National College of Ireland

A Semantic Metadata Integration Framework for Enhancing FAIRness in Cross-Disciplinary Open Science Repositories

MSc Research Project
Msc in Open Data Practice

Shiva Bhanu Akkenapally
Student ID:X24106020

School of Computing
National College of Ireland

Supervisor: Dr.Eng. Bogdan-Costel Mocanu

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Shiva Bhanu Akkenapally
Student ID:	X24106020
Programme:	Msc in Open Data Practice
Year:	2025
Module:	MSc Research Project
Supervisor:	Dr.Eng. Bogdan-Costel Mocanu
Submission Due Date:	11/12/2025
Project Title:	A Semantic Metadata Integration Framework for Enhancing FAIRness in Cross-Disciplinary Open Science Repositories
Word Count:	19430
Page Count:	53

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	A. Shivabhanu
Date:	11/12/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

A Semantic Metadata Integration Framework for Enhancing FAIRness in Cross- Disciplinary Open Science Repositories

Shiva Bhanu Akkenapally
X24106020

Abstract

Open data repositories, such as Zenodo, Figshare and the Open Science Framework (OSF) provide a great deal of metadata about datasets, software and research artefacts, but all repositories have their own schema, nomenclature and extent of semantic expressiveness, thus creating interoperability divides, so that machines will not treat similar concepts as similar and do not help discover, integrate and re-use across repositories. It is a project, which constructs a semantic metadata integration platform from which data are harvested in Zenodo, Figshare, and OSF and converted to a single DataCite based schema and enhanced by matching them to existing ontologies like schema.org, DCAT, Dublin core Terms, and the DataCite ontology. The harmonised metadata is made in RDF knowledge graph form having ontology based reasoning layer and is able to deduce some additional relationships such as finding hierarchical subjects and related resources. To determine and quantitatively measure FAIRness before and after enrichment, FAIR-Checker is employed. Results indicate that the statistical significance of the differences in the interoperability and reusability scores is established proving the fact that ontology-based enrichment and RDF representation is a powerful motor to help in making research metadata machine-actionable. The suggested workflow is modular, repository independent extensional offering a non-specific model of large-scale semantic integration between open science ecosystems.

Keywords: Semantic metadata integration; Open science repositories; Zenodo; Figshare; OSF; DataCite; FAIR principles; schema.org; DCAT; RDF; Ontologies; Knowledge graph; FAIR-Checker.

1 Introduction

The swift movement to open science has changed the manner in which scholarly knowledge is created, shared and reused. In the last 10 years, research communities, funding bodies, and publishers have been urging researchers to release their datasets, software, workflows, and other supplementary material as publicly available. It has been the impetus behind the increase in large open research repositories like Zenodo[1], Figshare[2], and the Open Science Framework (OSF)[3], each downloading and uploading millions of

[1] `https://zenodo.org`
[2] `https://figshare.com`
[3] `https://osf.io`

research outputs every year and offers essential infrastructure to support the preservation of research output, issue persistent identifiers, facilitate reproducibility, and facilitate worldwide collaboration. Nevertheless, even in the middle of open research ecosystem, these repositories vary widely in metadata schemes, underlying API, semantic richness and expressiveness. This frequently leads to metadata of similar concepts, e.g. authors, types of datasets, or subject categories, being presented in varied formats, fields, and vocabularies, resulting in large interoperability gaps at the cross platform level. (Lagoze et al.; 2006). This heterogeneity restricts the machine-actionability of metadata and capabilities of humans and automated systems to uncover, synthesise, and contextualise the research outputs.

Metadata inconsistencies are not merely a technical inconvenience—they represent a major impediment to the larger vision of FAIR (Findable, Accessible, data stewardship: Interoperable, Reusable.Initiative (2018)In practice, FAIR non-compliance is not necessarily related to non-open data, but rather to non-rich, standardized, computer interpretable metadata. A collection of common terms would be required to have semantic standards. And schema, explicit semantic association, data discovery tools, indexing systems and The AI based research assistants experience some challenges in interpreting metadata in a coherent and consistent way. meaningful way.

