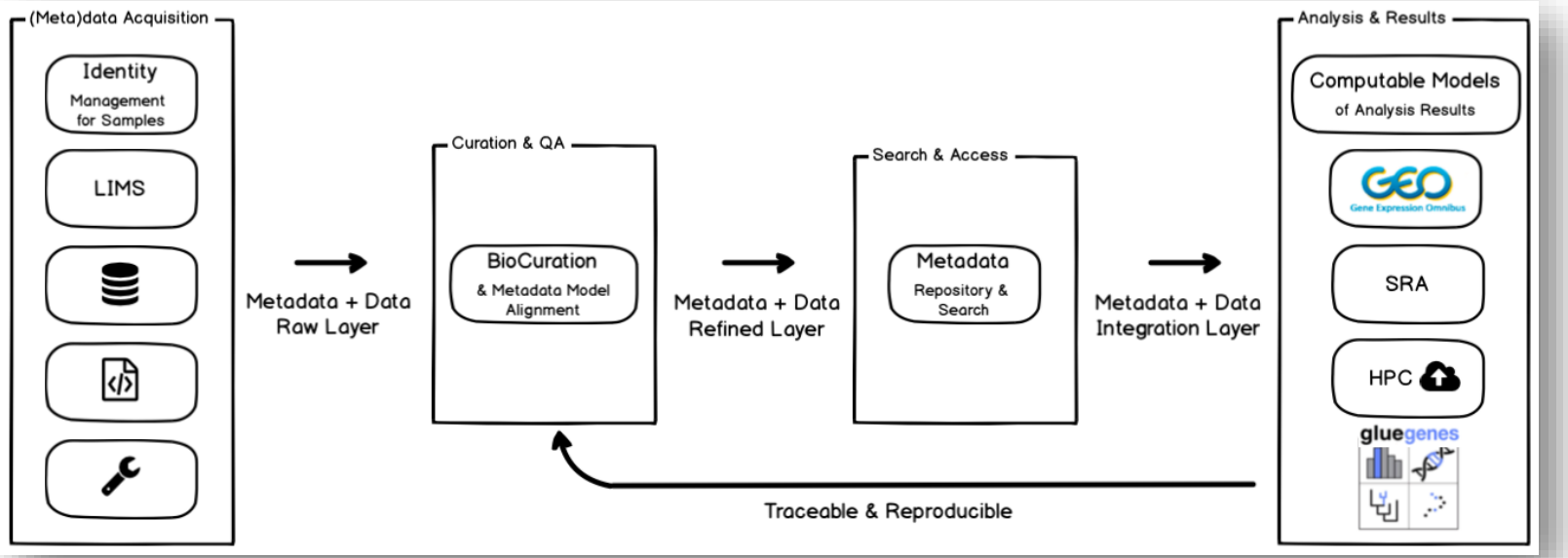


Web service using ro-crate-java

Data file format: CSV

LD-Vocabularies: RO-Crate Context, QUDT, CSVW

BioConnect

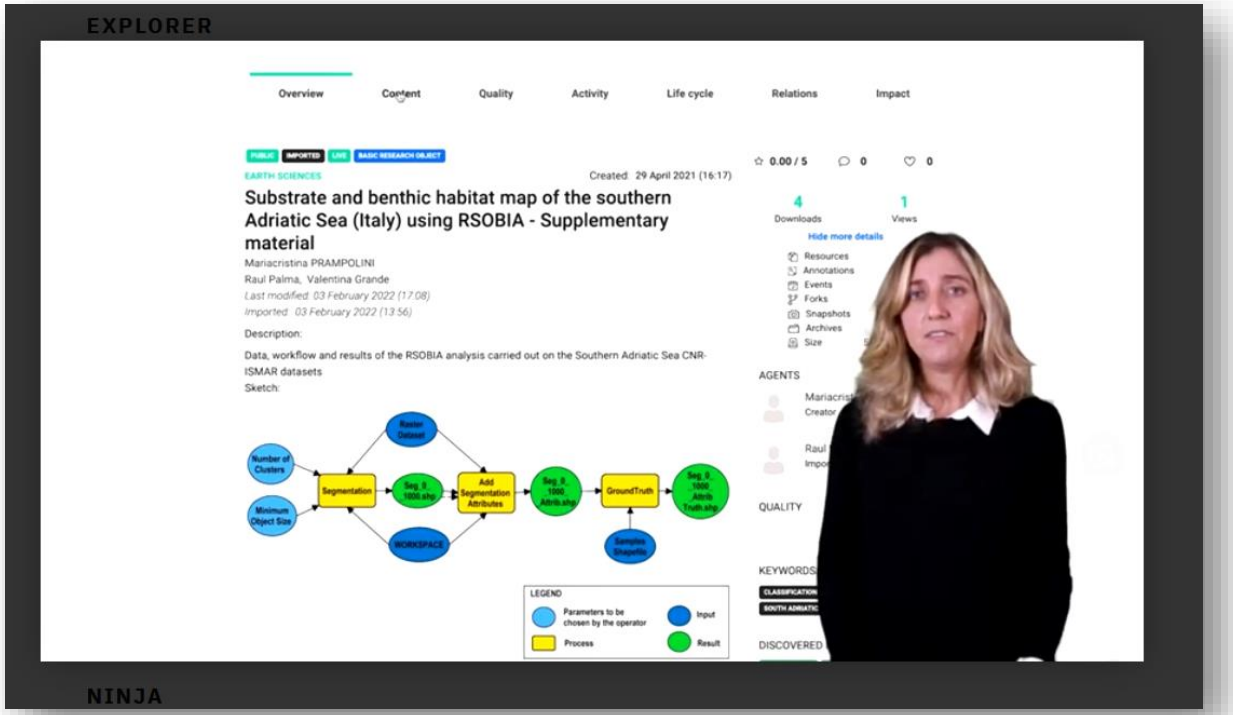


Packaging + Structure + File type defs for tabular data

RO-Crate *ISA format* *Frictionless data*

Abigail Miller https://doi.org/10.5281/zenodo.7116702

EOSC, Copernicus, Earth Science



https://youtu.be/Rsuxn0m4bIM

RO-Crate + Data Cubes

Mixed object sharing

Reproducibility

https://www.reliance-project.eu/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824087

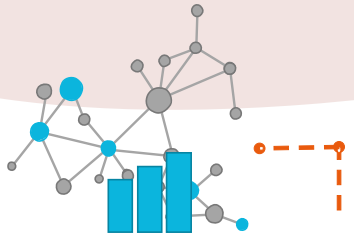
Raul Palma, Oscar Corcho, Daniel Garijo

Annotating plant research data

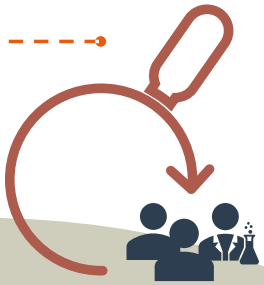
EXPERIMENTAL DATA

ANNOTATION

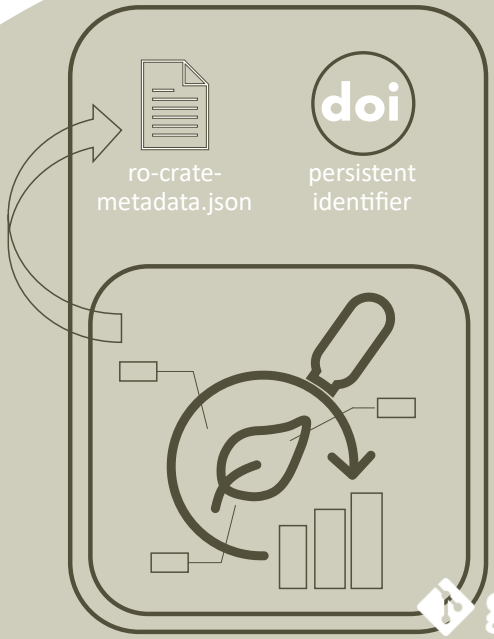
COMPUTATION



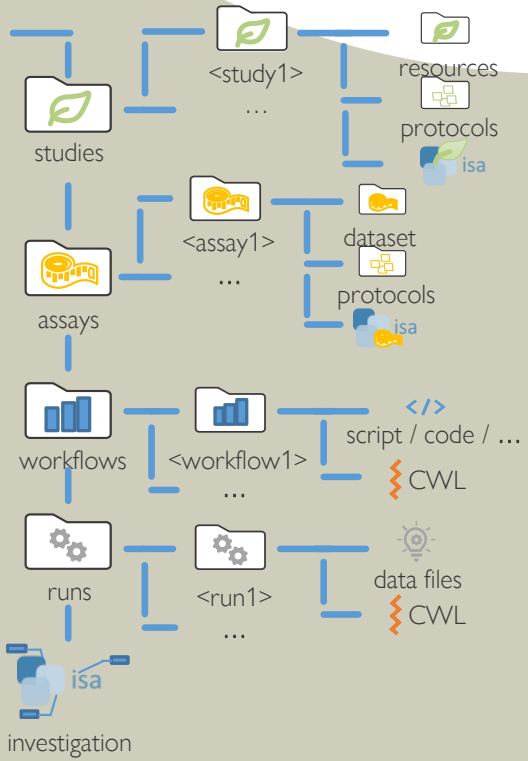
Data PLANT



https://nfdi4plants.de/



ANNOTATED RESEARCH CONTEXT (ARC)



