CCP: A Cloud Computing Platform for FAIR, Open Earth Science

Alfredo Oliviero [1], Marco Lettere [2], Andrea Dell'Amico [1], Pasquale Pagano [1]

1. Istituto di Scienza e Tecnologie dell'Informazione "A. Faedo" - Consiglio Nazionale delle Ricerche, Pisa, Italia
2. Nubisware S.r.l.

Wien, 01/05/2025

The Cloud Computing Platform

A platform for running scientific methods in a scalable, cloud-based environment (VREs)

- Provides on-demand, scalable computing through a web interface or REST APIs
- Supports running methods imported by users, available in the platform, or shared by colleagues
- Enables collaboration by sharing data, tools, and results in the same environment
- Built on containerization and JSON-based APIs for flexibility and seamless integration

What is D4Science

D4SCIENCE

Sharing is encouraged

A digital **infrastructure** for **collaborative scientific research**, combining over 500 software components into a coherent and centrally managed system of **hardware**, **software**, and **data resources**

D4Science is owned and managed by CNR

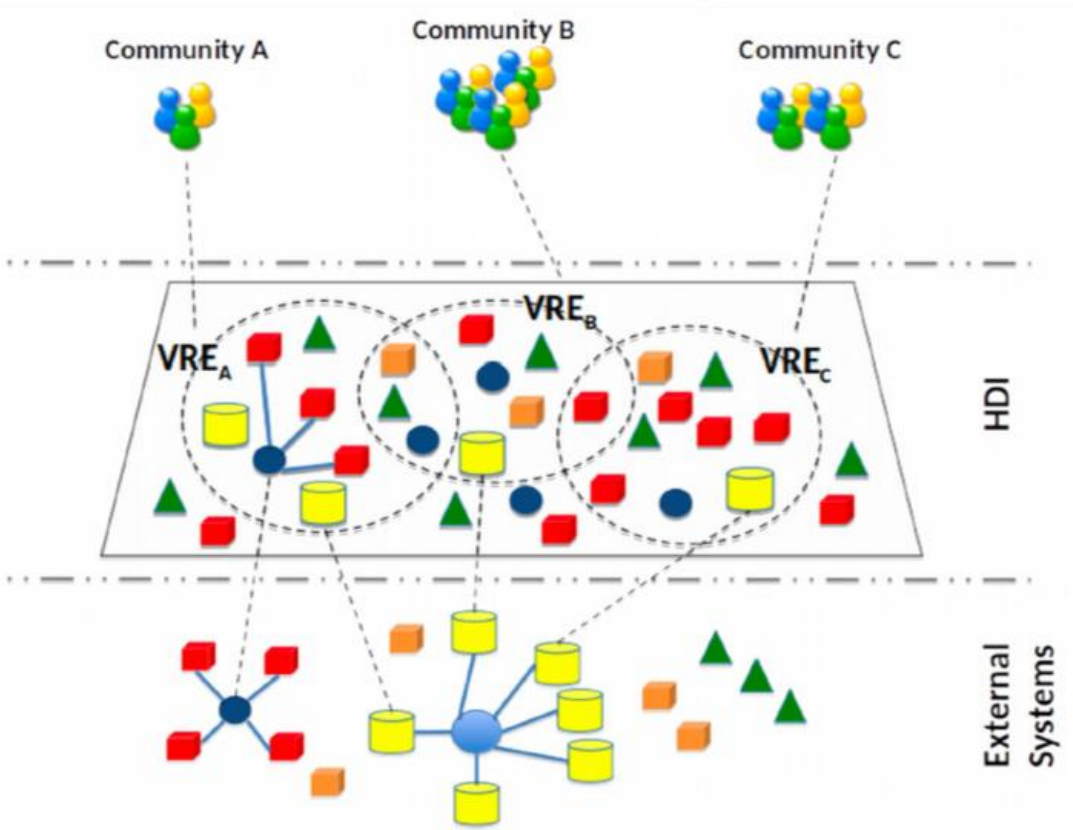
Pisa CPUs

1.624 • Dec. '22 → 6.976 • June '23 → 11.100 • Dec. '24

- Federated digital infrastructure:
- 4 sites (1 Pisa, 3 @ GARR)
- 11.100 CPUs core
- 29 TB RAM
- 1.400 TB Storage

