

Towards Automating FAIR Dataset Record Deployment as a Continuous Integration Process: Use of GitHub Actions (GHA) on RDA maDMP Metadata

Miguel-Angel Sicilia^{ID}, Alberto Ballesteros-Rodríguez^{ID}, Elena García-Barriocanal^{ID}, Marçal Mora-Cantalops^{ID}, and Alberto Nogales^{ID}

University of Alcalá, Computer Science Department & Instituto Ramón y Cajal de Investigación Sanitaria (IRyCIS)

{msicilia, alberto.ballesterosr, elena.garciab, marcal.mora, alberto.nogales}@uah.es

Abstract

The FAIR (Findability, Accessibility, Interoperability and Reusability) principles emphasise making research outputs machine-actionable and consistently maintained. This paper proposes applying Continuous Integration/Continuous Deployment (CI/CD) practices to FAIR metadata publishing by leveraging GitHub Actions (GHA) as the automation backbone. Machine-actionable Data Management Plans (maDMPs) conforming to the RDA Common Standard serve as the source of truth for dataset metadata, automatically validated against institutional quality policies and deployed to external FAIR repositories such as Zenodo and Dataverse upon each versioned release. The specific contribution is an organisational automation pattern that treats metadata publication as a software release process and positions validated GitHub repositories as a harvestable upstream layer for institutional CRIS systems. A proof-of-concept prototype shows the approach has a low technical entry barrier, provides full auditability through workflow execution logs, and enforces consistent metadata quality across all published datasets.

Keywords: FAIR principles; metadata automation; GitHub Actions; maDMP; continuous integration; Zenodo; research data management

1 Background

The FAIR (Findability, Accessibility, Interoperability and Reusability) principles aim to enhance the value and impact of research data by ensuring that data can be easily discovered, accessed, integrated with other datasets, and reused for new research purposes. The FAIR principles have gained widespread adoption since their publication in 2016 [8]. However, implementation considerations are important as a supplement to these principles for effective data stewardship. Jacobsen et al. [3] proposed a collection of implementation considerations that demonstrate the need to establish clear, organisation-wide policies setting standards for elements such as clear licensing, what constitutes “rich” metadata, and how to guarantee provenance and versioning.

While the FAIR principles emphasise automation and machine actionability in accessing research outputs, there is a complementary view in which automation enables FAIR records to be produced and maintained with a consistent level of quality. Underwood [7] reframes the concept of metadata into a “dynamic” context embedded within CI/CD (Continuous Integration/Continuous Deployment) practice. Similarly, Suetake et al. [6] demonstrate how automated metadata validation and testing in workflow sharing systems can significantly improve the reusability of published research outputs. CI/CD is an established practice in the software industry that automates the building, testing, and deployment of software changes, ensuring consistent quality and rapid delivery. When applied to FAIR metadata publishing, CI/CD principles enable automated validation, quality checks, and deployment of metadata records to multiple repositories, ensuring that metadata remains accurate, up-to-date, and compliant with organisational policies throughout the research lifecycle. However, to the best of our knowledge, no established pattern exists for integrating these practices as an institution-wide, policy-enforced pipeline — the gap this work addresses.

2 Context

GitHub Actions (GHA) has emerged as a leading CI/CD tool, with practitioners increasingly migrating to it from legacy alternatives such as Travis CI [5]. We have used GHA to prototype an approach to continuous metadata publishing. For organisations that use GitHub as the official repository for automated versioning of metadata, GHA provides a natural extension to integrate the concept of dynamic, quality-controlled metadata publishing and updating.

In our research institute, metadata for datasets originates from multiple sources across different research projects, including REDCap¹ and other project-specific data collection systems. We integrate the dataset definitions specified in each project’s Data Management Plan (DMP) with the operational metadata generated during the research process. This integrated metadata is then uploaded to OpenMetadata², which serves as our internal research repository and provides a centralised catalog of dataset information. GitHub repositories complement this internal infrastructure by serving as the versioned source of truth for validated, publication-ready metadata records that are subsequently deployed to external FAIR repositories.

Machine-actionable Data Management Plans (maDMPs) are structured, digital representations of DMPs that can be read, interpreted, and acted upon by software without human intervention [4]. The RDA maDMP Common Standard provides a JSON schema for expressing dataset descriptions, access policies, and preservation intentions in a form that automated tools can validate and process. Within our institute, maDMPs serve as the authoritative source of dataset descriptions that feed both the internal OpenMetadata catalogue and the external FAIR repositories.