The inability in present repository metadata is more broadly applicable to global. research infrastructure. The funding agencies and policy makers are increasingly demanding. Metadata submissions in compliance with the FAIR as a part of data management plans.Mons (2020) Global projects exist in the form of the European Open Science Cloud (EOSC), GO FAIR, and others. NIH FAIR projects focus on the semantic alignment, ontological grounding, and Linked. Information as the basis blocks of the future research ecosystems.Board (2018) Yet, in practice, many repositories still rely on repository-specific metadata structures that lack interoperability by design.

Semantic Web, such as RDF, ontologies, linked vocabularies, and so on. knowledge graphs, offers a chance to overcome such pitfalls.Berners-Lee et al. (2001) With mapping heterogeneous repository metadata to common ontologies like schema.org, Metadata It can be augmented with semantic context, made with DCAT, Dublin Core, and DataCite. understandable by machines, and embedded in unified knowledge graphs which are useful in. cross-repository query and automated reasoning. Already knowledge graphs have. radicalised areas like web search, digital libraries, biomedicine, and cultural. heritage, proving their worth in the representation of large, interconnected, semantically-rich datasets.Hogan et al. (2021)

In spite of these improvements, there is still semantic integration between open research repositories. rarely implemented. Zanado, Figshare and OSF reveal APIs that enable harvesting but. fail to give semantically normalised metadata. Scientists interested in accumulating. Platform metadata, more often than not, ends up being based on custom scripts, ad-hoc. manual alignment tables, or parsers - do not scale, cannot support. automated interoperability. Moreover, little empirical research has been done to determine the effectiveness of the. effect of semantic enrichment on FAIRness by using systematic measures like those. obtained through FAIR-Checker or F-UJI.Research (2023)

The following through the dissertation will help fill these gaps through the development, implementation and evaluation of it. semantic metadata integration model aligning the metadata in Zenodo, Figshare and OSF combined to a single RDF knowledge graph. Three frameworks are involved. constituents: (1) programmatic metadata extraction of

repositories in total, And (2) schema normalisation with a pivot model on DataCite, and Scalability Scalability of schema.org and DCAT semantic enrichment and schemas, etc. vocabularies. This can be possible since the metadata created are RDF triple. automatic reasoning, cross platform data association and machine operations.

Besides the identification of solutions to technical discrepancies, this work provides to improve FAIR. Interoperability (I1, I2, I3) and Reusability (R1, R1.2) through defining the manner in which semantic. asymmetry enhances the machine interpretability. FAIR-ness before and after FAIR-Checker provides a semantic integration as an evaluation in a quantitative manner. data on the effectiveness of the process. The outcomes of this project give an outline of how metadata integration that is scalable and based on ontology can be accomplished. FAIR based metadata refinement across the open science environment.

The other half of this chapter intervenes in the context, motivation and conceptual. that underlied my research problem in this dissertation. It outlines the meta heterogeneity of open research repositories such as Zenodo, Figshare, and OSF, and provides the justification of the creation of a semantic metadata. faith-based on the FAIR and the Linked Data principles. The chapter also gives the research questions, objectives and contributions that the work is expected to have.

1.1 Background and Context

The world has been integrated into open science and this has significantly changed the way things are. data in research and software and academic publications are developed, shared and reused. Regulated by policy demands, technology and higher research demands. transparency, open science facilitates the overall dissemination of the outcome of research in. writable and rewritable documents. It is in this regard that open research archives like Zenodo, Figshare and the open science framework (OSF) have come to the fore. publications, storage, preservation and searching infrastructures where the publication, storage, preservation and discovery of are assisted. Digital academic objects in a spectrum of knowledge.

Millions of publicly available datasets can be found in all these repositories, preprints, software packages, workflows and other resources. They also support such important fields as persistent identifiers (typically DOIs), opulent metadata fields, standardised citation formats, version tracking and version tracking. In these functionalities, open research repositories assist in raising the profile, availability, Credibility, repeatability of scientific products. The more the researchers were dependent. when the work is not in the form of traditional journals and is posted on electronic archives, the importance of the quality of metadata, interoperability, and semantic alignment grows exponentially.

