HERMES: RESEARCH SOFTWARE ON WINGS

Automating software publication with rich metadata

19.09.2022 I OLIVER BERTUCH[1], STEPHAN DRUSKAT[2], OLIVER KNODEL[3], GUIDO JUCKELAND[3], MICHAEL MEINEL[2], TOBIAS SCHLAUCH[2], JEFFREY KELLING[3]

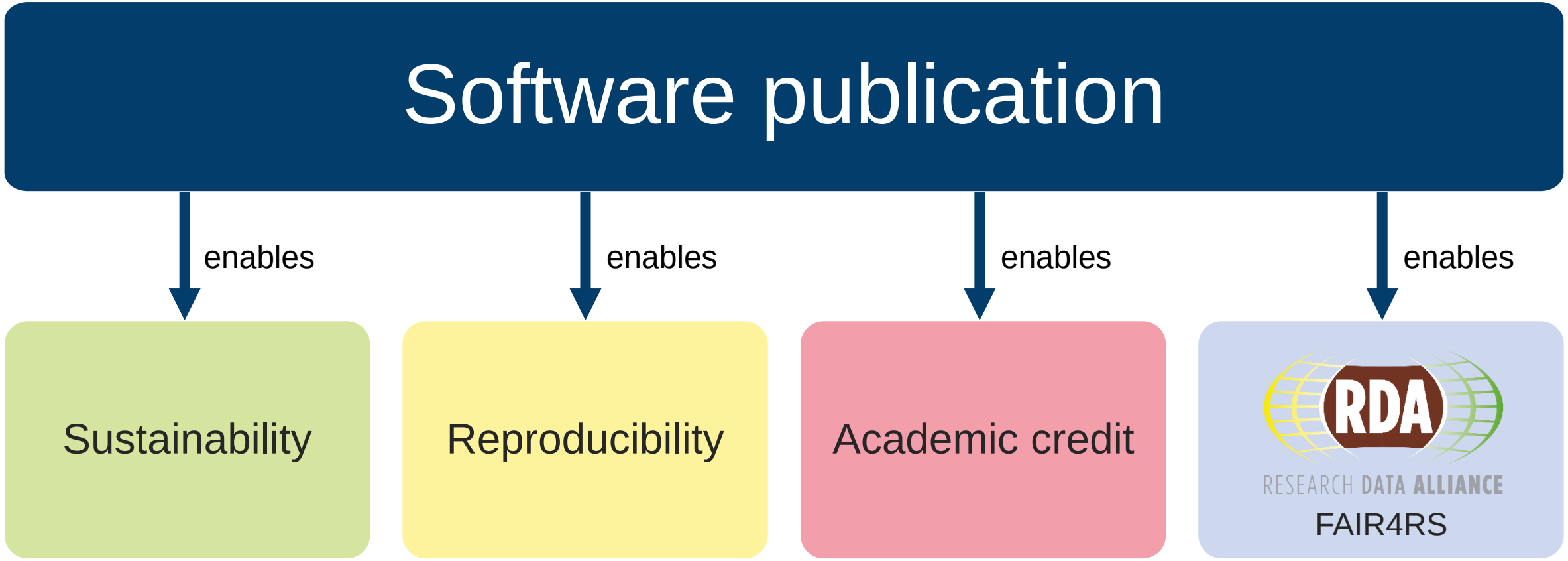
[1] FORSCHUNGSZENTRUM JÜLICH GMBH, GERMANY. [2] GERMAN AEROSPACE CENTER (DLR), GERMANY.
[3] HELMHOLTZ-ZENTRUM DRESDEN-ROSSENDORF (HZDR), GERMANY