3 The Integration Pattern

The contribution of this work is an *integration pattern*: a policy-enforced pipeline that treats metadata publication as a software release process. The novelty lies in (a) using the organisational `.github` repository as the enforcement point for institution-wide metadata quality rules, (b) applying semantic versioning conventions to dataset records, and (c) chaining maDMP validation, FAIR metric enforcement, and multi-repository deployment into a single auditable

¹<https://project-redcap.org/>

²<https://open-metadata.org/>

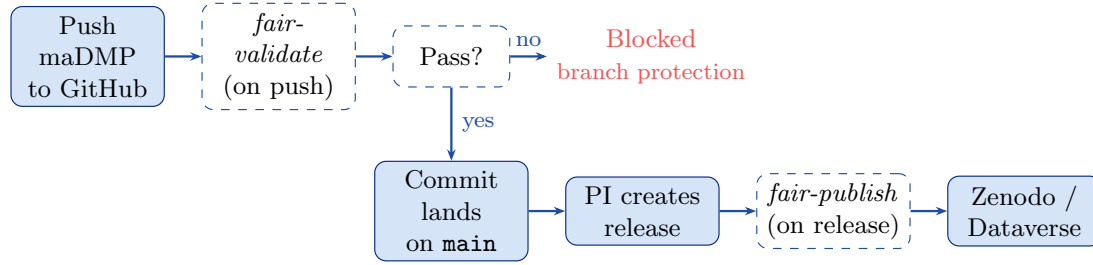


Figure 1: Two-workflow CI/CD pipeline for FAIR metadata publishing (proof-of-concept prototype). The branch protection rule makes the validation gate hard: non-validated commits cannot reach `main` and therefore cannot trigger a release.

workflow triggered by a versioned release event. Individual components — GHA, the maDMP schema, Zenodo integrations — are independently available; the contribution is their composition as an enforceable organisational pattern, which distinguishes this approach from ad-hoc upload scripts or single-repository integrations.

The pattern rests on three explicit operational assumptions. First, maDMP documents are authored using an external tool with a researcher-friendly interface; the pipeline described here is the back-end and does not address the authoring front-end, which is assumed to exist. Second, repositories are owned by the institution rather than by individual principal investigators (PIs), so the institution retains control of the maDMP and its publication history; this avoids state inconsistencies that could arise from researchers forking or deleting their own repositories. Third, the two-workflow design — validation on push, deployment on release — only constitutes a hard quality gate if GitHub branch protection rules are configured at the organisational level to require the validation workflow to pass before any commit reaches the main branch. Because releases are always created from commits on main, this configuration prevents a PI from triggering a deployment from non-validated metadata; without it, the pipeline degrades to a warning mechanism rather than an enforcement gate.

Figure 1 illustrates the overall flow of the prototype, which has been developed as a proof of concept and represents ongoing work within our institute.

The pattern also has direct implications for CRIS integration. Within a CRIS context, the GitHub repositories holding validated maDMP metadata can serve as a harvestable data layer for institutional CRIS systems (e.g., DSpace-CRIS, Pure, Converis). The structured JSON metadata conforming to the RDA maDMP schema provides a machine-readable substrate that CRIS systems can ingest via their APIs or standard protocols, reducing the manual data-entry burden common in research information management. The policy-driven quality enforcement embedded in the GHA pipeline further ensures that metadata harvested by CRIS systems meets institutional standards without requiring CRIS-level re-validation. This positions the proposed workflow as a complementary upstream layer to CRIS infrastructure, feeding validated, version-controlled dataset records into the broader research information ecosystem.

4 Prototype

4.1 Dataset Validation

The maDMP is produced adhering to the RDA maDMP schema³. This schema provides the common data model, upon which internal quality policies are layered using the `madmpy` library [1]. The quality validations include: (a) coherence of dataset artifacts with the specifications in the latest version of the actionable DMP and (b) organisation-wide policies on metadata completeness and use of terminologies. For example, within our institute, a common quality rule requires the provision of dataset coding using SNOMED CT or other relevant biomedical terminologies, ensuring semantic interoperability and alignment with established domain standards.

As Candela et al. [2] discuss, there is a variety of FAIR-related metrics and assessment tools available. These metrics can be enforced as part of the automated GHA process, such that if new or updated metadata records do not match the organisation's standards for quality, the publishing process fails. This automated enforcement ensures consistent adherence to FAIR principles across all published datasets.