RO metadata file is automatically generated, converting from ISA folder structure and annotations

Stored using Git LFS to support large data

isa

Investigation
Study
Assay

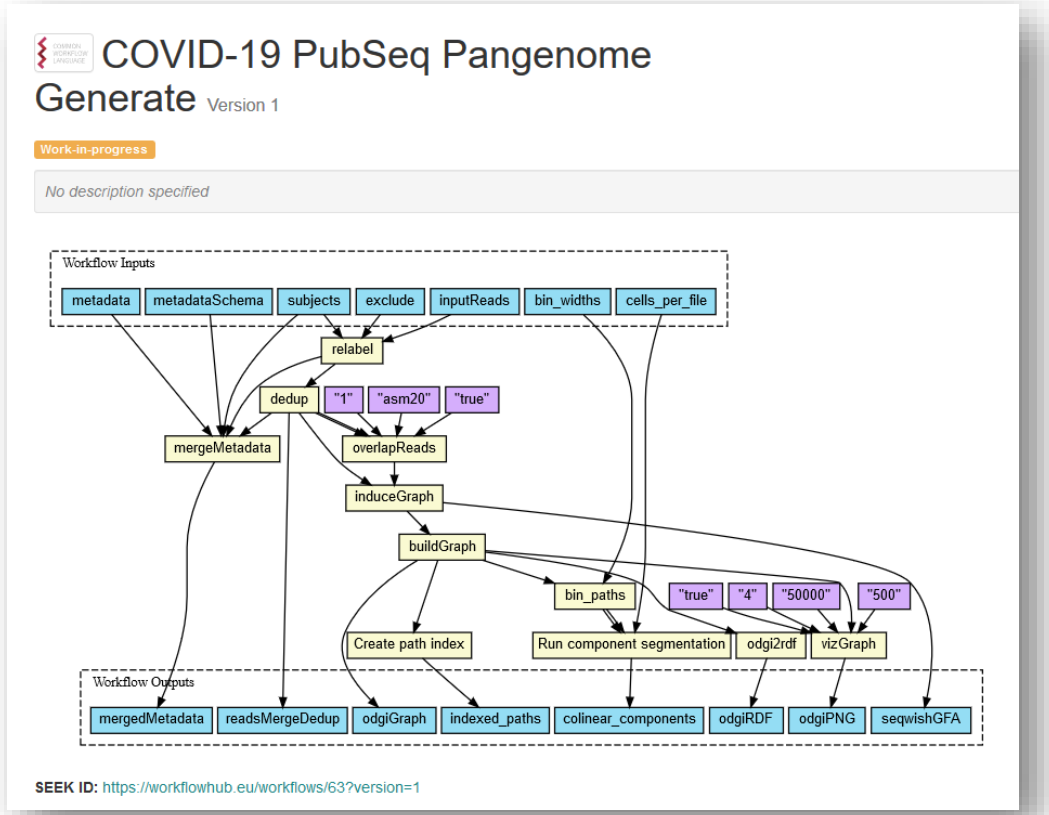
https://isa-tools.org/

Reuse established standards including ISA Model, CWL

https://nfdi4plants.de/content/learn-more/annotated-research-context.html

nfdi Nationale Forschungsdaten Infrastruktur

Computational Workflows

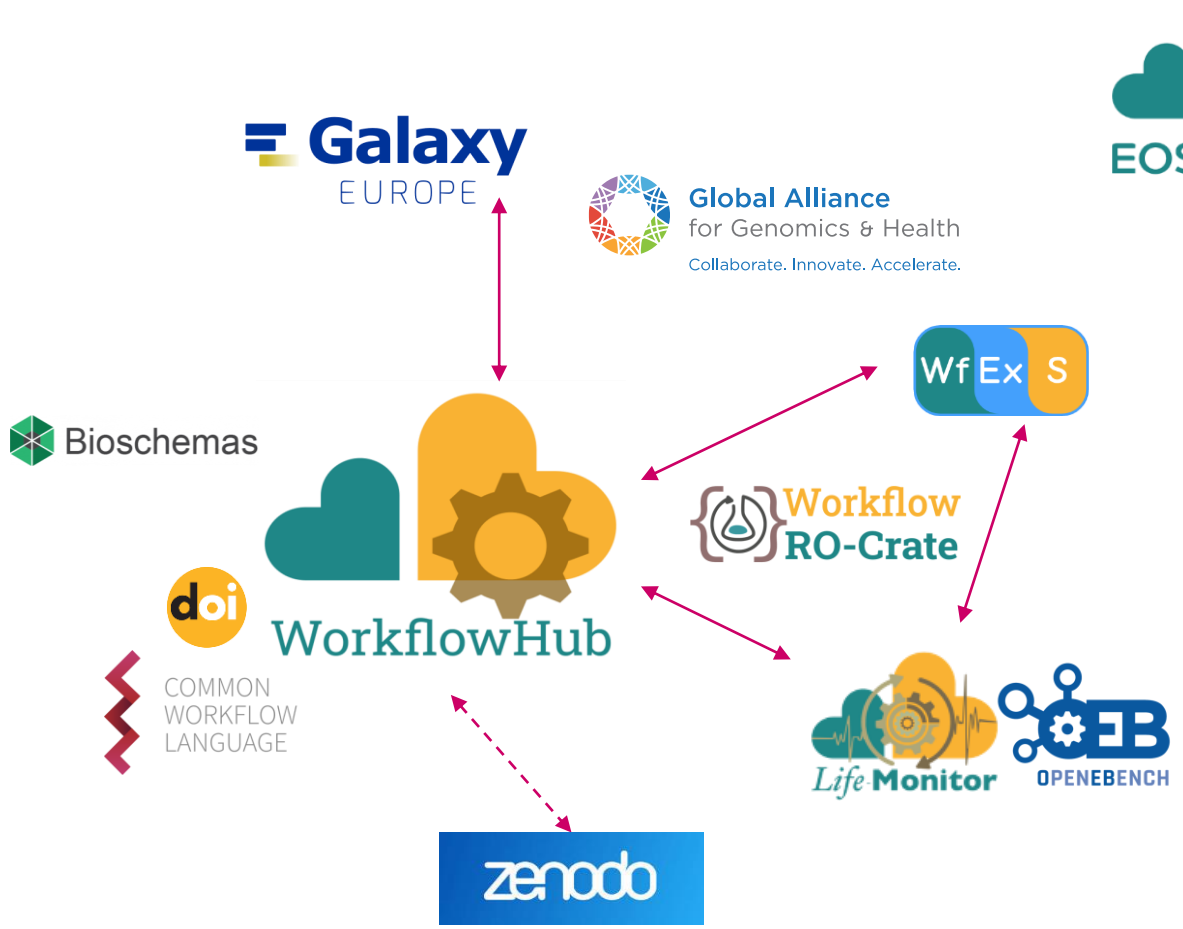


Packaging workflow files & companion objects

https://workflowhub.eu/

Building an ecosystem of FAIR Workflows

- Exchange of rich Workflow RO-Crates within an emerging ecosystem of workflow services
- Workflow Crates transfer
  - identifiers, authors, license, workflow system
  - executable workflows in their native format (e.g. Galaxy)
  - interoperable CWL description of the workflow
  - software citations (e.g. tools used)
  - required data sources
  - test suites
  - workflow execution provenance



