27-29 May 2024

International conference on Marine Data and Information Systems

SeaDataNet AISBL · Havforskningsinstituttet · Ifremer · MARIS · MEDIN · National Oceanography Centre · INGV · iMagine · ICDE · eosc Blue-Cloud2026

eosc | Blue-Cloud2026

Workflows for marine metadata and data management

IRD, FORTH, CNR, FIRMS network (FAO & tuna-RFMOs)

Julien Barde, MARBEC, Univ Montpellier, CNRS, Ifremer, IRD, Montpellier, France, julien.barde@ird.fr
Emmanuel Blondel, UN-FAO, Fisheries and Aquaculture Division, Rome, Italy, emmanuel.blondel@fao.org
Bastien Grasset, MARBEC, Univ Montpellier, CNRS, Ifremer, IRD, Réunion, bastien.grasset@ird.fr

Funded by the European Union

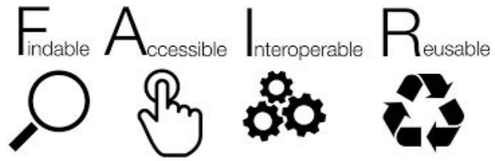
The Blue-Cloud 2026 project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101094227.

EOSC, Blue-Cloud 2026 project: VRE and Vlabs

- **General context**
  - **EOSC - Blue-Cloud 2026 HORIZON project**, VRE & VLabs, **VLab 5 – "Global Fisheries Atlas"** focusing on **fisheries data:**
    - past european research projects: FP7 iMarine, H2020 BlueBridge, H2020 OpenAire-Connect, H2020 Blue-Cloud
    - FIRMS network (FAO, research bodies and RFB: WECAFC, RECOFI, tuna-RFMOs..)
  - same approach with other domains / data types: FAO, french research bodies (IRD, INRAE, CNRS..)..

- **Workflows** for **FAIRification** of spatial (meta-)data
  - implement FAIR DMPs with widely used tools: e.g. OGC standards, SDIs & R software..
    - **R packages** since few years (mainly with Java and Python until now)
  - automate redundant data management tasks / minimize duplication of efforts
    - read, write, publish spatial (meta-)data (e.g. in GeoNetwork, GeoNode..)
    - mapping standard (e.g. CF conventions to ISO 19115)
    - assign DOIs, ensure reproducibility..

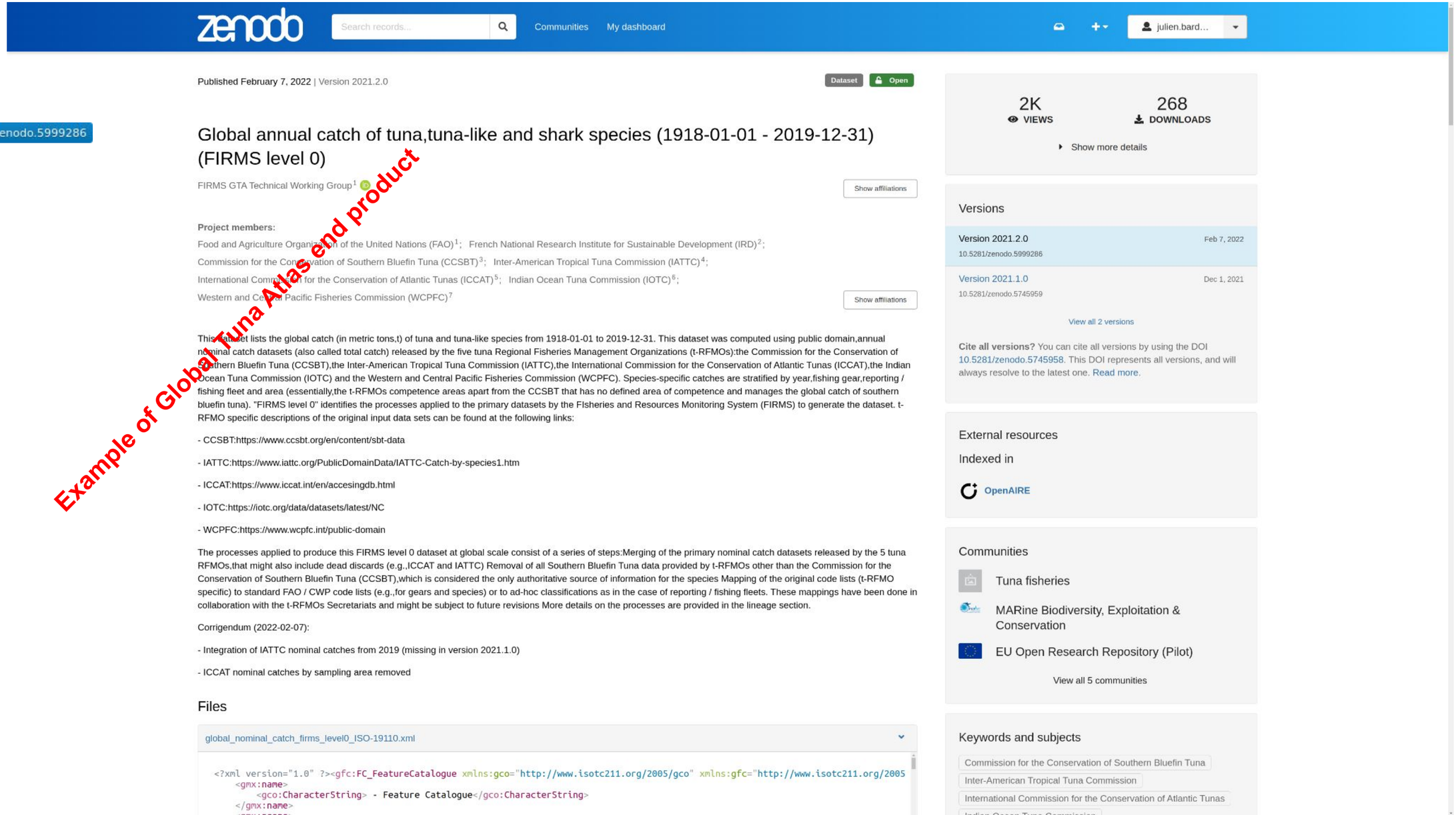
FAIRification: workflows for what data ?



We identified a set of key use cases in the marine domain using main data types and formats:

- **fisheries data** (our pilot use case): e.g. FIRMS **Global Tuna Atlas** (CSV + Postgis DB)
    - nominal catch (per year / management area)
    - gridded data (catches, efforts, size classes..)
    - tagging data..
- **vector** data:
    - flat files in usual vector formats: csv, gpkg, shp, GML..
    - any spatial relation (table or view) stored in a spatial relational databases (SQL + SFS)
- **raster** data:
    - usual raster formats: geotiff / COG, NetCD-CF..
    - e.g. drone: raw data and photogrammetric process results (orthophotos, DEM..)
- **sensors** data: any data stored in a NetCDF-CF (model outputs, satellites, in situ sensors..)