OVERVIEW

- Software publication
- Project details

SOFTWARE PUBLICATION

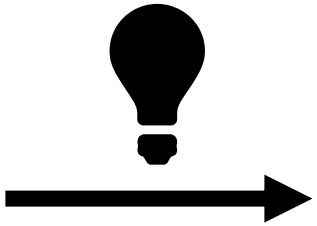
SOFTWARE PUBLICATION



SOFTWARE PUBLICATION: STATE OF THE ART

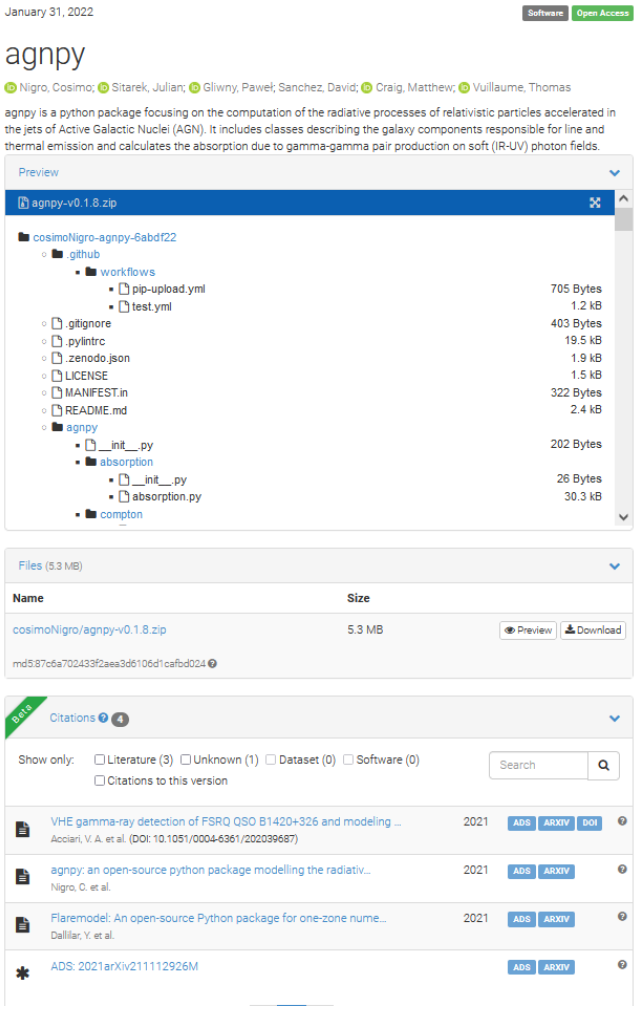
```python
model = getattr(spectra, spectrum_dict["type"])

if norm.unit in (u.Unit("erg"), u.Unit("erg cm-3")) and norm_type != "integral":
    raise NameError(
        "Normalisation different than 'integral' available only for 'spectrum_norm' in cm-3"
    )

# check the units of the normalisation
# cm-3 is the only one allowing more than one normalisation type
if norm.unit == u.Unit("cm-3"):
    if norm_type == "differential":
        final_model = model(norm, **spectrum_dict["parameters"])
    elif norm_type == "gamma=1":
        final_model = model.from_norm_at_gamma_1(
```

```json
{
    "@context": "https://doi.org/10.5063/schema/codemeta-2.0",
    "@type": "SoftwareSourceCode",
    "license": "https://spdx.org/licenses/BSD-3-Clause",
    "codeRepository": "https://github.com/cosimoNigro/agnpy",
    "contIntegration": "https://github.com/cosimoNigro/agnpy/actions",
    "dateCreated": "2019-12-17",
    "datePublished": "2022-01-31",
    "dateModified": "2021-08-02",
    "downloadUrl": "https://github.com/cosimoNigro/agnpy/releases/tag/v0.1.6",
    "issueTracker": "https://github.com/cosimoNigro/agnpy/issues",
    "name": "agnpy",
    "version": "0.1.8",
    "identifier": "10.5281/zenodo.4055175",
    "description": "agnpy is a python package focusing on the computation of the
    "applicationCategory": "astrophysics",
    "funding": "ESCAPE EU H2020 824064",
    "developmentStatus": "active",
    "isPartOf": "https://www.astropy.org/affiliated/#affiliated-packages",
```

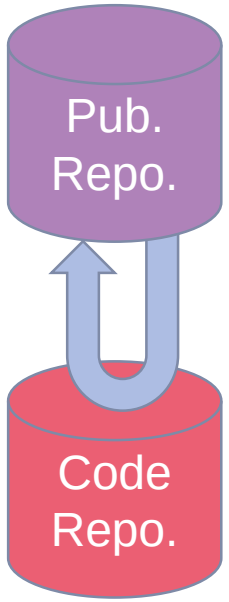


Software metadata + (software artifacts)

PID + landing page (metadata)

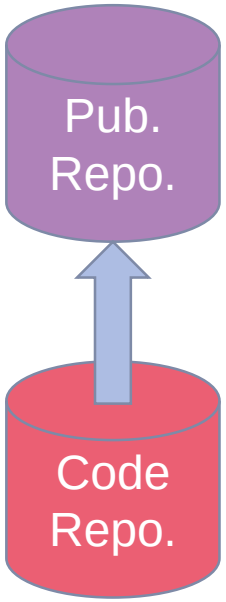


SOFTWARE PUBLICATION: STATE OF THE ART II



Pull-based workflows

- Code & metadata must be accessible
- Less control over extracted metadata
- Zenodo (and SWH) most prominent