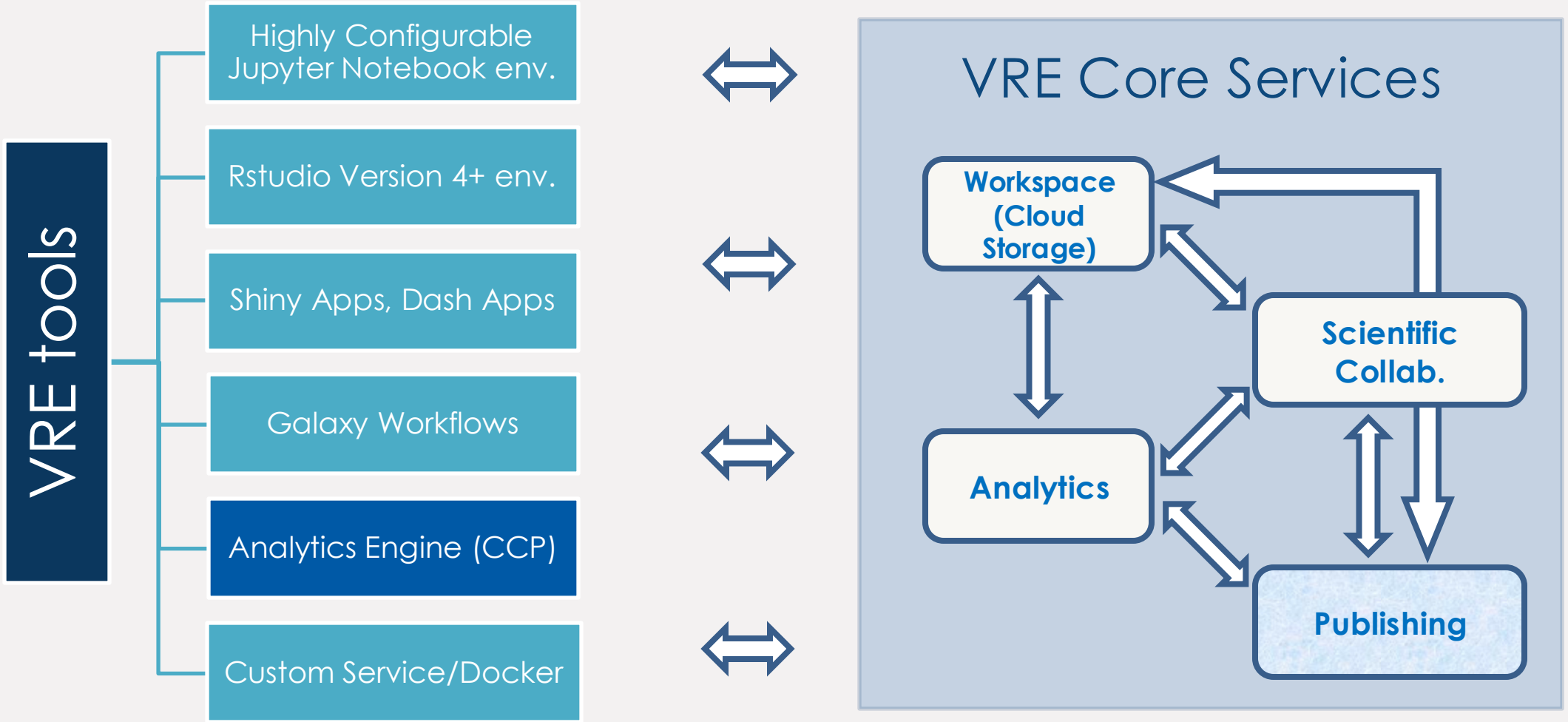
It supports ESFRI RIs, national and European projects, national and international initiatives

Virtual Research Environments (VREs)