FAIRification end product, DOI as a cornerstone: e.g. [FIRMS Global Tuna Atlas]



FAIRification end product: e.g. FIRMS Global Tuna Atlas

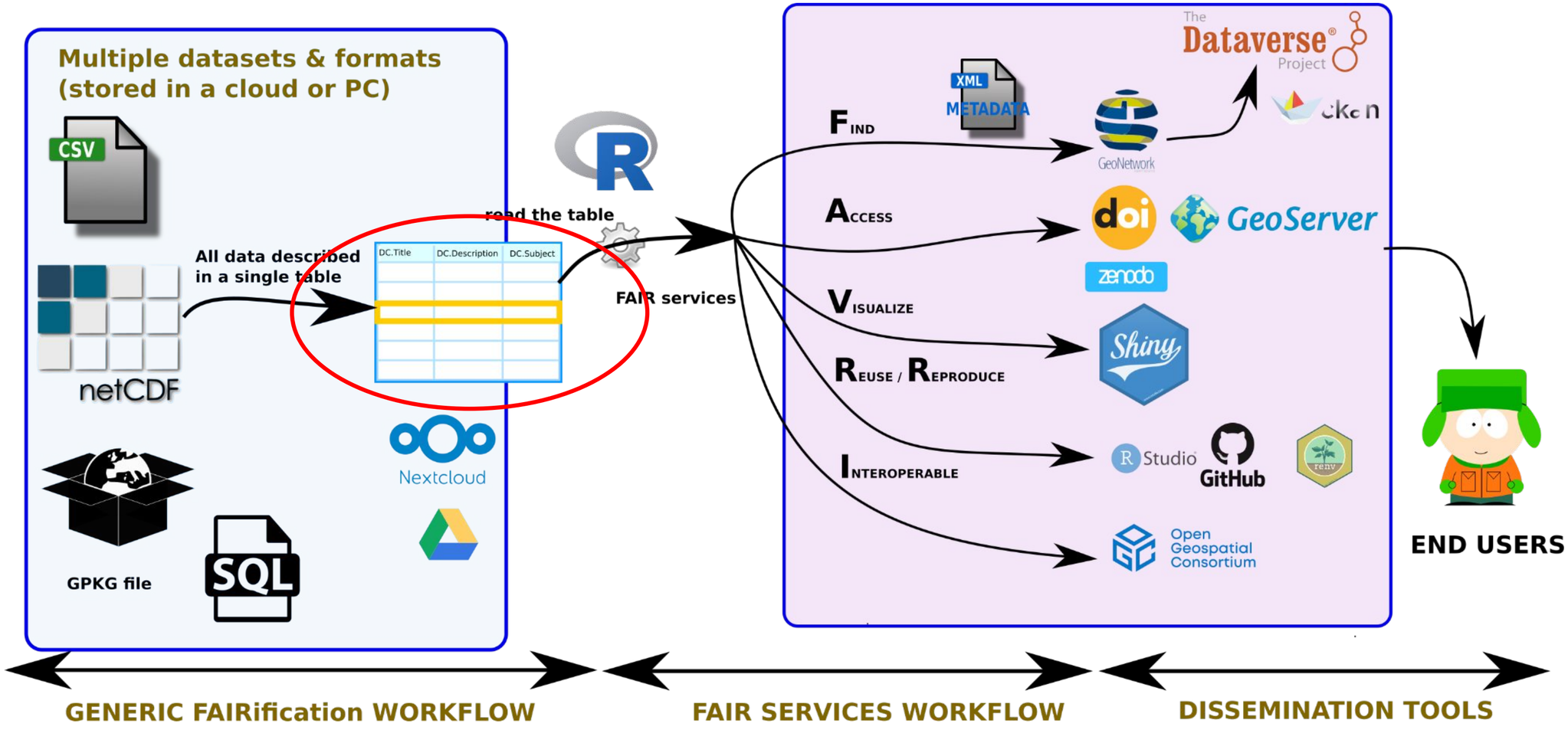
Interest of workflows to simplify metadata creation & publication processes

Workflows to automate the **main data management steps** (by using R scripts and packages):

- **Read** & **write** various **inputs** & **outputs:** by using the main standards for (meta-)data **formats** and access **protocols**, e.g.:
  - Spatial OGC standards:
    - metadata: 19115 / 19139 (discovery), 19110 (usage), 19119..
    - data: SQL / SFS, gpkg (shp..), CF compliant NetCDF / NCML..
    - protocols: CSW, WMS / WFS / WCS, OPeNDAP..
  - Biodiversity Information Standards (TDWG): e.g. EML, Darwin Core
  - Ad hoc APIs: Zenodo, GeoNetwork, GeoServer, Dataverse, Nextcloud (OCS) ..
- **Transform**: metadata standards mapping, data structure harmonization
- **Load / publish**: (meta-)data in widely used tools (eg GeoNetwork, GeoServer, Zenodo..)
- **Long term storage and versioning**: DOIs assignment by implementing DataCite schema (e.g. in Zenodo data repository)

A method for editing metadata through basic tools : general overview

Simple tools and ergonomics (no endless html forms): data managers and users collaborate to fill **a single table** (e.g CSV) using **basic GUIs** (spreadsheets in a cloud or PC)



A pivot model to comply with widely used metadata standards: structure

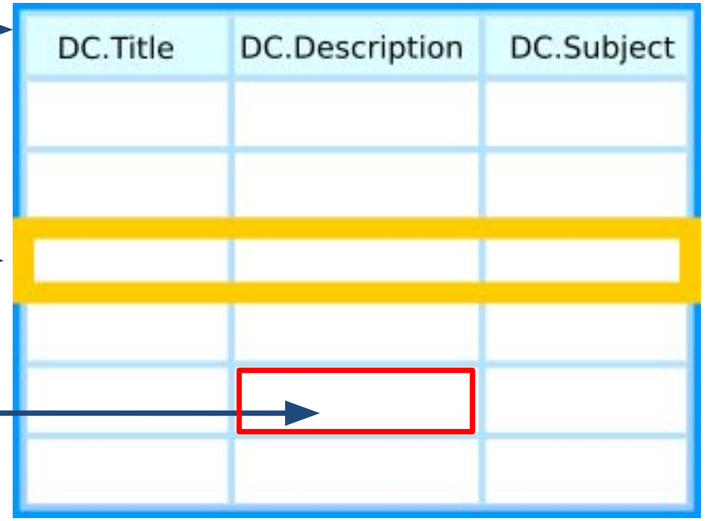
geoflow R package implements a simple metadata model made of:

- **15 Dublin Core (DCMI) metadata elements**: used as columns names in our tabular format
  - DCMI is a domain agnostic standard providing a list of key metadata elements
- **additional metadata elements to comply with other standards** (optional)**:** directly named in the cells of the columns by using simple syntactic rules (key-value pairs, KVP)
  - **OGC** (spatial)
  - **EML** & **Darwin Core** (spatial)
  - **Datacite** (global and cross domain dissemination and long term access through DOIs)