Push-based workflows

- Works for all sw. types (CSS, ISS & OSS)
- Complete control over metadata
- Central service and/or decentral scripting

PROJECT DETAILS

HERMES: PROJECT

HERMES

HMC | HELMHOLTZ METADATA COLLABORATION

DLR JÜLICH Forschungszentrum HZDR HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF

- 07/2021 – 06/2023
- Aim: Support RSEs in automatedly publishing their software with rich metadata

Software publications with rich metadata

State of the art, automated workflows and HERMES concept

Druskat, Stephan [1,*], Bertuch, Oliver [2], Juckeland, Guido [3], Knodel, Oliver [3], and Schlauch, Tobias [1]

[1]Deutsches Zentrum für Luft- und Raumfahrt e. V.
[2]Forschungszentrum Jülich GmbH
[3]Helmholtz-Zentrum Dresden-Rossendorf e. V.
[*]Corresponding Author (team@software-metadata.pub)

January 25, 2022
Version 1

Abstract

To satisfy the principles of FAIR software, software sustainability and software citation, research software must be formally published. Publication repositories make this possible and provide published software versions with unique and persistent identifiers. However, software publication is still a tedious, mostly manual process.

To streamline software publication, HERMES, a project funded by the Helmholtz Metadata Collaboration, develops automated workflows to publish research software with rich metadata.

The tooling developed by the project utilizes continuous integration solutions to retrieve, collate, and process existing metadata in source repositories, and publish them on publication repositories, including checks against existing metadata requirements. To accompany the tooling and enable researchers to easily reuse it, the project also provides comprehensive documentation and templates for widely used CI solutions. In this paper, we outline the concept for these workflows, and describe how our solution advance the state of the art in research software publication.

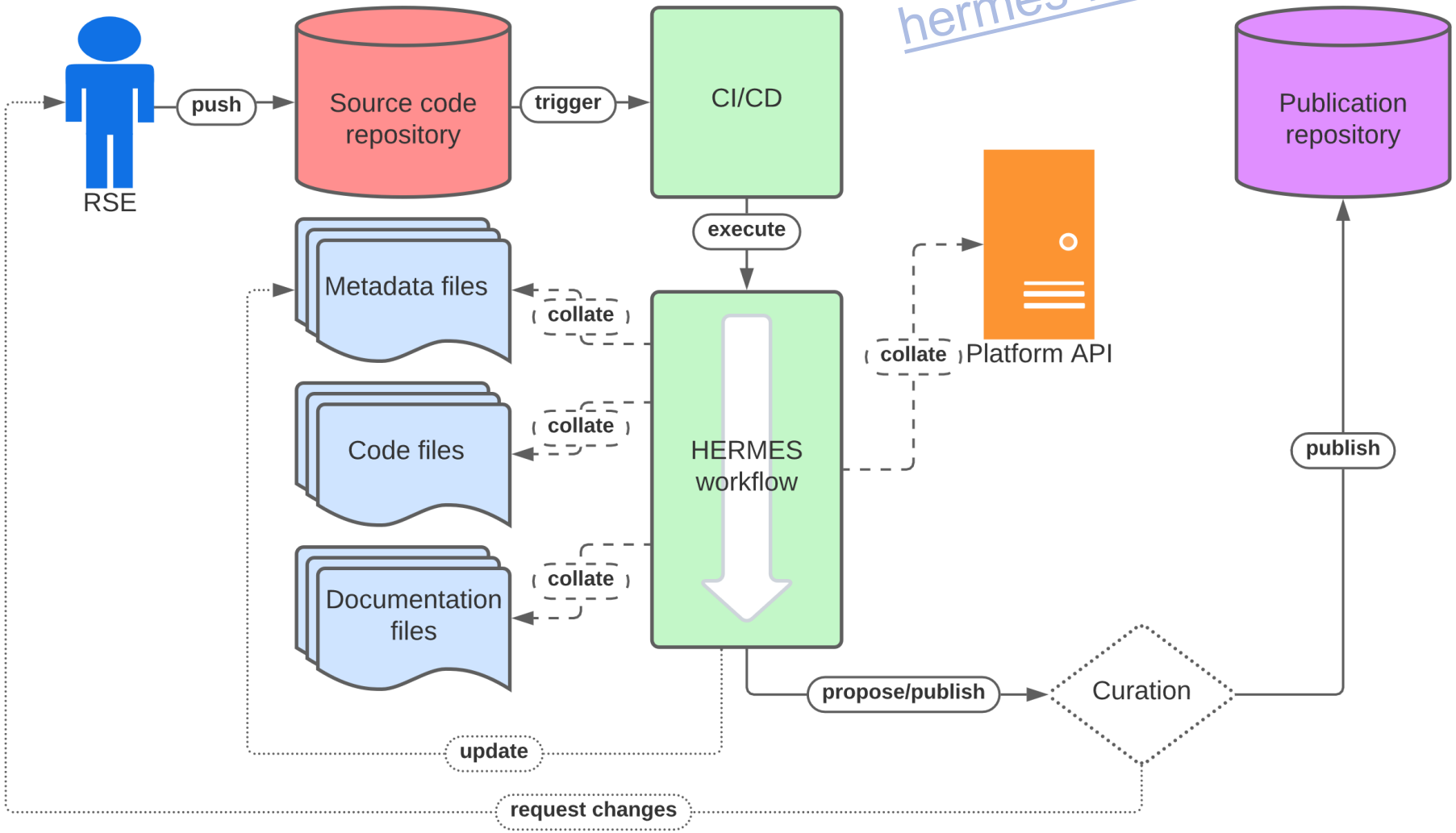
[arXiv:2201.09015 | PubPeer] | software-metadata.pub