- Secure, web-based platforms tailored to scientific workflows
- Integrate both local resources and external services (e.g. cloud storage, distributed computing)
- All accessible via a unified, on-demand workspace in the browser
- Enable researchers to:
  - Organize and store data
  - Run and monitor computational methods
  - Share tools and results
  - Collaborate across teams and disciplines

VRE - Collaborative Data Analysis



What is CCP

Method import support via a dedicated tool supporting any prog. language

Per-method Web UI and REST API access via Standard (OGC API – Processes)

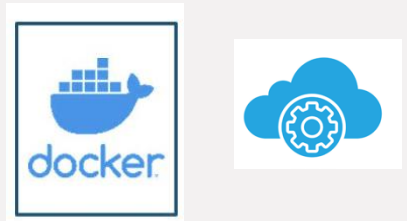
"*out-of-the-box*" methods **as-a-Service** and automatic code generation for Python, R, Julia, Jupyter

Provenance management

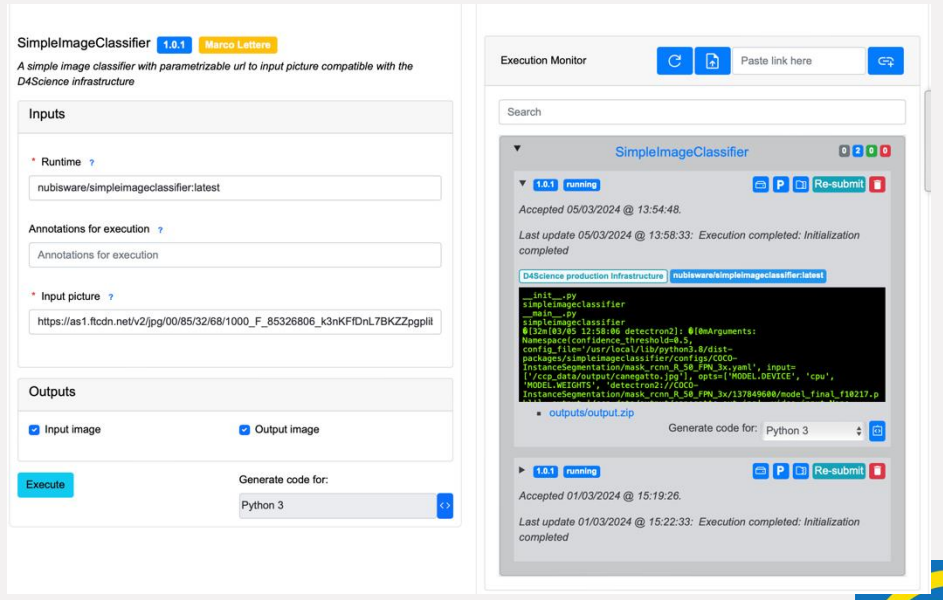
Defining Methods