The history of open research repositories has undergone various forms of development although they share the same goals. siloin nature technological environments and this renders them quite heterogeneous. technical interfaces, metadata models and schemas. The Japanese CERN created Zenodo, which was developed by OpenAIRE. divides the metadata into OAI -PMH feeds in Dublin Core, MARCXML and DataCite, and simultaneously, giving a REST API in the form of the JSON format and custom fields.CERN OpenAIRE (2024) Figshare offers metadata in the form of DataCite like metadata structure to OAI harvesting but. uses its own REST-based interactions using its own JSON schema.Figshare (2024) OSF relies entirely on a nested JSON API structured around project nodes, contributors, and relationships, with different conventions for representing

authorship, subject categories, and linked artefacts.for Open Science (2024)

The result of these structural and semantic inconsistencies is misaligned metadata. both meaning and format. Consider, to take an instance, the basic notion of an intellectual. contributor is manifested in varying forms in repositories: Zenodo uses the *creator* field, Figshare uses *author*, and OSF applies a nested *contributors* object. Likewise, thematic descriptors may appear under *subjects*, *keywords*, *tags*, or *categories*. Identifier schemes differ as well—some repositories use DOIs exclusively, while others use internal node identifiers, handles, slugs, or compound URIs. These variations make automated integration challenging and often require researchers to design bespoke scripts or manually curate mapping tables to reconcile the inconsistencies.

Importantly, the implications of such heterogeneity extend far beyond simple technical friction. Metadata inconsistencies directly hinder the implementation of the FAIR (Findable, Accessible, Interoperable, Reusable) principles, which have become a global benchmark for research data stewardship.Initiative (2018); Mons (2020) For instance:

- **Findability** is compromised when equivalent concepts cannot be indexed uniformly by search engines or discovery platforms.
- **Interoperability** (FAIR I1–I3) fails when metadata lack shared vocabularies, structured formats, or formal semantics.
- **Reusability** suffers when provenance, licensing information, or subject descriptors are incomplete or expressed inconsistently.

Several studies have shown that FAIR compliance is often inhibited not by the absence of data, but by insufficient or inconsistent metadata that cannot be interpreted semantically by machines.Wilkinson et al. (2016); Wolstencroft et al. (2016)

The Semantic Web and Linked Data principles provide a collection of principles to alleviate these challenges. technologies and methodologies facilitating machine-understandable, semantically rich. metadata representations.Berners-Lee et al. (2001) Through ontologies like schema.org, Metadata may be expressed in RDF, and in the DataCite ontology, in formal. semantics which are associated with external resources and which are made available to automated reasoning. agents. Advanced knowledge graphs based on such enriched metadata are useful. functions such as cross repository search, federated querying, concept disambiguation, and relationship inference. Hogan et al. (2021); Groth and Moreau (2020)

However, despite the demonstrated potential of Semantic Web technologies, most open research repositories do not provide semantic metadata natively. Their APIs rarely include ontology terms, machine-readable relationships, or graph-based representations. Consequently, cross-repository semantic integration remains difficult, and metadata from Zenodo, Figshare, and OSF continue to exist in isolation rather than as components of a unified research knowledge ecosystem.

This project finds a place in this problem space with the creation of a semantic metadata. integration framework to seek to harmonise metadata across Zenodo, Figshare and OSF into a coherent RDF knowledge graph. With the help of metadata harvesting, schema normalisation, ontology alignment, semantic enrichment, the framework improves the interoperability of metadata and is discovery and reuse friendly. The resulting graph of knowledge gives not only a technical approach to metadata heterogeneity and also a conceptual illustration of the. worth of semantic integration in the open science ecosystem.

Table 1: Comparison of metadata exposure formats and schemas across Zenodo, Figshare, and OSF.