https://workflowhub.eu/
https://www.researchobject.org/workflow-run-crate/

A gentle road towards reproducible data management by recording provenance

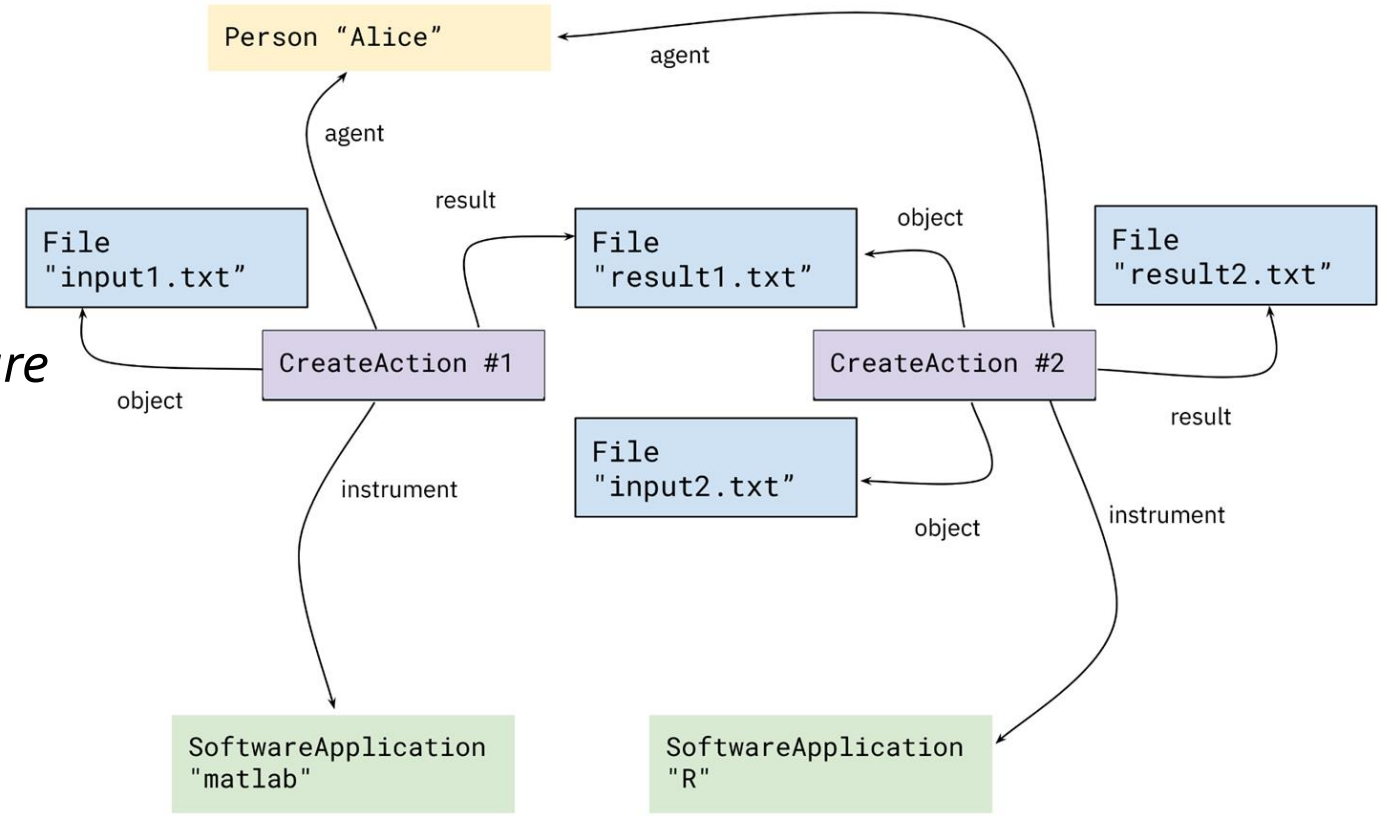
"Just enough" *provenance model* using schema.org Actions:

input1.txt is the *object* Alice used to *create* result1.txt with *instrument* matlab

No distinction between *hardware* and *software* (yet easier to open lid on software)

Provenance chain of connected actions
→ implicit *workflow:* can it be automated?

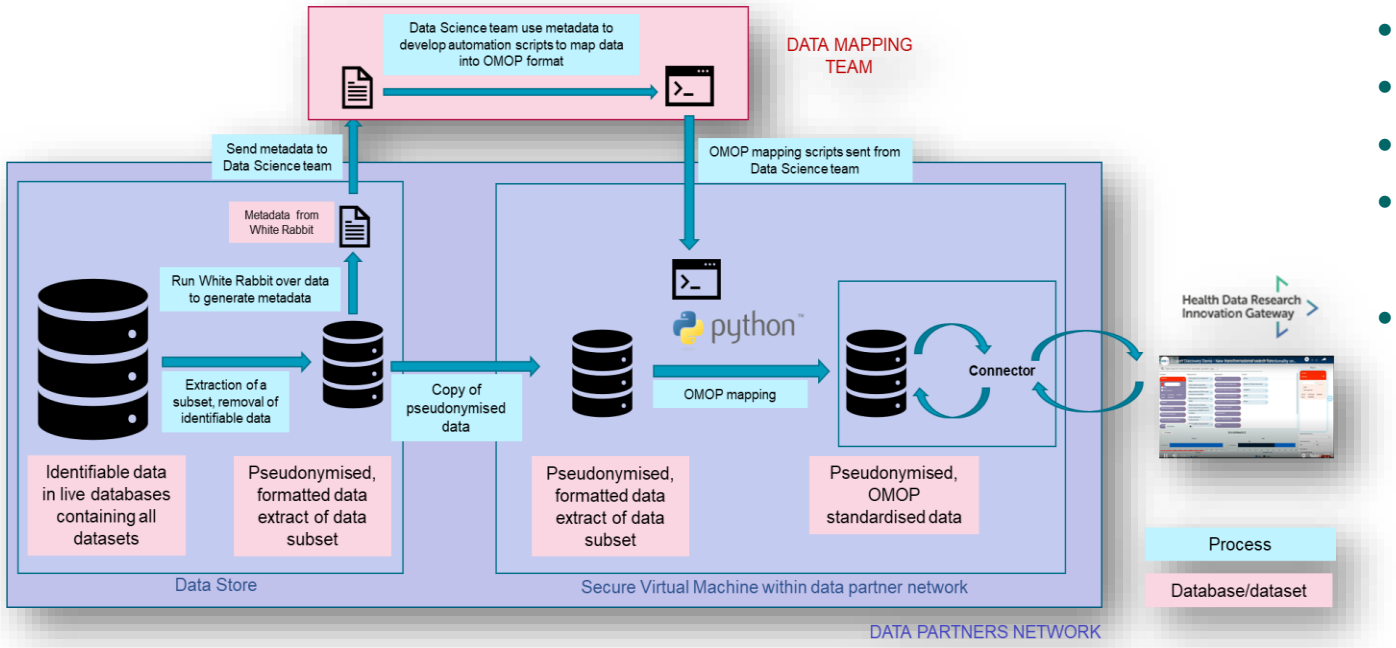
Understanding *which tools were used why* is more important than unpractical 1:1 re-run.



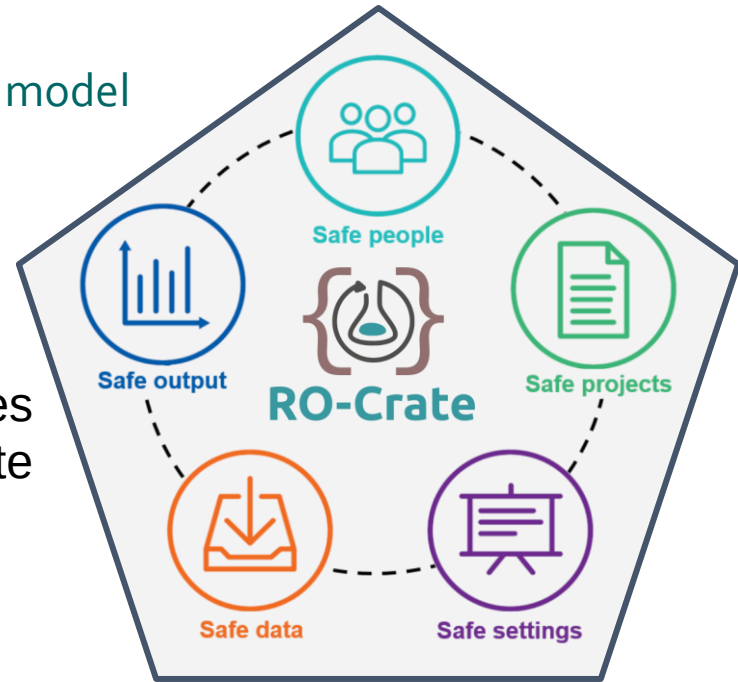
https://www.researchobject.org/ro-crate/1.1/provenance
http://www.researchobject.org/workflow-run-crate/