HERMES: OUTPUTS (FOR THIS ITERATION)

- **Software**
  - Software for software publication workflow automation (workflow runner + modular pipelines)
- **CI templates**
  - GitLab CI, GitHub Actions, Jenkins, [Travis CI]
- **Improved research software-readiness in publication repositories**
  - Position paper “research software-ready repositories”
  - Respective contributions to Dataverse + InvenioRDM (data models, UI)
- **Training materials**
  - Adaption of open Helmholtz training materials (HIFIS) to include workflow usage
- **Project website**
  - One-stop shop for information and documentation
- **Policy proposals**
  - Proposals for updates to policies/guidelines at Helmholtz and cross-institutional

HERMES: CONCEPT I

Follow us on
hermes-hmc/workflow

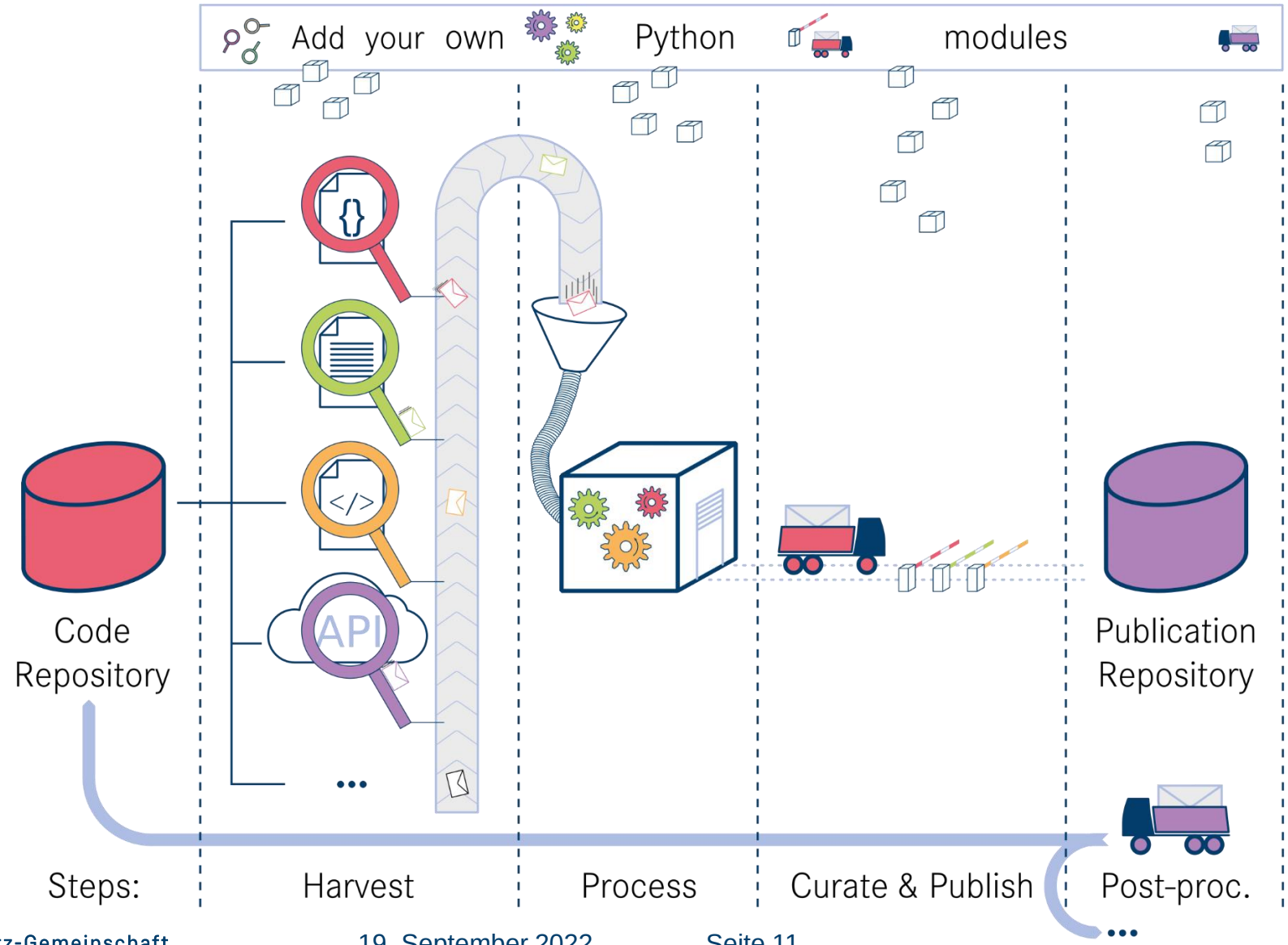


Where we are:

- Harvesting:
  Citation File Format, CodeMeta, Git metadata
- Processing:
  #TODO (unified data model)
- Curation/Deposition:
  #TODO
  (user feedback via logs)