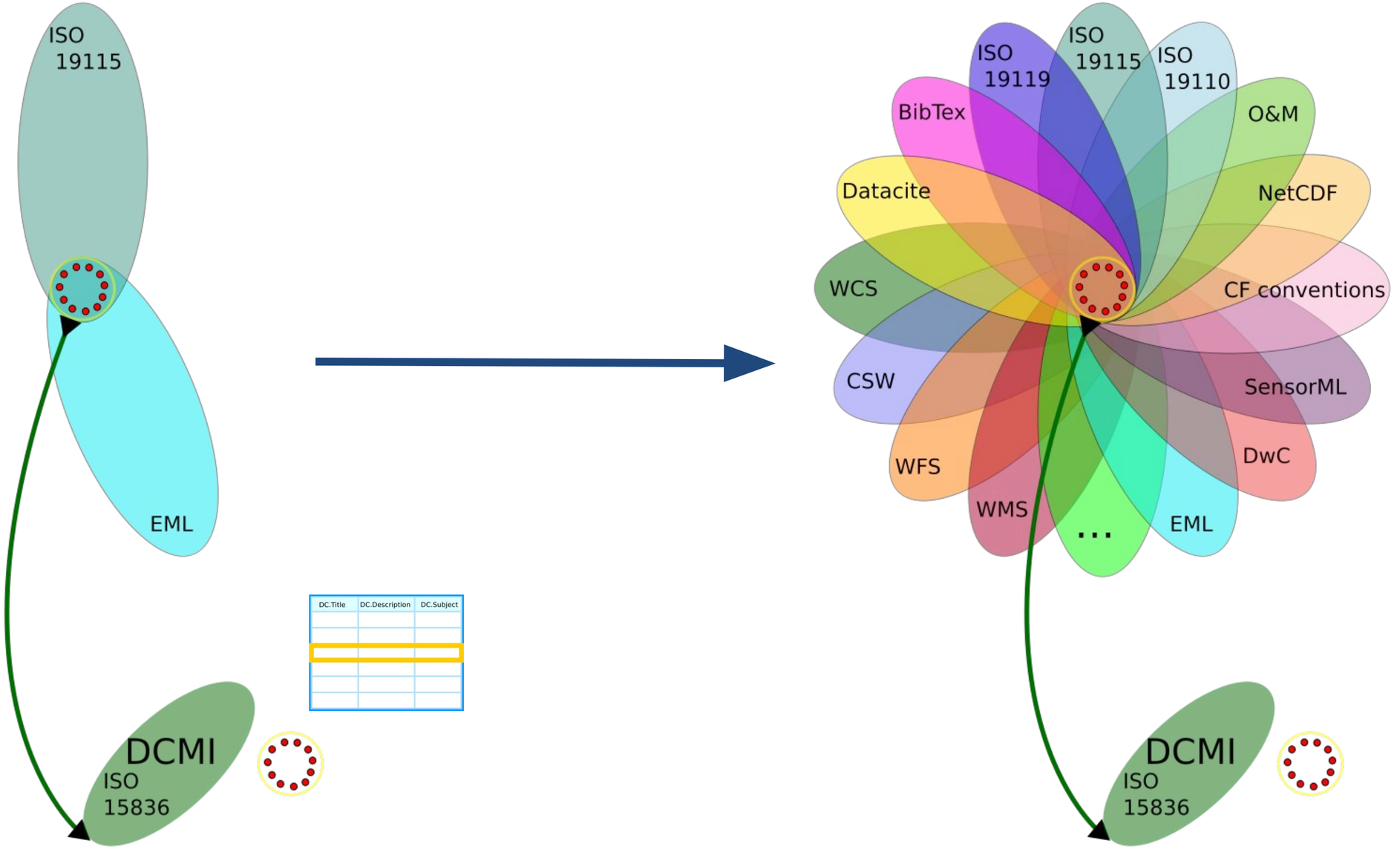
one **column** = one DCMI metadata element

one **row** = one description of a dataset

one **cell** = 1..* sub-metadata elements
(required by OGC, EML, Datacite..)



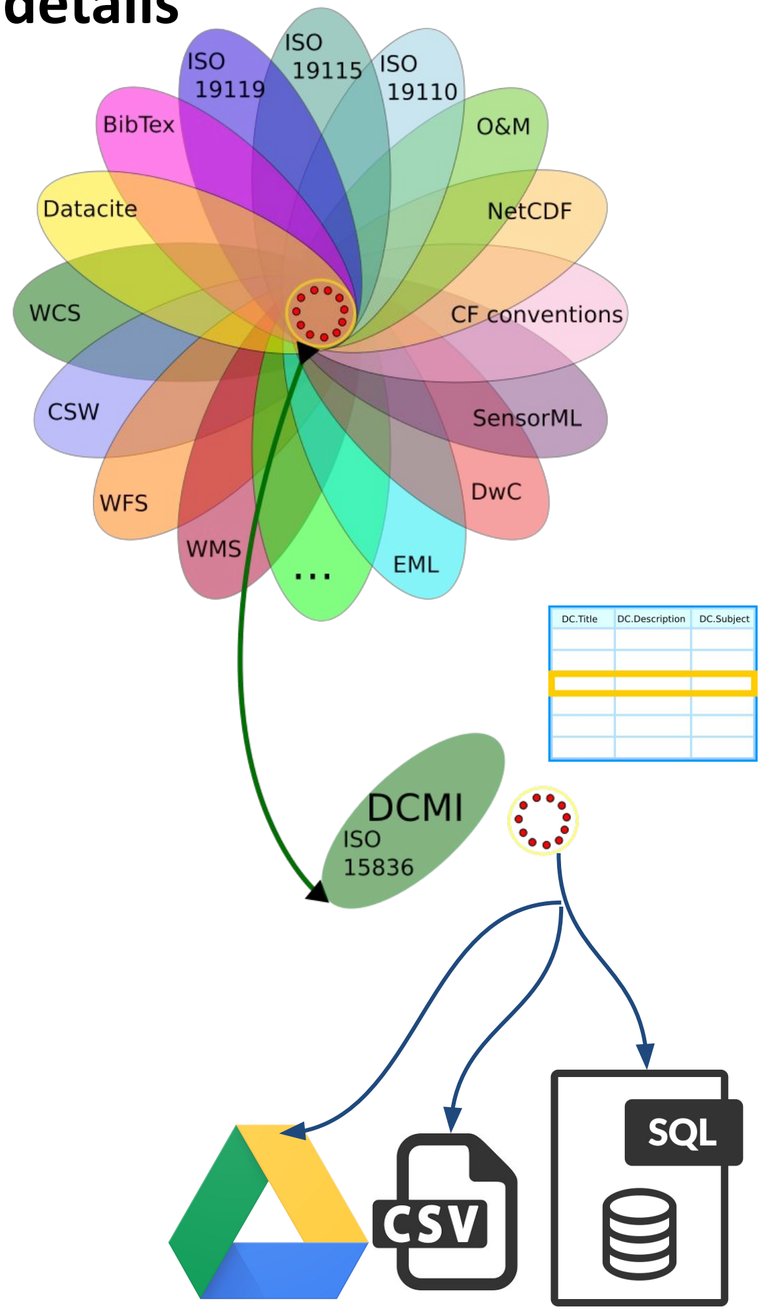
A pivot model to comply with widely used metadata standards: example



A pivot model to comply with widely used metadata standards: details

Dublin Core metadata elements are used to labels the columns of a table (CSV..)