Federated Pipelines & Provenance Packaging

Federated analytics, distributed research pipelines in **Trusted Research Environments** for sensitive data



- Controlled access to *sensitive* data
- Exchange between data platforms
- Reporting & sharing pipelines
- Reporting results & provenance
- *Common Provenance Model* between different organizations
- Mapping to common data model



Five Safes
RO-Crate

Tom Giles, Rudolf Wittner

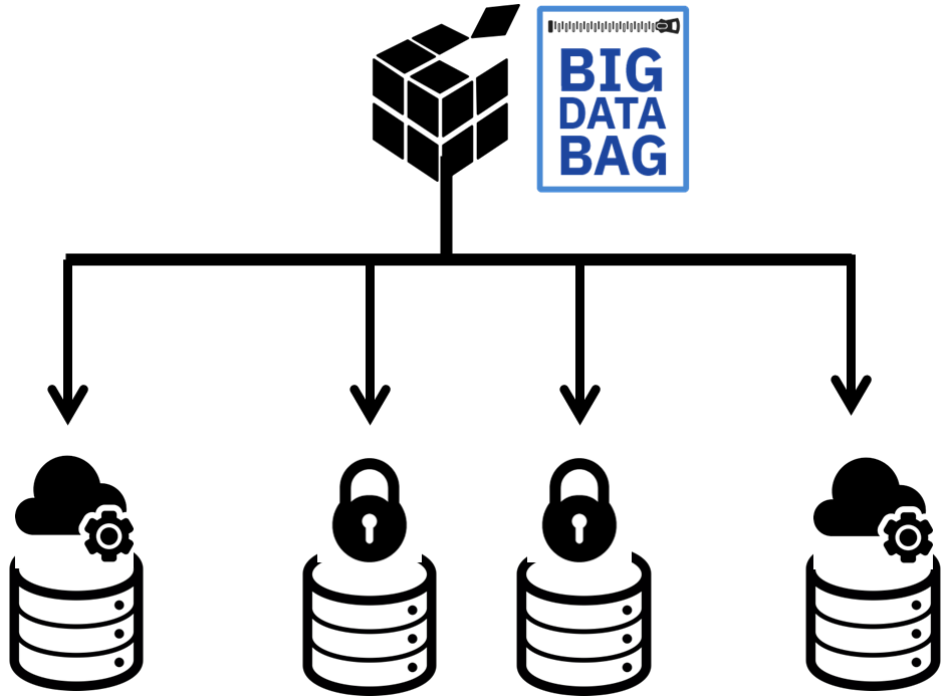
EOSC-Life BY-COVID BBMRI-ERIC HDRUK Health Data Research UK

Handling big & sensitive data

Scalable collections of references while data stays at host

Big genomic & clinical data, images etc, distributed over multiple locations.

Retain & archive processed datasets
Reference & transfer large data on demand
Controlled access
Moving data between archives



Ravi Madduri, Kyle Chard, Carl Kesselman, Ian Foster

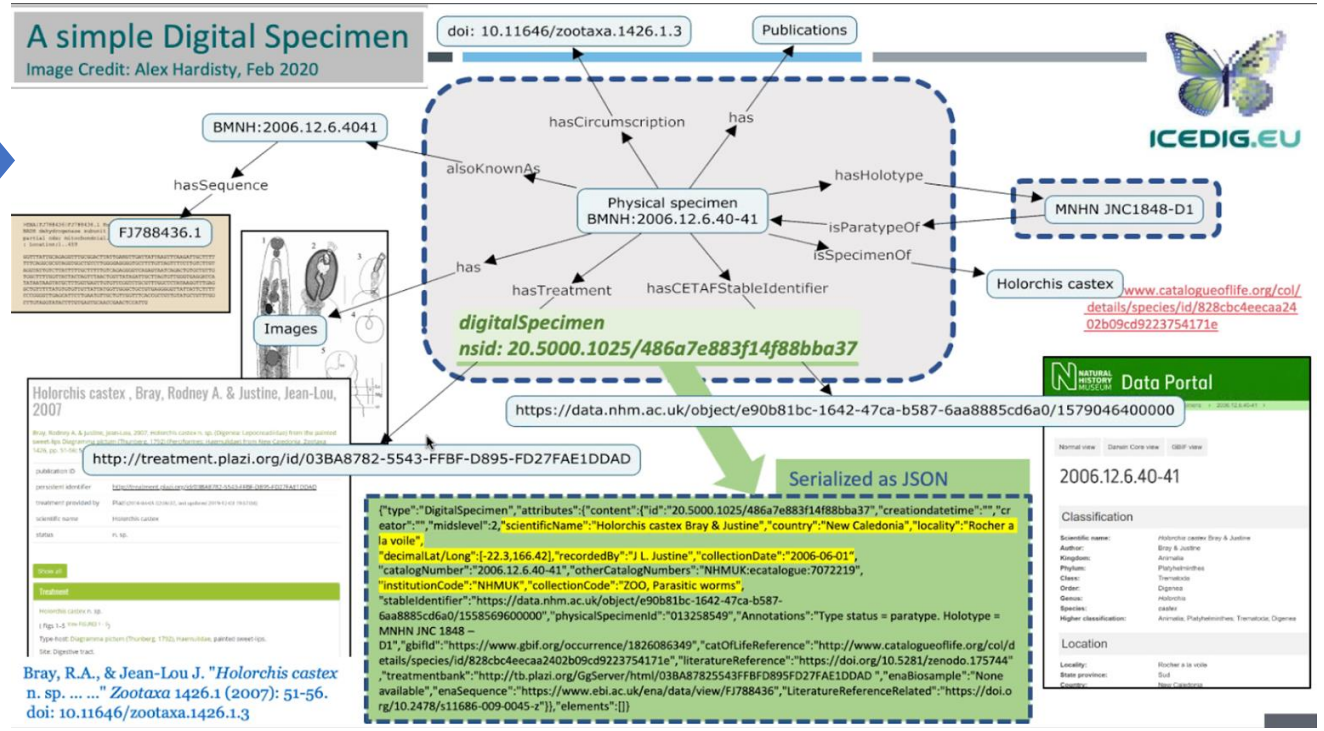
https://doi.org/10.1109/BigData.2016.7840618