Aspect	**Zenodo**	**Figshare**	**OSF**
Metadata Schema	DataCite-based	Custom schema	Custom schema
API Format	JSON REST API	JSON REST API	JSON REST API
Identifier Scheme	DOI (DataCite)	DOI (DataCite)	OSF IDs; DOIs optional
Keyword/Subject Tags	"keywords"	"tags"	"subjects"
Versioning Support	Yes	Yes	Yes
Licensing Exposure	Explicit licence field	Explicit licence field	Mixed; depends on project
Semantic Richness	Moderate (structured)	Limited (less standardised)	Variable (user-defined)
Export Formats	JSON, OAI-PMH, RDF (Beta)	JSON, CSV	JSON, CSV

1.2 Motivation

Open research repositories have become indispensable infrastructure for accelerating scientific progress. They facilitate transparent research practices, support reproducibility, and enable the long-term preservation of datasets, software, and scholarly artefacts. However, their full potential remains untapped when metadata cannot be effectively combined, queried, or interpreted by machines. Although each repository serves a similar purpose—enabling researchers to publish and share research outputs—the metadata they expose vary widely in structure, content, semantics, and granularity.

There are often difficulties in searching through repositories by researchers, develop cross-platform search utilities, or incorporate metadata to perform large scale analyses. AI-based research assistants, metadata enhancement systems, dataset recommendation. engines, and scientometric work all need coherent, machine-actionable metadata. to operate effectively. These applications cannot easily do without semantic alignment. norm heterogeneous fields and can commonly be preprocessed manually, introducing. mistakes and adds to workload.

Several motivating factors underpin the need for this research:

- **Growing metadata volume:** The scale of open research artefacts continues to expand rapidly. Zenodo alone hosts more than 3 million items, while OSF and Figshare collectively host millions more. Manual inspection or curation of metadata is not feasible at this scale, necessitating automated, machine-actionable integration approaches.Borgman (2015)

- **Lack of semantic consistency:** Metadata fields such as authorship, resource type, identifiers, or subject categories appear under different names and formats across platforms. Without ontology-grounded semantics, equivalences between concepts cannot be inferred automatically, making integrated discovery difficult (Wilkinson et al.; 2016; Lagoze et al.; 2006).

- **FAIR compliance demands:** Funding organizations like the European Commission, NIH and UKRI are demanding more and more. metadata that is going to be FAIR-compliant. FAIR Interoperability (I1–I3) directly requires the shared vocabularies, formal semantics and qualified references.Initiative (2018) Yet most repositories provide syntactic metadata only, with little semantic structure.

- **Emergence of knowledge graphs:** Knowledge graphs have emerged as a technology of representation of structured knowledge. information, which makes relationship reasoning, semantic search, and interlinking possible. between heterogeneous sources.Hogan et al. (2021) They offer an ideal mechanism for representing integrated research metadata enriched with ontological meaning.

- **Need for automation and scalability:** Automated tools are becoming more and more favored in data mining, data discovery, and meta-analysis by researchers. Cleaning or case-by-case mapping, manually, is not scalable. This is done through a semantic integration pipeline that allows repeatable, scalable, and extensible harmonisation of various metadatas, which is in line with best practice in scalable metadata processing. (**??**Wilkinson et al.; 2016).

Based on these considerations, it is timely and appropriate to have a strong ontology-based integration framework. needed to overcome the drawbacks of existing repository ecosystems. Semantic metadata integration provides a way of unlocking the full value of open repositories of research. and allowing cross-repository intelligent discovery.

1.3 Problem Statement

Although Zenodo, Figshare, and OSF serve similar purposes, the metadata they expose differ substantially in terminology, structure, encoding formats, and conceptual organisation. This creates a fragmented landscape where metadata describing similar concepts cannot be automatically reconciled. The lack of semantic alignment prevents interoperability, makes FAIR compliance difficult, and restricts the development of automated discovery tools.

Particularly, the metadata in these repositories are heterogeneous, inconsistent and. deficient in machine-actionable semantics. There is no common representation to reconcile. contributor information, types of resources, identifiers, subject descriptors or provenance. metadata. This restricts the cross-repository search, aggregation and intelligent metadata. interpretation.

There is currently no comprehensive framework that semantically integrates metadata from Zenodo, Figshare, and OSF into a unified knowledge graph while assessing how semantic enrichment affects FAIRness. Existing solutions either focus on one repository, use non-semantic mappings, or do not evaluate FAIR impact.