- **Identifier**: “string_identifier” + DOIs, URI, URNs..
- **Title**: “free text”
- **Description:** “prefix:free text” (abstract, purpose, additional info..)
- **Creator**: “role:person” / “role:email” + UID, ORCID / FOAF
- **Subject**: thesaurus:keywords KVP or controlled vocabularies (eg GEMET, GCMD)
- **SpatialCoverage:** eWKT (dynamic if “Data” column valued)
- **TemporalCoverage**: ISO (dynamic if “Data” column valued with “time” dimension)
- **Date**: controlled syntax
- **Type**: free text
- **Format**: free text
- **Language**: code (e.g. fre, eng..) → norme ISO 639
- **Relation**: type:relation KVP or URLs, URIs...
- **Provenance**: “prefix:free text”
- **Rights**: “prefix:free text” or Creative Commons….
- **Data**: “prefix:free text” Rules to attach data

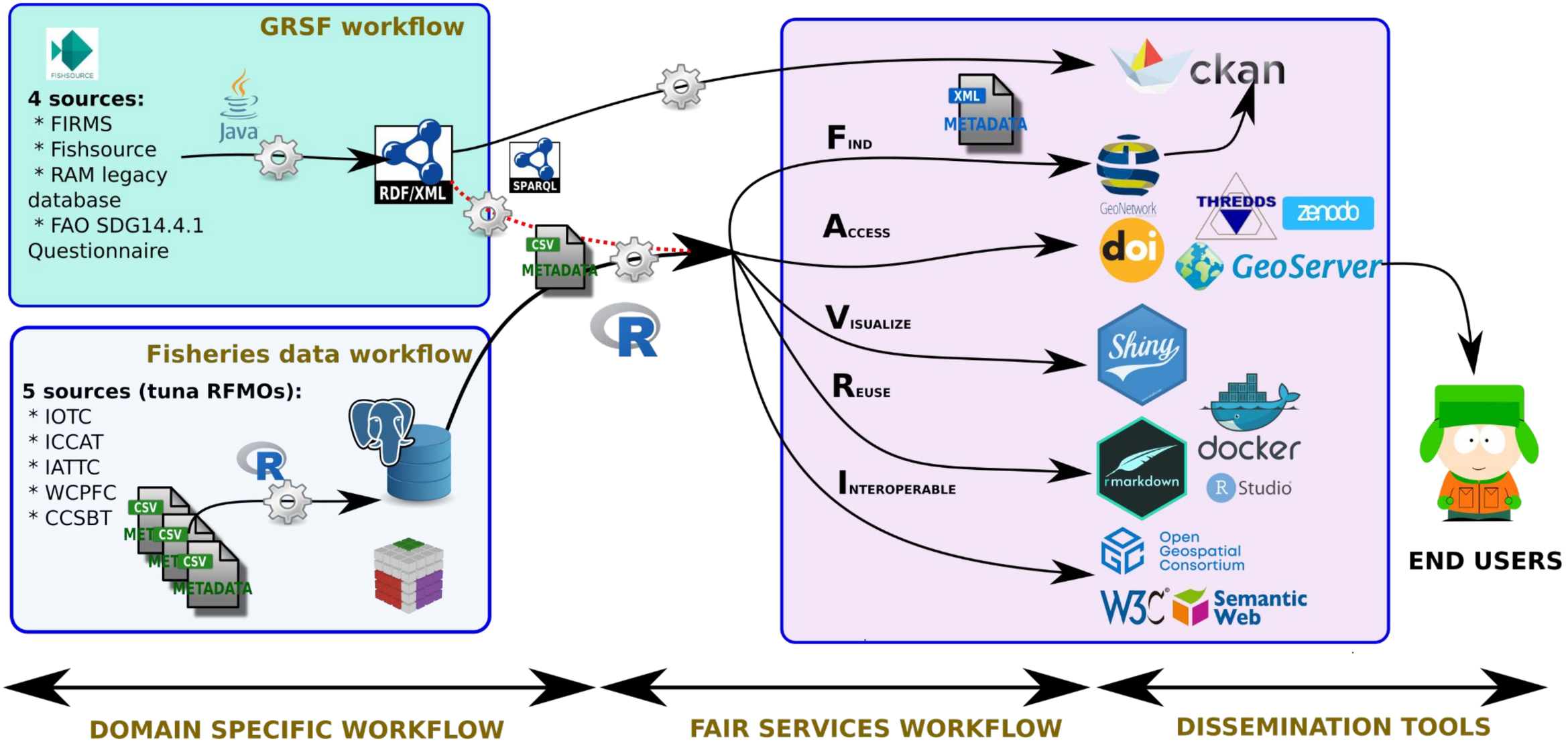


Implementing workflows with R programming language within a VRE

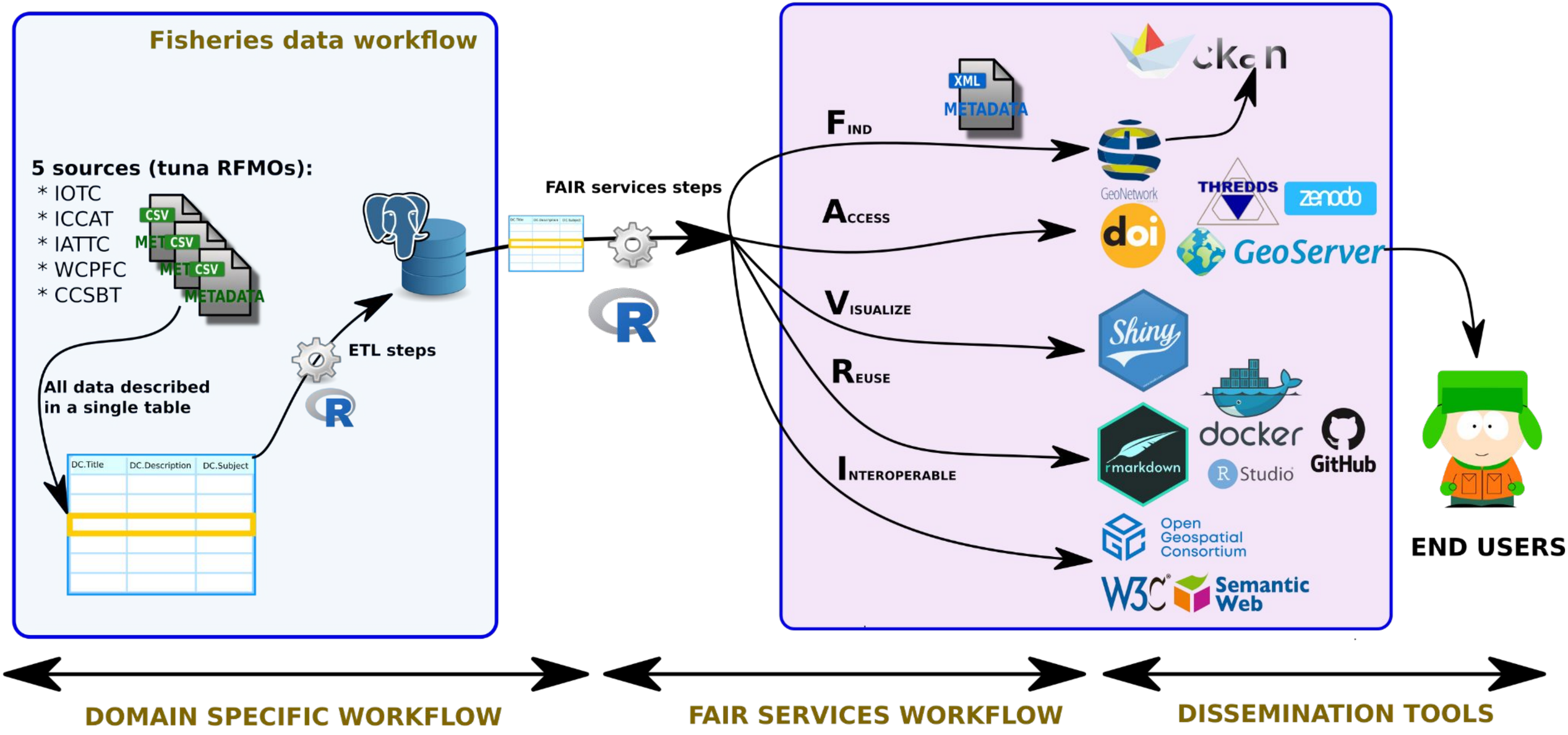
Our VRE / VLab environment requires multiple **software** and **R packages**:

- **Workflow orchestration**: geoflow package works on top of other packages
- **Multiple SDI software** to manage usual steps of the workflow:
  - extract (meta-)data from **nextcloud** (ocs4R), **Postgres** / **Postgis** (DBI)
  - read and write metadata: geometa (19115), ows4R (CSW)
  - publish metadata in **GeoNetwork**: geonapi, ows4R (CSW-T)
  - publish data in **GeoServer**: geosapi
  - assign DOIs (out of the VLab): **Zenodo** (zen4R)
- **Environment: JupyterHub** / **RStudio** IDE customized for all VLab users
  - pre-installed packages & environments (using renv package to foster reproducibility)
  - cloned GitHub repositories (TBD)

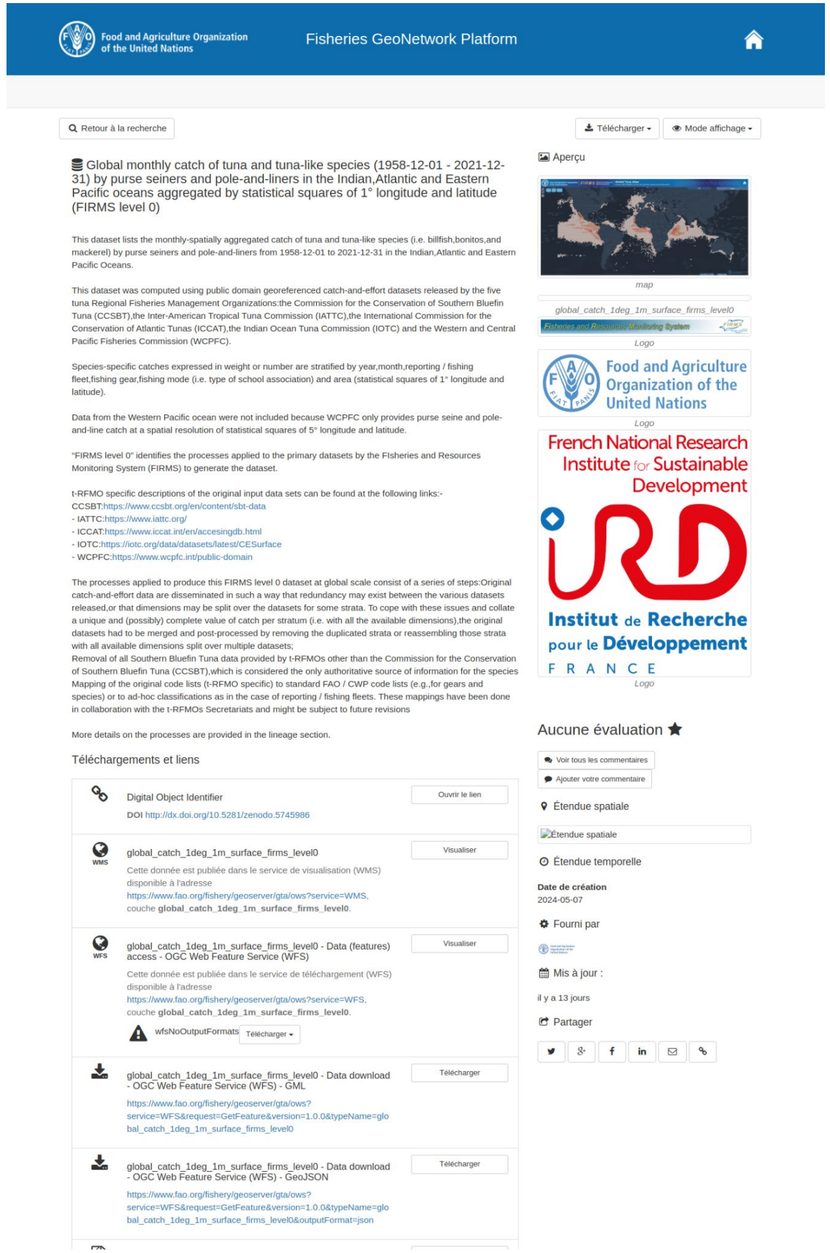
Examples of marine metadata and data management with R workflows: VLab5