BDDS

Biodiversity Digital Objects and Digital Twinning

Digital Surrogate FAIR Digital Object

Bags of references



SYNTHESYS+ Synthesis of Systematic Resources a DiSSCo project

BIODT biodiversitydigitaltwin

predicting biodiversity dynamics

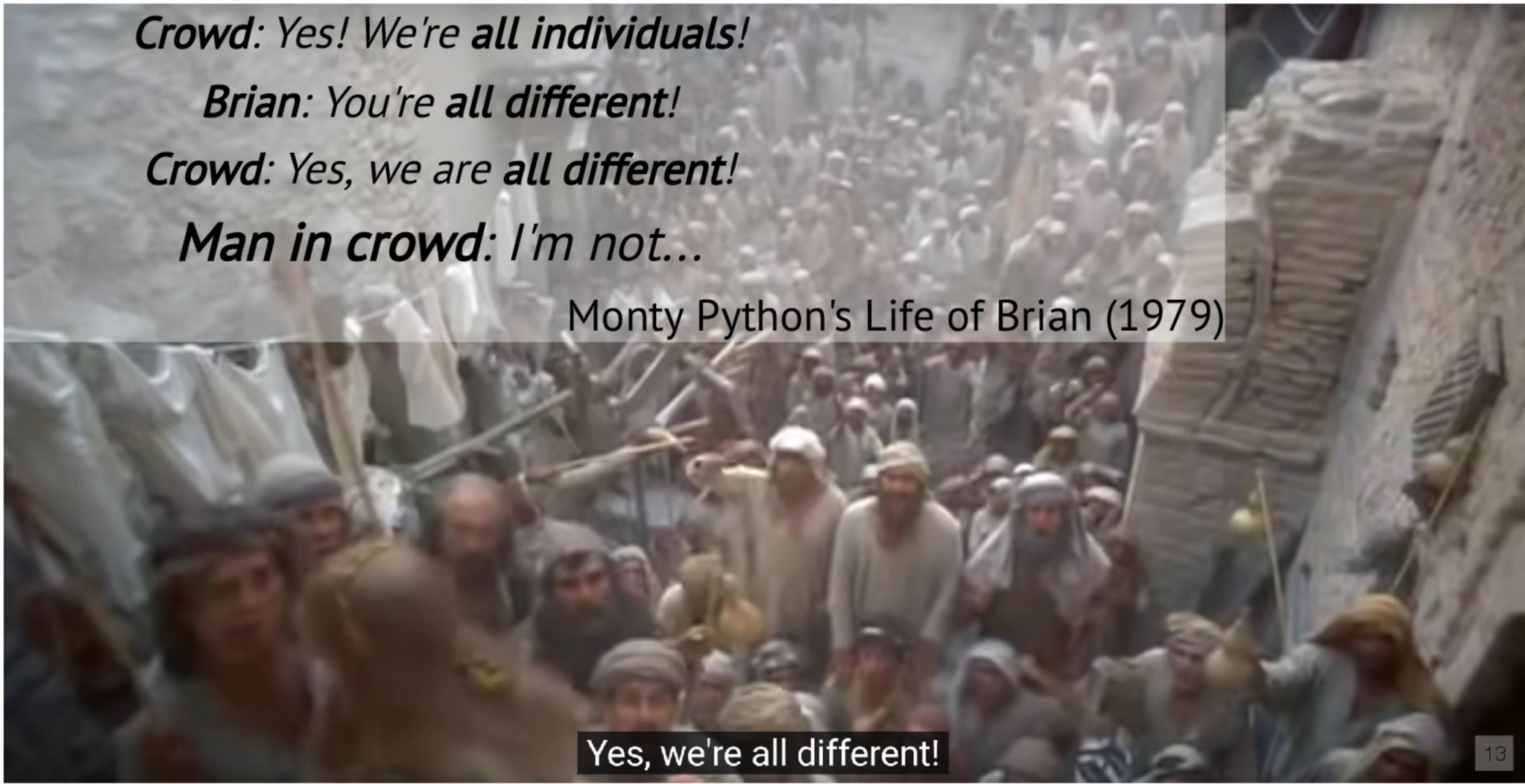
courtesy of Alex Hardisty, Dimitris Koureas

Hardisty 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**.** *Data Intelligence* **4**(2): 320–341. https://doi.org/10.1162/dint_a_00134

https://biodt.eu/

Importance of profiles

> ***Brian**: Look, you've got it all wrong! You don't need to follow me. You don't need to follow anybody! You've got to think for yourselves! You're **all individuals**!*

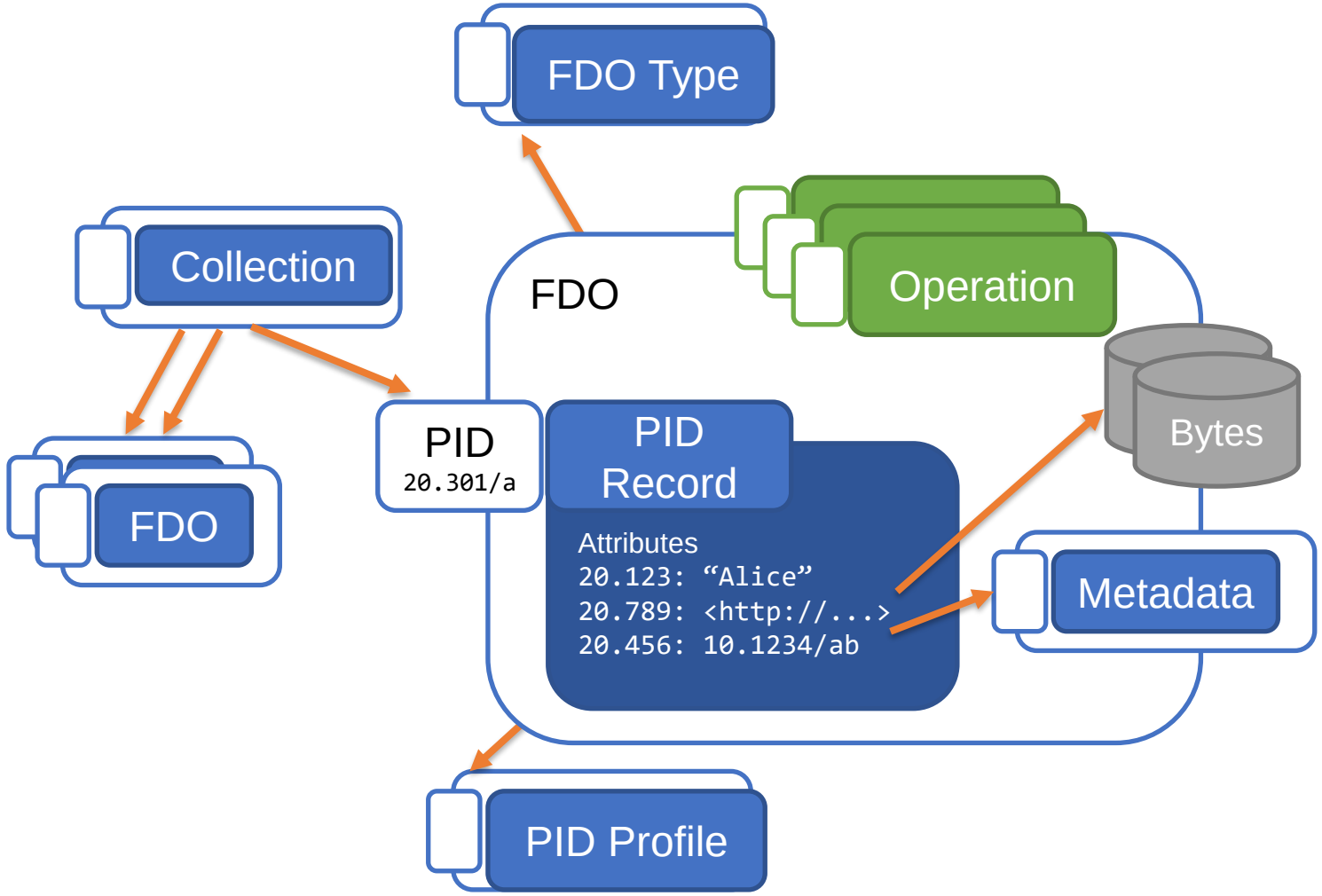


FAIR Digital Object (FDO) – conceptual view

Predictable implementation of FAIR for **active objects -** not just static data

- Distributed architecture
- **Self-describing** digital objects
- Several types of **metadata**
- Encapsulation of **operations**

RO-Crate implements FDO with current web stack



Soiland-Reyes, Sefton, et al (2022): **Creating lightweight FAIR Digital Objects with RO-Crate**. *Research Ideas and Outcomes*, 1st Intl Conf on FAIR Digital Objects