Method Implementation: Script, Algorithm, library

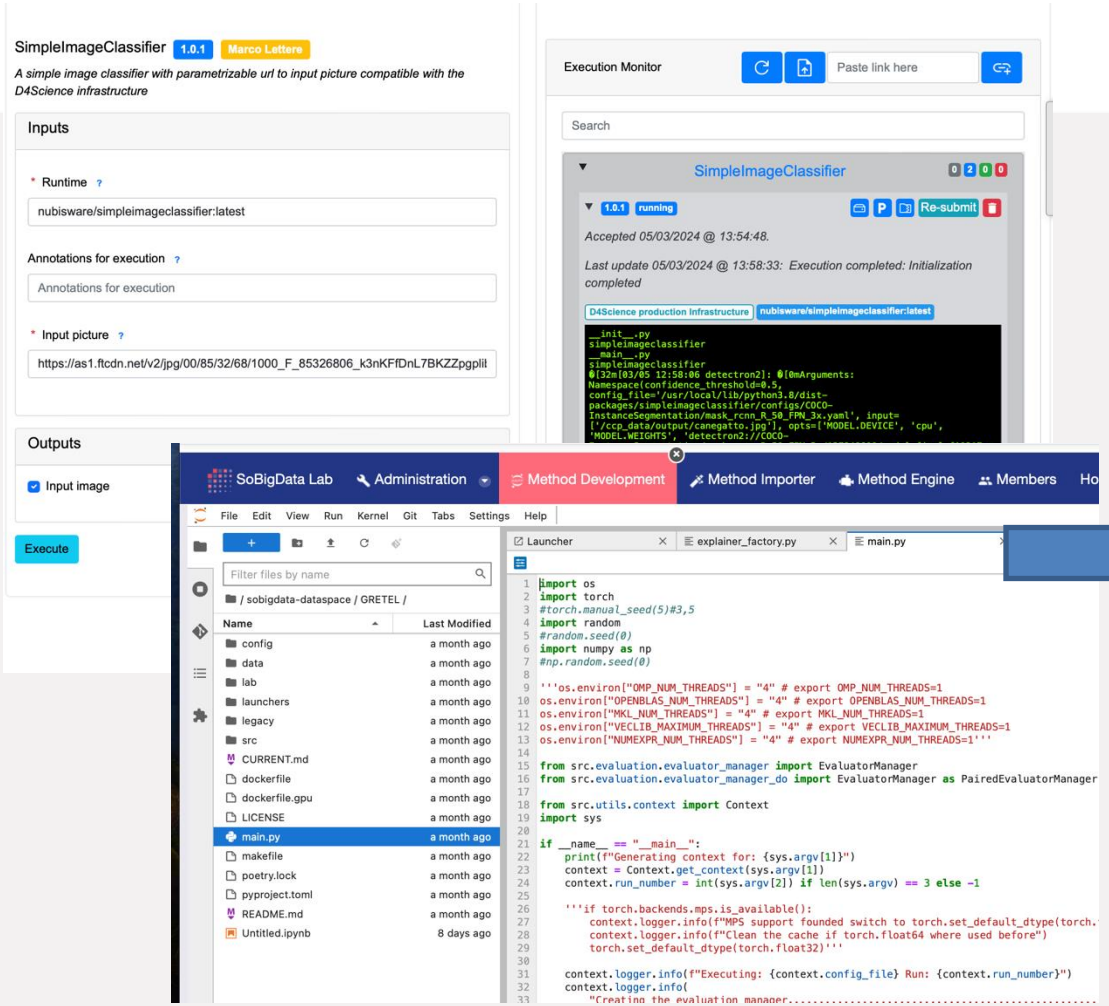
Importing phase

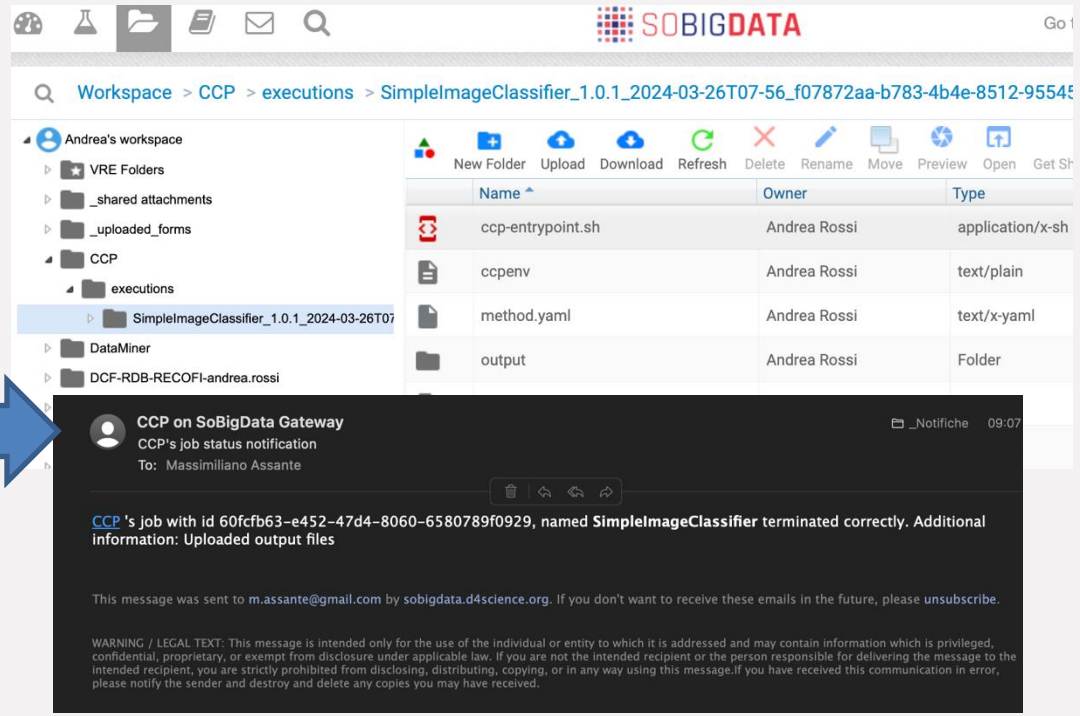


Resulting Web UI & REST API



Method Executions



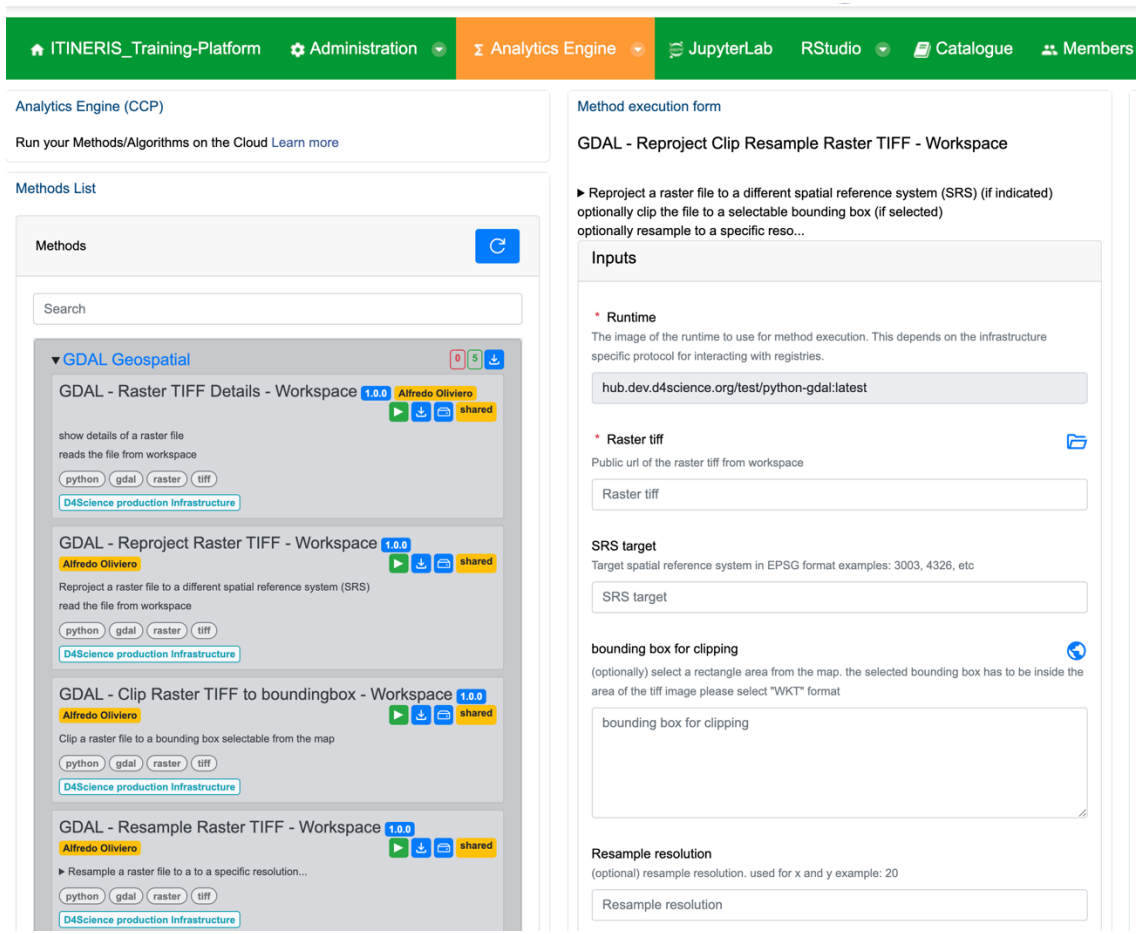


Web UI / Rest API execution

StorageHub/ Email results

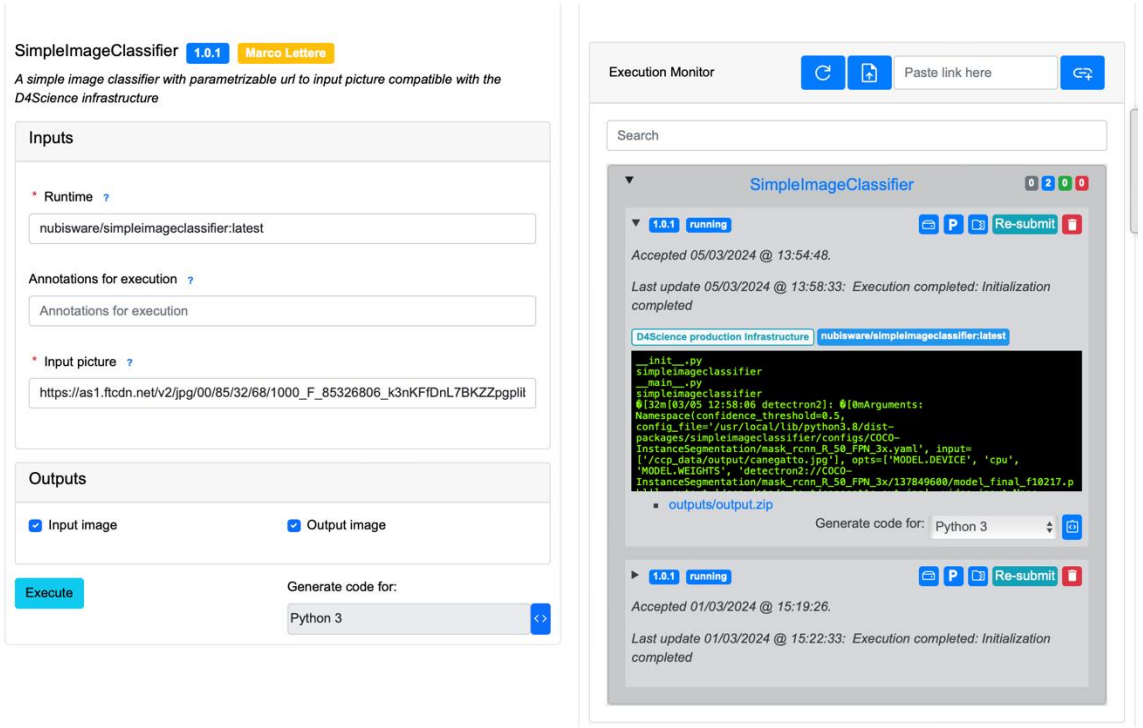
Sharing is encouraged

Use Case: Geospatial Processing



- A Python-based Docker image with tools like GDAL and the Copernicus Climate Data Store (CDS) CDSAPI is provided
- Users can extend it or build their own containers.
- Example methods:
  - Raster handling: extract metadata, convert formats, reproject raster
  - Terrain analysis: compute slope and aspect
  - Vector operations: buffering, spatial joins, overlays
  - Advanced workflows: spatial interpolation, network analysis