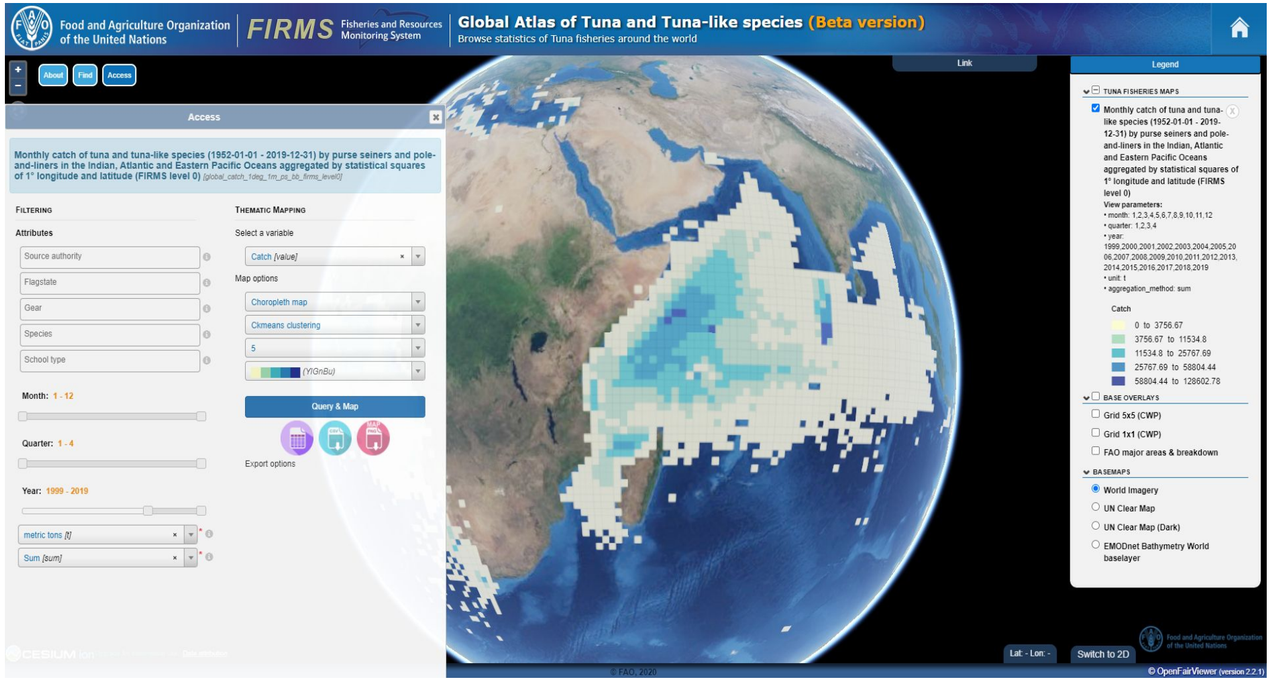


Examples of marine metadata and data management with R workflows: VLab5

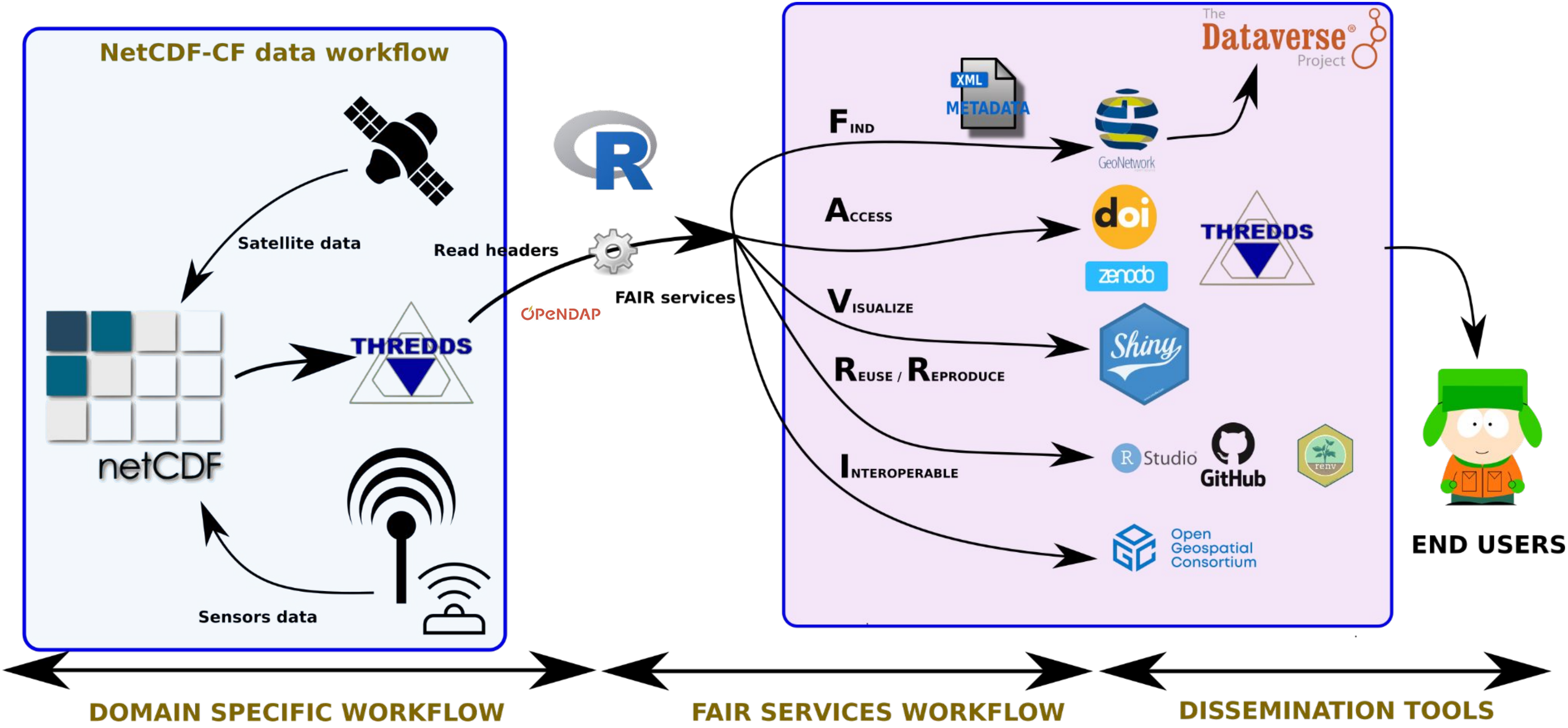


Examples of marine metadata and data management with R workflows : fisheries data