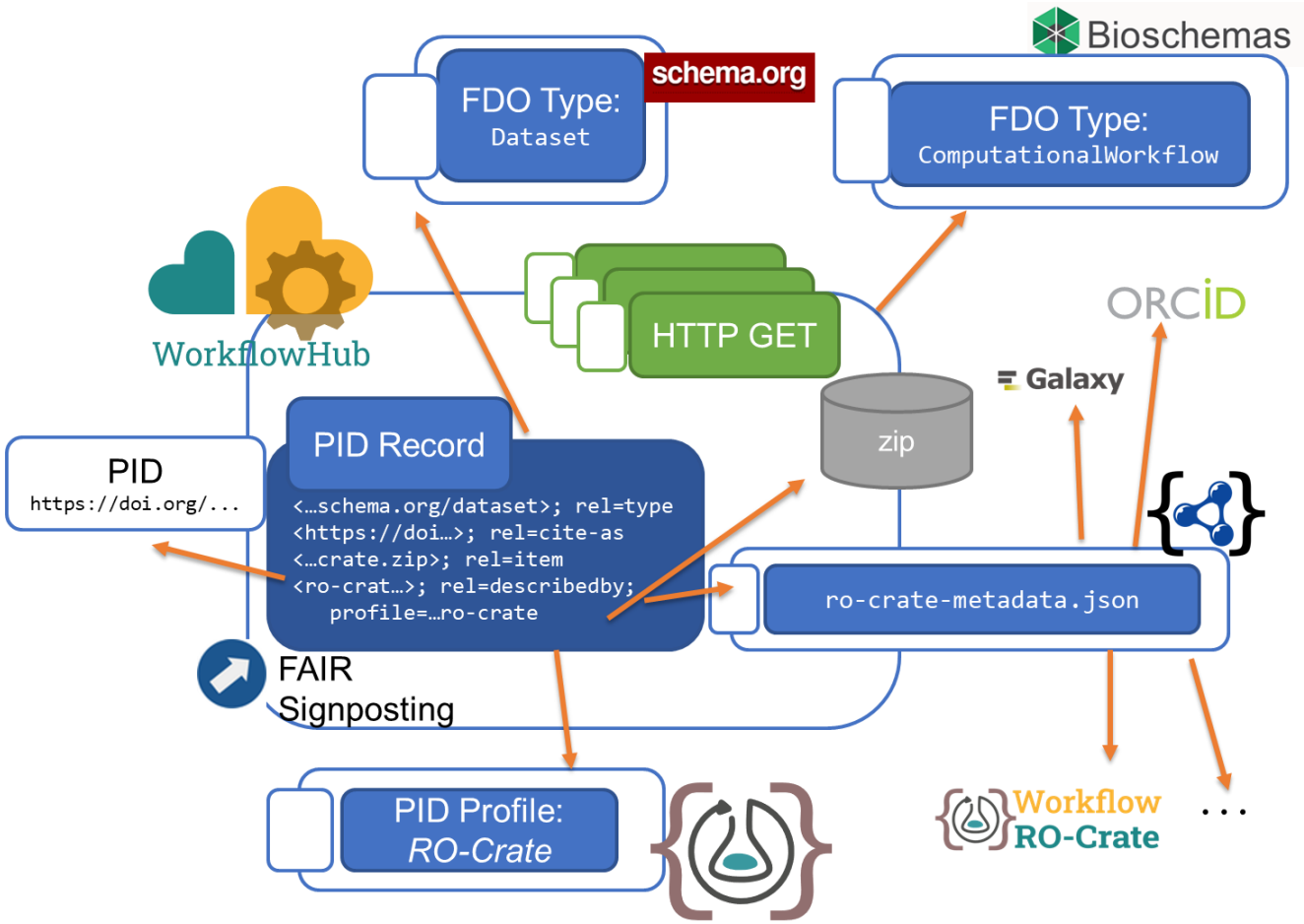
FAIR Digital Objects for Science: From Data Pieces to Actionable Knowledge Units: https://doi.org/10.3390/publications8020021

FAIR RO-Crates as a realization of FDOs
A bunch of papers/posters at FDO2022

Implementing FDO web with current web standards

FAIR Signposting – to look up RO-Crate describe object

- Types from schema.org
- Metadata files
- Directives to profiles the FDO conforms to
- Community APIs (like GA4GH)



https://signposting.org/FAIR/

Soiland-Reyes, Sefton, et al (2022): **Creating lightweight FAIR Digital Objects with RO-Crate**. *Research Ideas and Outcomes*, 1st Intl Conf on FAIR Digital Objects

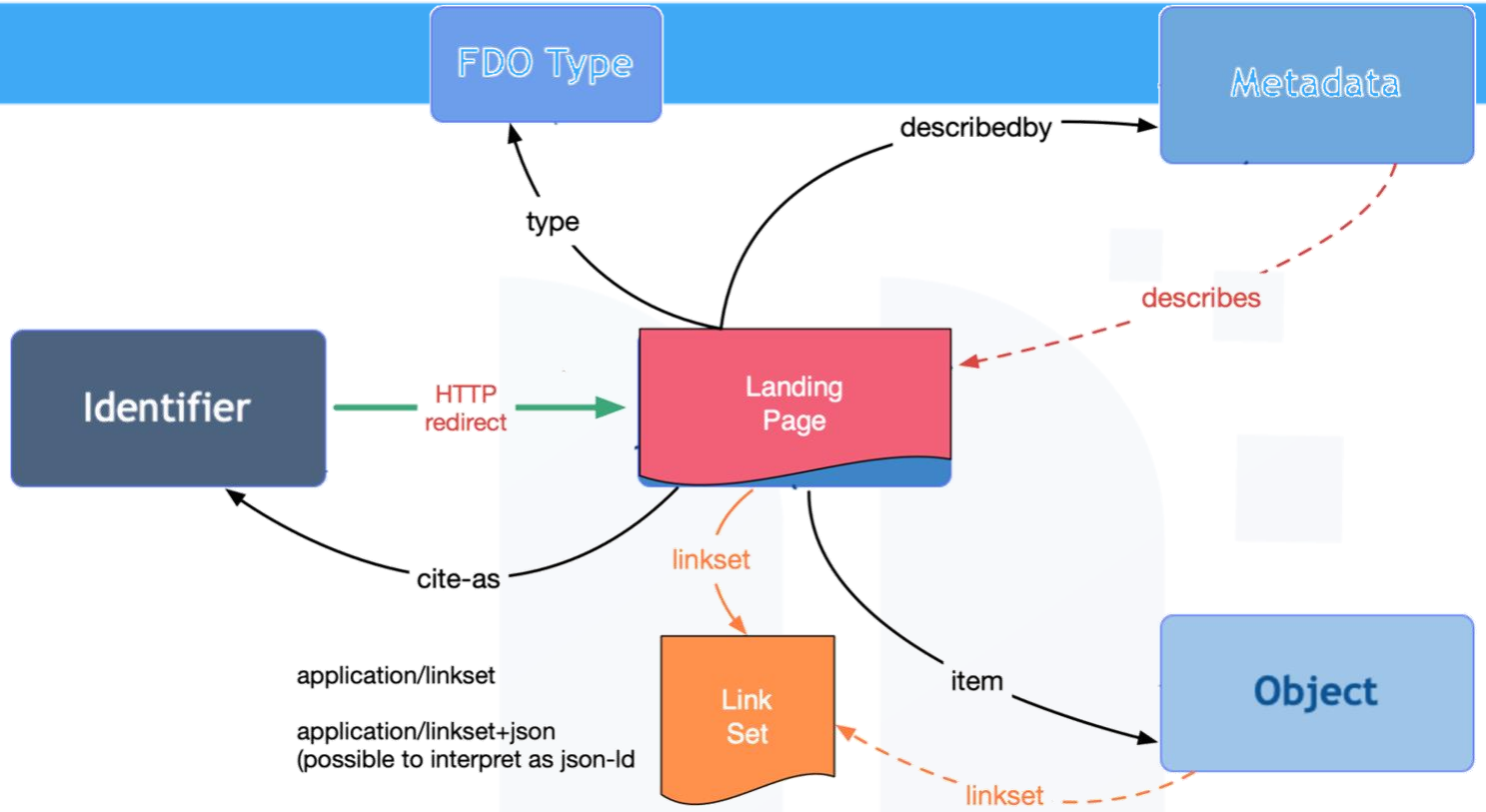
FAIR Signposting

1.3. Typed Links in the FAIR Signposting Profile

The Relation Types that are used for the FAIR Signposting Profile as a means to meaningfully interlink resources that represent a scholarly artifact on the web are shown in the below table. The general description of their meaning is based on the more formal language used in the specification that define them. Their specific use for the FAIR Signposting Profile is provided in the descriptions of Level 1 and Level 2, below.

Relation Type	Description
`author`	The target of the link is a URI for an author of the resource that is the origin of the link.
`cite-as`	The target of the link is a persistent URI for the resource that is the origin of the link.
`describedby`	The target of the link provides metadata that describes the resource that is the origin of the link.
`type`	The target of the link is the URI for a class of resources to which the resource that is the origin of the link belongs.
`license`	The target of the link is the URI of a license that applies to the resource that is the origin of the link.
`item`	The origin of the link is a collection of resources and the target of the link is a resource that belongs to that collection