The final actionable metadata is pushed to GitHub using the organisation account and tagged with topics to reflect the publication status of the outcomes (these tags can be used to mark datasets as archived or no longer available). Consequently, repositories become the source of truth from which downstream FAIR metadata records are produced.

4.2 Version Update and Deployment

The GitHub `release` event (of type `published`) triggers the rest of the process, whereby the validated metadata is transformed and pushed to one or several repositories (in the prototype, Zenodo and Dataverse are the available targets). In the case of Zenodo, an existing GHA integration is already available⁴; for Dataverse⁵, it was implemented by scripting the metadata upload using the official APIs.

The use of GHA to trigger and control metadata publishing across multiple repositories provides several key benefits: (a) consistency, as the same validated metadata is deployed to all target repositories without manual intervention or copy-paste errors; (b) auditability, since all deployments are logged and traceable through GitHub's built-in workflow execution history; (c) efficiency, as a single release action can propagate metadata updates to multiple endpoints simultaneously; and (d) flexibility, allowing the organisation to add or remove target repositories by modifying the workflow configuration without changing the underlying metadata validation processes.

A new release is created by the PI whenever a dataset reaches a stable version milestone, following semantic versioning conventions. Since the actions can be configured at the organisation level using the special `.github` repository, it is possible to enforce organisational procedures as they evolve.

³<https://github.com/RDA-DMP-Common/RDA-DMP-Common-Standard>

⁴<https://github.com/marketplace/actions/zenodraft>

⁵<https://guides.dataverse.org/en/latest/api/index.html>

5 Conclusions and Limitations

GitHub Actions establish a viable approach to maintaining FAIR metadata consistently and up to date across different repositories, enabling a policy-guided, organisation-wide account of metadata quality and richness. The approach has a relatively low technical entry barrier for organisations with varying IT capacities and implements the “continuous metadata” approach to guarantee FAIR publication of outcomes. The specific contribution is an organisational automation pattern rather than individual tool development, making it directly replicable by institutions using GitHub without requiring custom software beyond the definition of their own validation policies.

The current prototype works only with publicly accessible metadata; actual data upload is not yet implemented, as most datasets produced at our institute are not publicly disclosed due to patient privacy concerns. The prototype notifies the responsible PI by email on successful publication but does not yet send alerts when validation or deployment errors occur. Furthermore, this has only been tested with examples from past projects, and the integration with the internal system, OpenMetadata, is not yet automated. An important limitation is that the automation of the evolution of the internal FAIR policies themselves is not yet implemented, which would be needed to maintain consistent metadata quality as institutional standards evolve. Future work will focus on closing these gaps, extending the prototype to support private datasets and automated policy versioning, and evaluating the pattern across a broader set of institutional deployments.

References

- [1] Alberto Ballesteros-Rodriguez, Miguel Angel Sicilia, and Elena Garcia-Barriocanal. madmpy: A Python library for creating and validating Data Management Plans. *SoftwareX*, 31:102215, 2025.
- [2] Leonardo Candela, Donatella Mangione, and Giovanni Pavone. The FAIR assessment conundrum: Reflections on tools and metrics. *Data Science Journal*, 23(1), 2024.
- [3] Annika Jacobsen, Ricardo de Miranda Azevedo, Nick Juty, et al. FAIR principles: interpretations and implementation considerations. *Data Intelligence*, 2(1-2):10–29, 2020.
- [4] Tomasz Miksa, Stephanie Simms, Daniel Mietchen, and Sarah Jones. Ten principles for machine-actionable data management plans. *PLOS Computational Biology*, 15(3):e1006750, 2019.
- [5] Pooya Rostami Mazrae, Tom Mens, Mehdi Golzadeh, and Alexandre Decan. On the usage, co-usage and migration of CI/CD tools: A qualitative analysis. *Empirical Software Engineering*, 28(2):52, 2023.
- [6] Hirotaka Suetake, Tsukasa Fukusato, Takuya Igarashi, and Tazro Ohta. Workflow sharing with automated metadata validation and test execution to improve the reusability of published workflows. *GigaScience*, 12:giad006, 2023.
- [7] Mark Underwood. Continuous metadata in continuous integration, stream processing and enterprise DataOps. *Data Intelligence*, 5(1):275–288, 2023.
- [8] Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, et al. The FAIR guiding principles for scientific data management and stewardship. *Scientific Data*, 3(1):160018, 2016.