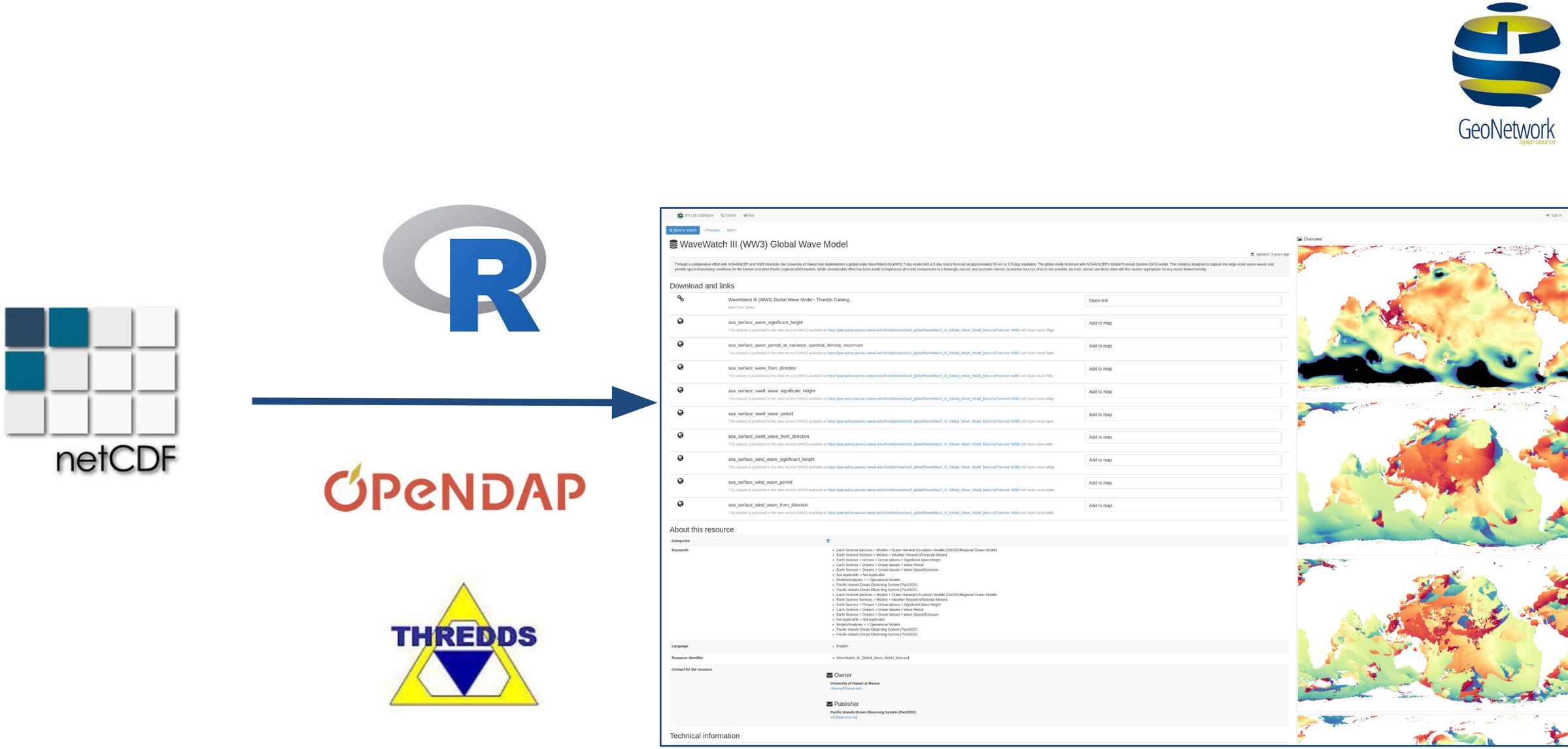




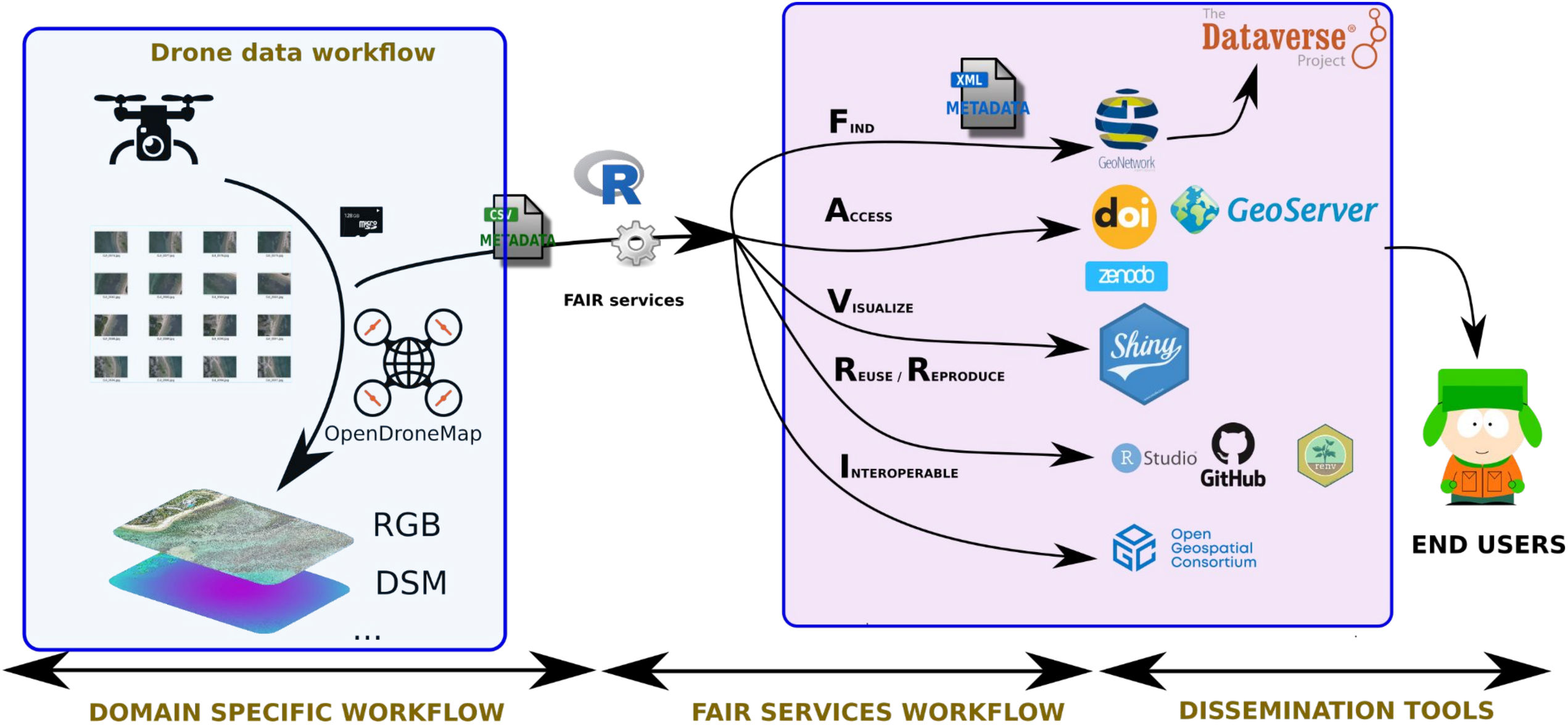
Examples of marine metadata and data management with R workflows: NetCDF data



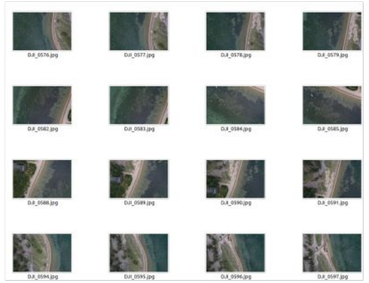
Examples of marine metadata and data management with R workflows: NetCDF data