https://signposting.org/FAIR/



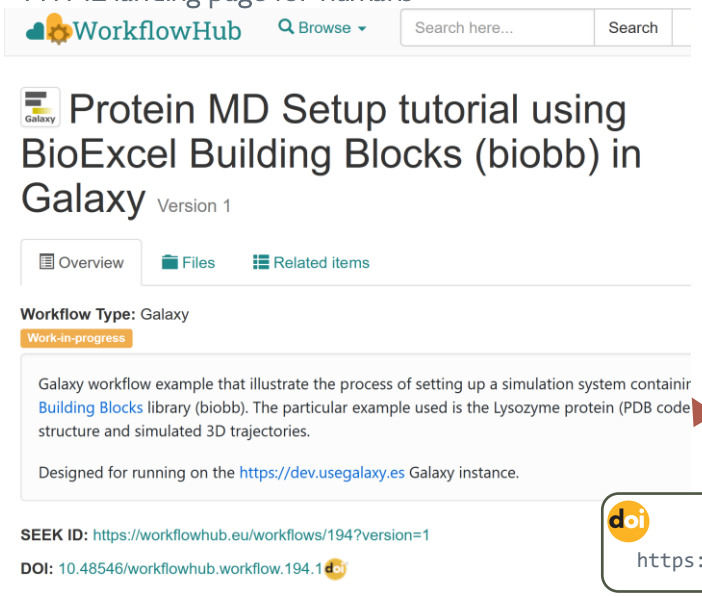

```
(a2a) stain@xena11:~$ signposting https://doi.org/10.48546/workflowhub.workflow.255.1
Signposting for https://workflowhub.eu/workflows/255?version=1
CiteAs: <https://doi.org/10.48546/workflowhub.workflow.255.1>
DescribedBy: <https://workflowhub.eu/workflows/255?version=1> application/ld+json
             <https://workflowhub.eu/workflows/255?version=1> application/vnd.datacite.datacite+xml
Item: <https://workflowhub.eu/workflows/255/ro_crate?version=1> application/zip
(a2a) stain@xena11:~$
```

https://pypi.org/project/signposting/

Resolving RO-Crate FDOs using FAIR Signposting

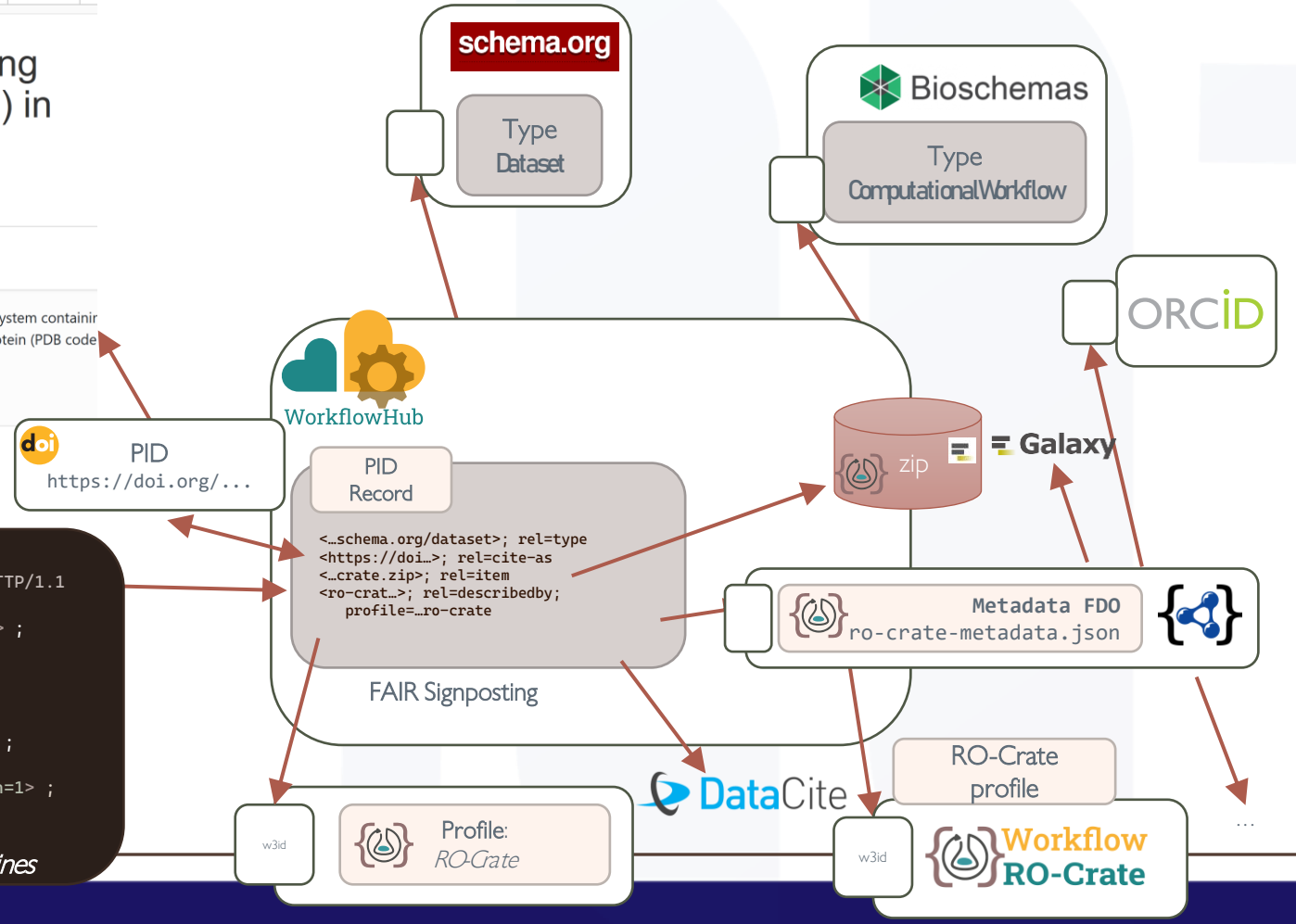
https://signposting.org/FAIR/

HTML landing page for *humans*




```
HEAD https://workflowhub.eu/workflows/255?version=1 HTTP/1.1
…
Link: <https://workflowhub.eu/workflows/255?version=1> ;
  rel="describedby";
  type="application/vnd.datacite.datacite+xml",
<https://workflowhub.eu/workflows/255?version=1> ;
  rel="describedby" ; type="application/ld+json",
<https://doi.org/10.48546/workflowhub.workflow.255.1> ;
  rel="cite-as",
<https://workflowhub.eu/workflows/255/ro_crate?version=1> ;
  rel="item" ; type="application/zip" ;
  profile="https://w3id.org/ro/crate"
```

HTTP Link headers for *machines*



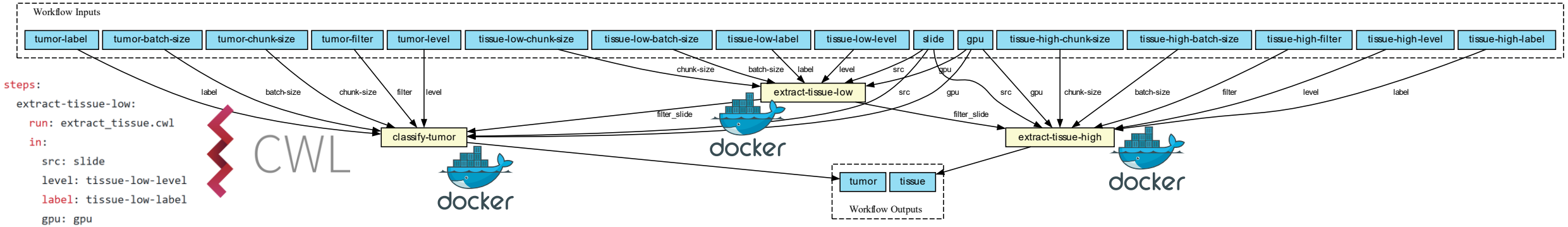
Machine learning pipeline #1

Download: training_1.mrxs

@id	input/training_1.mrxs
name [?]	training_1.mrxs
@type	File
encodingFormat [?]	MIRAX Virtual Slide Format
dateModified [?]	2021-10-18T09:30:56.515Z
mentions [?]	• arbitrary-file-A • arbitrary-file-B
Items that reference this one	
result [?]	Scan of sample
object [?]	Example run of pipeline
hasPart [?]	Example pipeline inputs

https://www.researchobject.org/workflow-run-crate/examples/draft/ml-pipeline/

Machine learning pipeline #2




```yaml
steps:
  extract-tissue-low:
    run: extract_tissue.cwl
    in:
      src: slide
      level: tissue-low-level
      label: tissue-low-label
      gpu: gpu
      chunk-size: tissue-low-chunk-size
      batch-size: tissue-low-batch-size
    out: [tissue]

  extract-tissue-high:
    run: extract_tissue.cwl
    in:
      src: slide
      level: tissue-high-level
      label: tissue-high-label
      filter_slide: extract-tissue-low/tissue
      filter: tissue-high-filter
      gpu: gpu
      chunk-size: tissue-high-chunk-size
      batch-size: tissue-high-batch-size
    out: [tissue]

  classify-tumor:
    run: classify_tumor.cwl
    in:
      src: slide
      level: tumor-level
      label: tumor-label
      filter_slide: extract-tissue-low/tissue
      filter: tumor-filter
      gpu: gpu
      chunk-size: tumor-chunk-size
      batch-size: tumor-batch-size
```

```yaml
cwlVersion: v1.1
class: CommandLineTool
requirements:
  InlineJavascriptRequirement: {}
  DockerRequirement:
    dockerPull: crs4/slaid:1.1.0-beta.25-tumor_model-level_1-v2.2-cudnn
  InitialWorkDirRequirement:
    listing:
      - $(inputs.src)
```