1.4 Research Questions

To address the challenge of metadata heterogeneity across open research repositories, this dissertation is guided by the following overarching research question (RQ):

How can semantic metadata integration be used to harmonise Zenodo, Figshare, and OSF metadata into a unified knowledge graph that measurably improves their FAIRness?

This central question is operationalised into the following sub-questions:

- **RQ1:** What schema-level, structural, and semantic inconsistencies exist across Zenodo, Figshare, and OSF metadata?
- **RQ2:** How can a DataCite-based pivot model be designed to enable cross-repository metadata normalisation?
- **RQ3:** Which ontologies (e.g., schema.org, DCAT, Dublin Core Terms, DataCite ontology) are most suitable for representing repository metadata semantics?
- **RQ4:** How can RDF and knowledge graph technologies be employed to construct an integrated graph representation of harmonised metadata?
- **RQ5:** How does semantic enrichment impact FAIRness metrics—particularly Interoperability and Reusability—when assessed using FAIR–Checker?

These research questions collectively guide the investigation from problem identification to modelling, implementation, and evaluation.

1.5 Objectives

The objectives of the project are intentionally aligned with the research questions to ensure a structured progression from analytical understanding to technical development and final evaluation.

- **O1 (Aligns with RQ1): Analytical Objective—Metadata Profiling**
  Compare and analyse the used metadata schemas, field structure, and terminologies. Figshare, OSF, and Zenodo to detect inconsistencies and semantic correspondence. challenges.
- **O2 (Aligns with RQ2): Analytical Objective—Pivot Schema Design**
  Develop unified metadata model that includes shared fields and is based on DataCite. merges semantically compatible metadata over repositories.
- **O3 (Aligns with RQ3): Technical Objective—Semantic Integration Pipeline**
  Design and implement a semantic integration pipeline that is (nonetheless) modular and performs metadata. harvesting, normalisation, parsing, ontology alignment and RDF transformation.
- **O4 (Aligns with RQ4): Technical Objective—Knowledge Graph Construction**
  Create an RDF knowledge graph, which reflects harmonised metadata using schema.org, DCAT, Dublin Core Terms and the DataCite ontology.
- **O5 (Aligns with RQ4): Technical Objective—Semantic Reasoning**
  Integrate ontology-based reasoning to infer additional relationships such as hierarchical subjects, semantic equivalences, and related resources.

- **O6 (Aligns with RQ5): Evaluative Objective—FAIRness Assessment** Conduct a before-and-after FAIRness evaluation using FAIR–Checker to quantify the impact of semantic enrichment on Interoperability and Reusability.

Together, these objectives ensure that the project builds a rigorous semantic integration workflow and evaluates its value within the FAIR principles framework.

1.6 Report Structure

The remainder of this dissertation is organised into seven chapters, each addressing a specific aspect of the research workflow from conceptual foundations to empirical evaluation. The structure of the report is as follows:

- **Chapter 2 – Literature Review:** This chapter examines existing research on FAIR principles, metadata standards, repository schemas, Semantic Web technologies, and prior work on metadata integration. It identifies the theoretical and technical gaps that motivate the development of a semantic integration framework.
- **Chapter 3 – Research Methodology:** The methodological approach in the project is described in this chapter, which involves methodology, data collection, data collection process, sampling, ethical. considerations, and methods of evaluation. It describes the way the study guarantees. rigour, reproducibility and practices aligned with FAIR.
- **Chapter 4 – System Design:** This is a chapter that discusses the conceptual and architectural design of the semantic. suo pipeline. It presents the pivot under the DataCite unified. ontology mapping strategy, schema, RDF modelling approach and knowledge graph. architecture.
- **Chapter 5 – Implementation:** The chapter outlines how each of the modules, including metadata, go about being implemented. harvesting (Zenodo, Figshare, OSF), parsing and normalisation, ontology alignment, RDF graph building, triple store implementation and inference. Key code snippets and algorithmic decision making is brought into focus.
- **Chapter 6 – Evaluation and Results:** This chapter considers the usefulness of the semantic integration approach. FAIRness is evaluated both prior and post enrichment with FAIR–Checker, and SPARQL. queries are employed to show better interoperability, data quality, and machine-actionability.
- **Chapter 7 – Discussion and Conclusion:** This chapter summarises the findings, discusses limitations, and outlines recommendations for future work, including scaling the knowledge graph, expanding ontology coverage, and supporting additional research repositories.