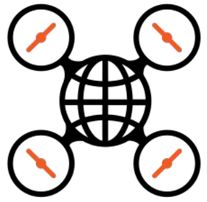


Examples of marine metadata and data management with R workflows: drone data

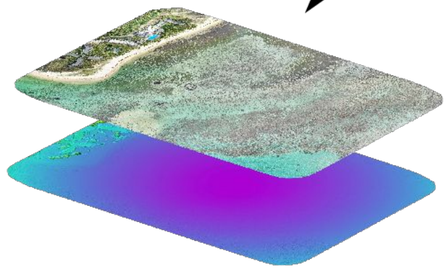


Examples of marine metadata and data management with R workflows: drone data





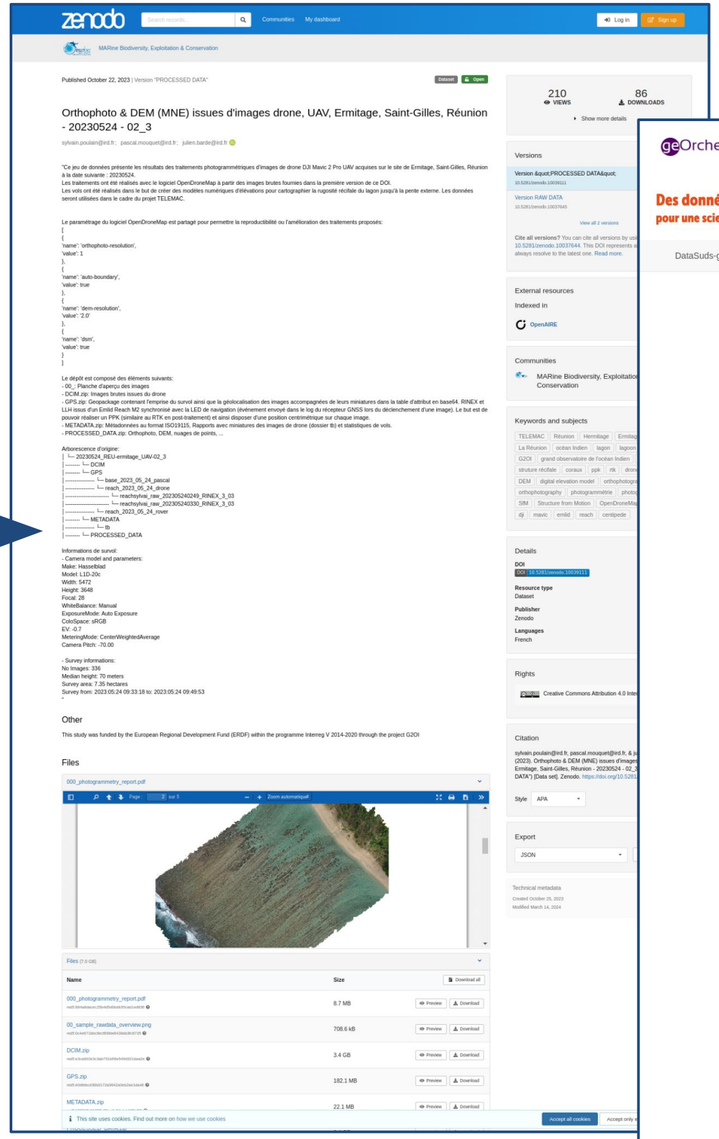
OpenDroneMap



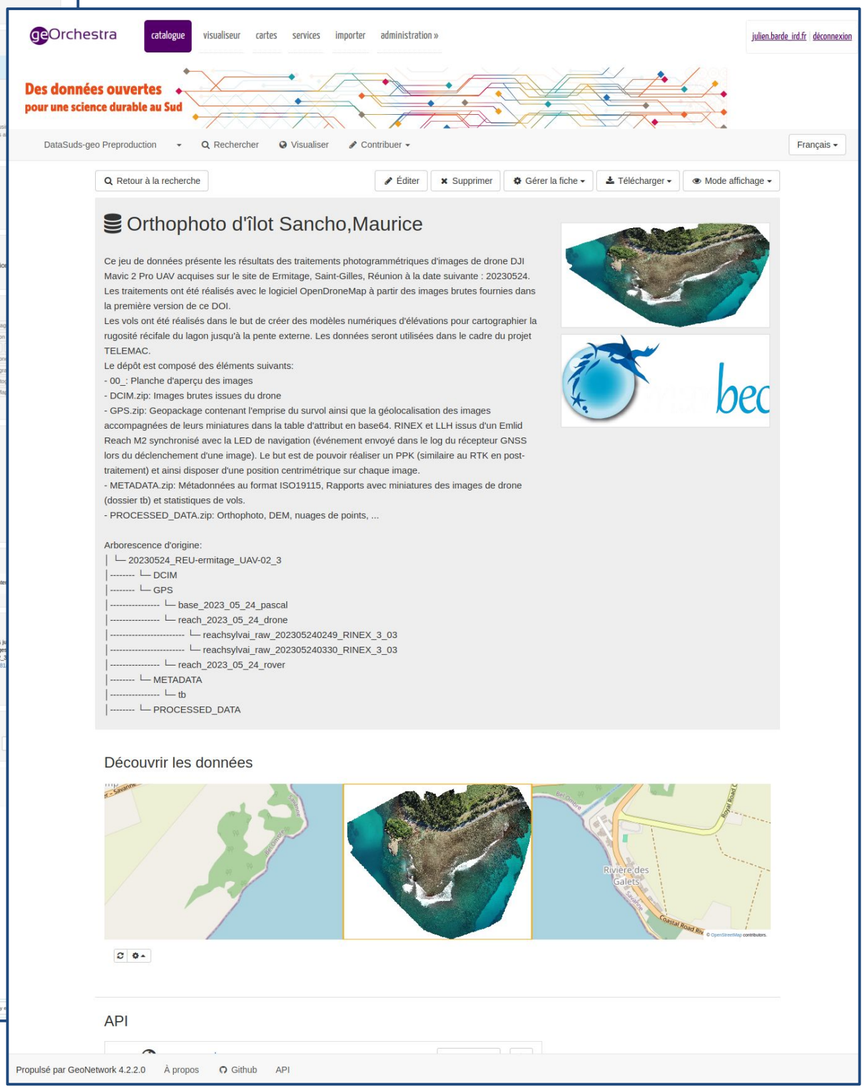
RGB

DSM

...



zenodo



GeoNetwork

eosc | Blue-Cloud2026

blue-cloud.org

@bluecloudeu

blue-cloud org

Funded by
the European Union

27-29 May 2024

International conference on Marine Data and Information Systems

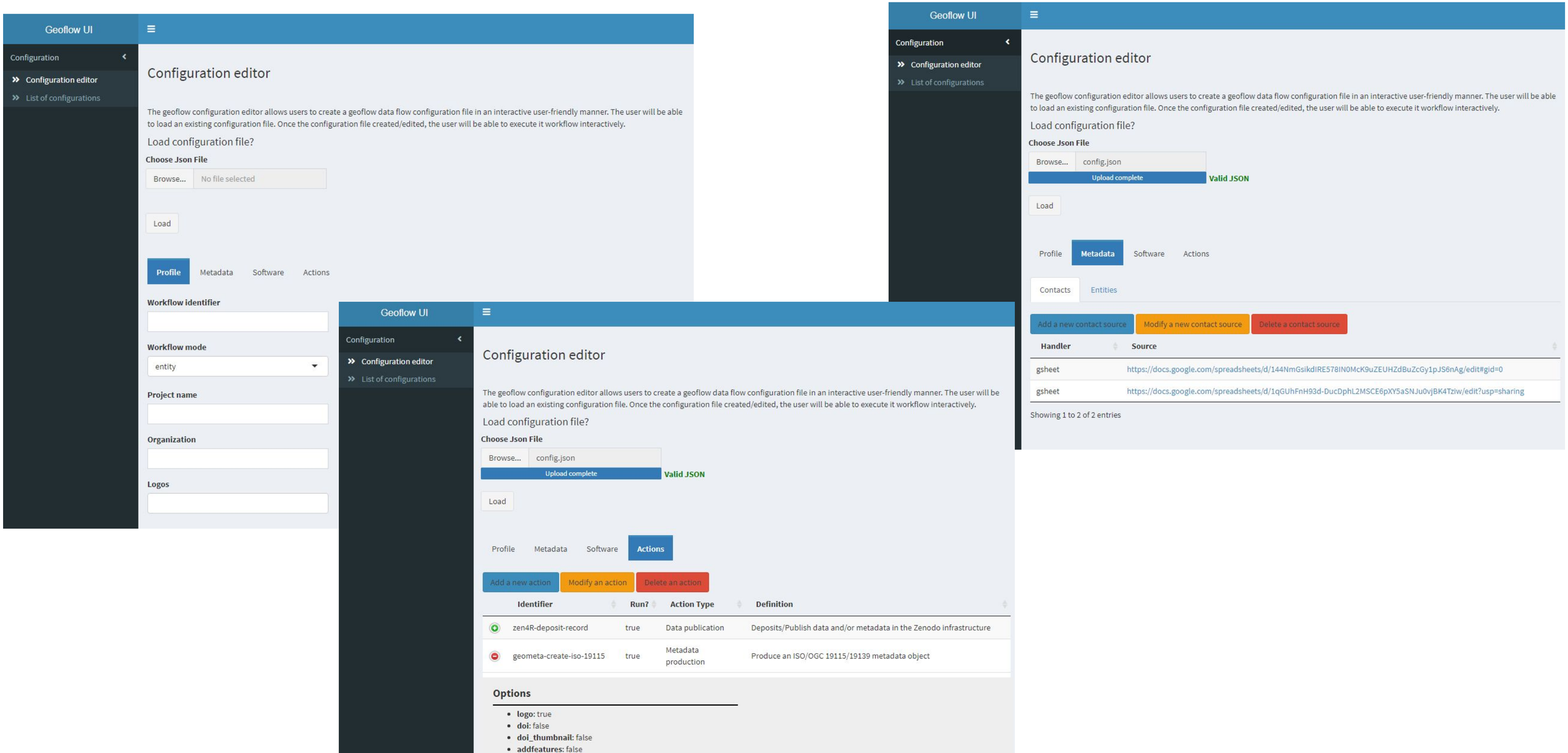
SeaDataNet AISBL · Havforskningsinstituttet · Ifremer · MARIS · MEDIN · National Oceanography Centre · INGV · iMagine · ICDE · eosc Blue-Cloud2026

Execution of a workflow with geoflow command: executeWorkflow("config.json")

geoflow requires a single configuration file “config.json”

```
{
 "id": "my-workflow",
 "mode": "entity",
 "metadata": { <metadata sources defined here> },
 "software": [ <pieces of software defined here> ],
 "actions": [ <actions defined here>  ],
 "profile": { <global profile (metadata) defined here> },
 "options": { <global options defined here> },
}
```

A geoflow shiny app to avoid manual edition of the “config.json” configuration file



Dynamic datasets: e.g. SQL queries (stored as views or not)