- Post-processing:
  #TODO (CodeMeta files)

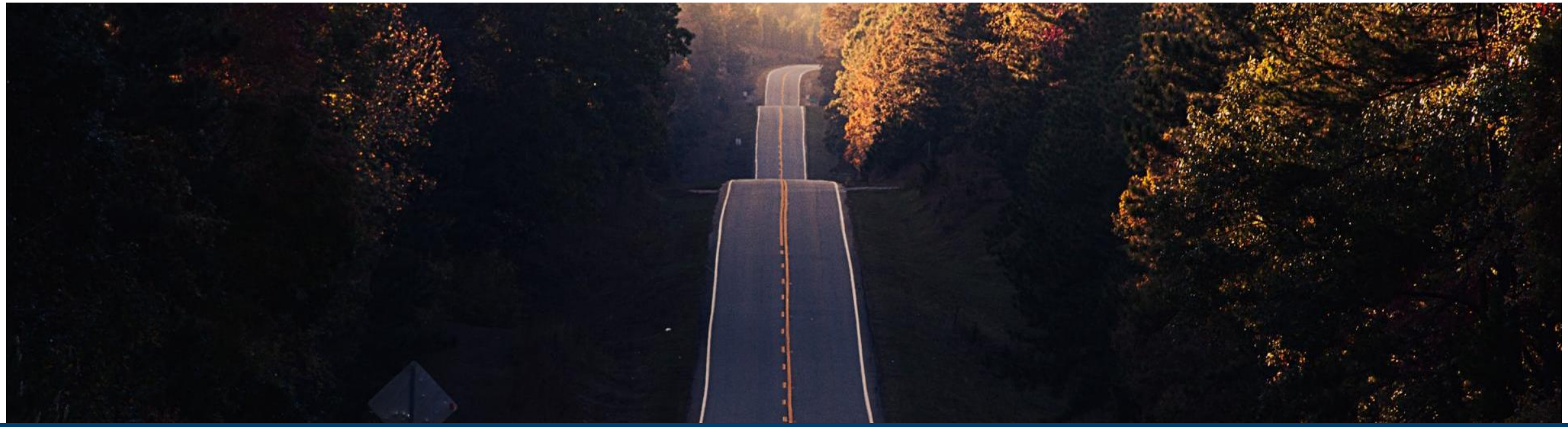
HERMES: CONCEPT II

Follow us on
hermes-hmc/workflow



HERMES: OUTLOOK

- **Project ends:**
  - Automated research software publication with rich metadata
- **Potential future work:**
  - New target repositories:
    - SURESOFT (TU Braunschweig)!
    - Helmholtz research software directory?
    - NFDI publication repositories?
  - New metadata types via extended metadata mining
  - Curation UI?
  - Support for research software KPIs



THANK YOU

TEAM@SOFTWARE-METADATA.PUB

GO.FZJ.DE/OBERTUCH

HERMES: METADATA

- **Metadata**
  - Differences in generation, scope, mode, aspects
  - Generic software metadata vs. software-specific metadata
- **Metadata formats**
  - Metadata files, snippets, third-party systems, API responses
  - Structured vs. unstructured
- **Sources**
  - Collectable structured metadata
  - (Metadata from minable structured data)
  - (Metadata from minable unstructured data)