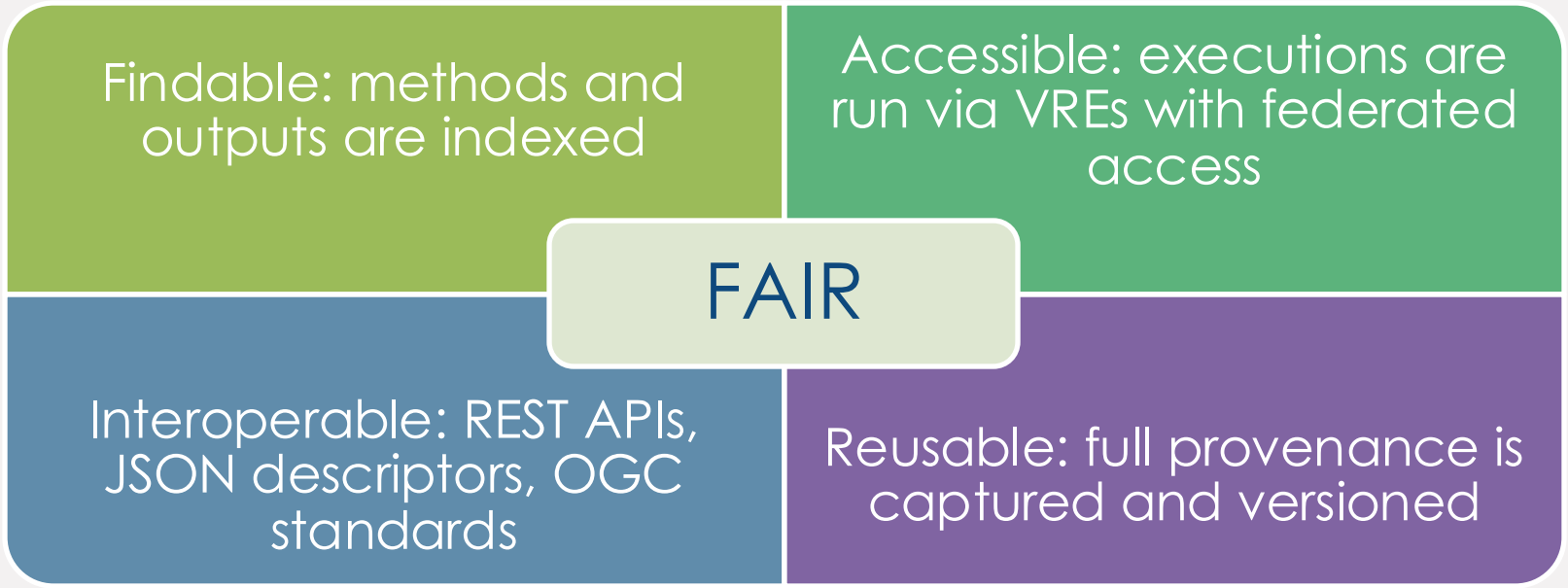
Use Case: Modeling and Simulation



- CCP supports CPU-intensive simulations within asynchronous, containerized workflows.
- Compatible with models implemented in:
  - **Python**, **R**, **Java**, **Fortran**, **C/C++**, and other scripting languages
  - Any **containerizable framework or simulation engine**
- Example workflows:
  - Groundwater flow and contaminant transport
  - Slope stability and landslide modeling
  - Seismic hazard and crustal deformation
  - Basin evolution and stratigraphic modeling
- Execution is reproducible and scalable across different infrastructures.

FAIR Principles

CCP is designed for Open Science and it fully aligns with the FAIR principles:



Provenance is tracked using standards like **PROV-O**, ensuring full documentation and reproducibility.

Why It Matters for scientists

Reproducibility across methods and data

Access to significant computational resources

Simplified execution without technical overhead

Integration into a robust infrastructure supported by CNR, widely adopted in European research

Driver for Open Science, making outputs transparent, traceable, and shareable

Conclusion

CCP brings scalable computation into your research, without the burden of system configuration

Define once, run anywhere: from quick tests to complex research

Designed for **Open Science**, built for **reproducibility**, and ready for **collaboration**

Already supporting major initiatives like **Blue-Cloud** and **ITINERIS**

Thank you

Any questions?

https://www.d4science.org/

D4SCIENCE