https://github.com/crs4/deephealth-pipelines
https://cwl-airflow.readthedocs.io/

Ten simple rules for making a software tool workflow-ready

Rule 1: Make sure a workflow engine can talk to your software easily
Rule 2: Make your tool simple to install
Rule 3: Document your tool
Rule 4: Make your tool maintainable
Rule 5: Follow the principle of least surprise
Rule 6: Make your tool parallelizable
Rule 7: Make your workflow tool a good citizen
Rule 8: Make output reproducible
Rule 9: Carefully consider human interaction
Rule 10: A software tool should just do one thing

https://doi.org/10.1371/journal.pcbi.1009823

Machine learning pipeline #2 as RO-Crate

Promort prediction run on 2021-06-29T16:07:14.873427+00:00

@id	#b5aa3374-235a-4741-b20d-99434d5b046d
name [?]	Promort prediction run on 2021-06-29T16:07:14.873427+00:00
@type	CreateAction
endTime [?]	2021-06-29T16:09:45.814390+00:00
instrument [?]	Promort tissue and tumor prediction
object [?]	• http://slide-repository:5000/slides/DHT00001-1.mrxs • mode: serial • tissue-low-level: 9 • tissue-low-label: tissue_low • tissue-high-level: 8 • tissue-low-chunk: 256 • tissue-high-label: tissue_high • tissue-high-filter: tissue_low>1 • tissue-high-chunk: 1536 • tumor-chunk: 1536 • gpu: 0 • tumor-level: 1 • tumor-label: tumor • tumor-filter: tissue_low>1
result [?]	• tissue_high.zip • tumor.zip
startTime [?]	2021-06-29T16:07:14.873427+00:00

https://www.researchobject.org/workflow-run-crate/examples/draft/ml-predict-pipeline/

Considerations for RO-Crates with Machine learning pipelines

- Ensure pipeline steps can run in a **container** (e.g. Singularity) or Conda
- Make training a **separate step** from evaluation and use
  - Trained **model** (e.g. network weights) as a FAIR object in itself
- Which **data sources** was the model trained on? Is the data *sensitive*?
  - Persistent identifiers, provenance, attribution
- **GPU** support in workflow engines are still at its infancy
  - Can block interoperability and even scalability!
- Capturing **parameters** are important for understanding
- **Updating** code of steps can make pipeline fragile
  - Follow best practices for research software and **versioning**

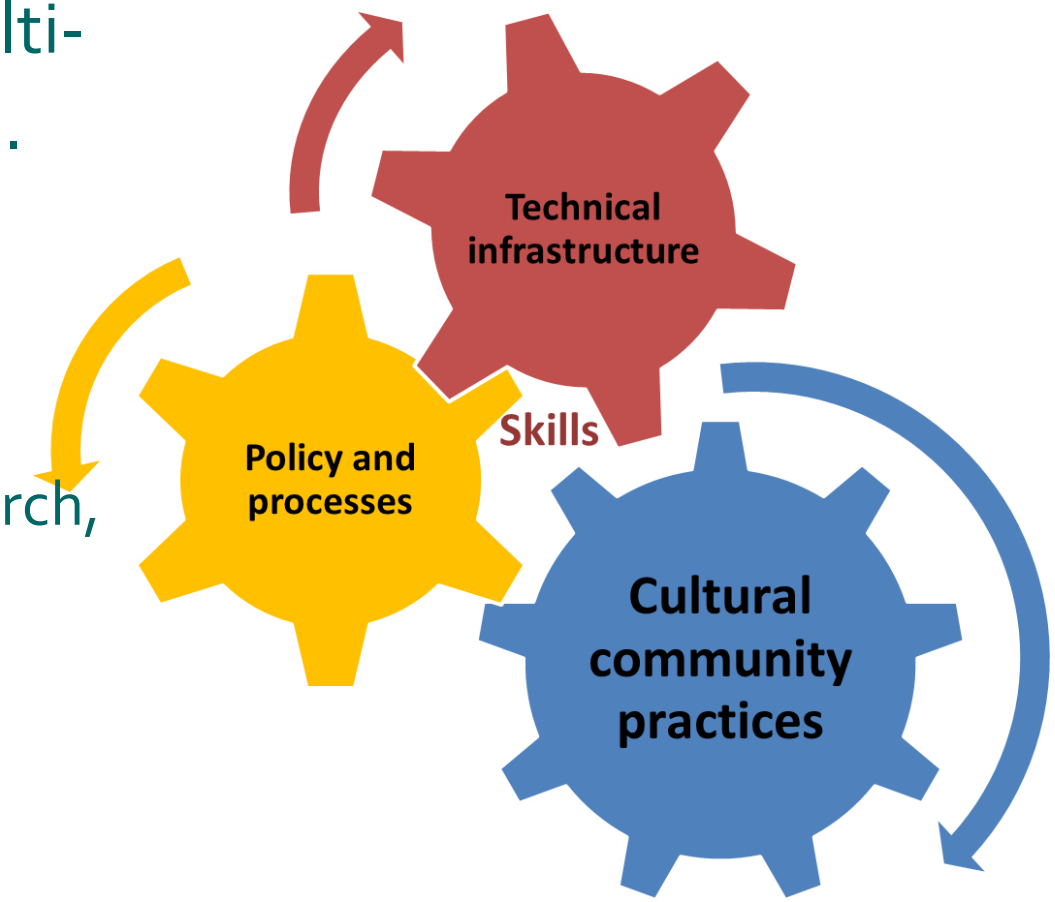
Metadata is a love note to the FAIR future….

….RO-Crate is the delivery package in a multi-platform, mixed object research ecosystem.

Keep it practical, real and simple

Adoption and diversity friendliness

Metadata middleware to drive the release of research, reproducible scholarship and knowledge graphs.



It takes a village

To make Research Objects normative

Promote to researchers but …… target Research Infrastructures to deliver

Acknowledgements

RO-Crate community

- 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)
- 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
- Marc Portier https://orcid.org/0000-0002-9648-6484
- Paul Brack https://orcid.org/0000-0002-5432-2748
- Milan Ojsteršek https://orcid.org/0000-0003-1743-8300
- Bert Droesbeke https://orcid.org/0000-0003-0522-5674
- Chenxu Niu https://orcid.org/0000-0002-2142-1731
- Kosuke Tanabe https://orcid.org/0000-0002-9986-7223
- Tomasz Miksa https://orcid.org/0000-0002-4929-7875
- Marco La Rosa https://orcid.org/0000-0001-5383-6993
- Cedric Decruw https://orcid.org/0000-0001-6387-5988
- Andreas Czerniak https://orcid.org/0000-0003-3883-4169
- Jeremy Jay https://orcid.org/0000-0002-5761-7533
- Sergio Serra https://orcid.org/0000-0002-0792-8157
- Ronald Siebes https://orcid.org/0000-0001-8772-7904
- Shaun de Witt https://orcid.org/0000-0003-4196-3658
- Shady El Damaty https://orcid.org/0000-0002-2318-4477
- Douglas Lowe https://orcid.org/0000-0002-1248-3594
- Xuanqi Li https://orcid.org/0000-0003-1498-6205
- Sveinung Gundersen https://orcid.org/0000-0001-9888-7954
- Muhammad Radifar https://orcid.org/0000-0001-9156-9478
- Rudolf Wittner https://orcid.org/0000-0002-0003-2024
- Oliver Woolland https://orcid.org/0000-0002-4565-9760
- Paul De Geest https://orcid.org/0000-0002-8940-4946
- Douglas Fils https://orcid.org/0000-0002-2257-9127
- Florian Wetzels https://orcid.org/0000-0002-5526-7138
- Raül Sirvent https://orcid.org/0000-0003-0606-2512
- Abigail Miller https://orcid.org/0000-0001-9228-2882
- Jake Emerson https://orcid.org/0000-0003-0617-9219
- Davide Fucci https://orcid.org/0000-0002-0679-4361
- Bruno P. Kinoshita https://orcid.org/0000-0001-8250-4074

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