In brief, this structure will offer a rational sequence of ideas to the beginning. implementation, analytical evaluation, and design, and finally, contributions to Metadata integration in the open science ecosystem in a FAI-compatible way.

2 Related Work

2.1 FAIR Principles

The FAIR (Findable, Accessible, Interoperable, Reusable) principles, introduced by Wilkinson et al. Wilkinson et al. (2016),have turned into a basic template of. enhancing the quality, transparency and machine-actionability of research data and metadata. FAIR has explicitly highlighted as opposed to many other previous guidelines in data management. machine-readability and algorithmic interoperability, the recognition that modern. scientific processes are becoming more and more based on automatic tools, massive data mining, and machine-learning research assistants.As such, FAIR is not merely a set of best practices for human users but a blueprint for designing digital objects that "computational agents can find, access, interoperate with, and reuse with minimal human intervention."

The FAIR principles are organised into four categories—Findable, Accessible, Interoperable, and Reusable—each with specific sub-principles (F1–F4, A1–A2, I1–I3, R1–R1.3). These principles collectively define the necessary conditions for ensuring that research data remain usable both within their original context and in future applications, including cross-disciplinary discovery, long-term preservation, and automated knowledge extraction. In the context of this dissertation, FAIR provides a critical evaluative lens for understanding the limitations of current repository metadata and for assessing improvements enabled by semantic enrichment.

Findability

Findability requires that digital objects be uniquely identified and described with rich metadata indexed in searchable resources. FAIR sub-principles F1–F4 specify that:

- (F1) Data and metadata should be assigned globally unique and persistent identifiers (e.g., DOIs).
- (F2) Metadata should be described with rich attributes to aid discovery.
- (F3) Metadata should explicitly include the identifier of the data they describe.
- (F4) Data and metadata should be registered or indexed in searchable platforms.

Although Zenodo and Figshare are automatic in assigning DOIs, the richness of metadata differs. significantly. Zanando promotes in-depth creator details, versioning, and other supporting. identifiers, but Figshare and OSF usually do not have structured subject descriptors or precise contributor roles.

Prior studies, such as Mons Mons (2020) and Duerr et al. Duerr et al. (2018), point out that schema variance in metadata thwarts cross-repository searchability. Datasets can be technically findable, but practically invisible to even. search engines in the absence of metadata fields, vague or non standardised metadata fields.

This problem is especially pronounced in federated search systems, where inconsistent terminologies (e.g., author vs creator vs contributor) undermine indexing and ranking.

Accessibility

FAIR Accessibility (A1–A2) focuses on ensuring that data and metadata remain retrievable over time through open and standardised communication protocols:

- (A1) Data should be retrievable using a standard protocol (e.g., HTTPS, REST, OAI–PMH).
- (A1.1) The protocol should be open, free, and universally implementable.
- (A1.2) Access restrictions should be clearly stated.
- (A2) Metadata should remain accessible even if data are no longer available.

Zenodo and Figshare offer both REST APIs and OAI–PMH interfaces enabling metadata harvesting, whereas OSF provides only REST-based access.CERN OpenAIRE (2024); Figshare (2024); for Open Science (2024) This difference complicates aggregated harvesting workflows and inhibits uniform accessibility across the three repositories.

Noteworthy, FAIR does not require that information be publicly available, instead, the metadata. giving a description of the object ought to be permanent. This is particularly significant to provenance over the long term, reference integrity and repeatability.

Interoperability

Interoperability is central to the research problem addressed in this dissertation. FAIR I1–I3 specify that digital objects should:

- (I1) Use formal, shared, machine-readable vocabularies.
- (I2) Use standardised knowledge representation languages (e.g., RDF, OWL).
- (I3) Include qualified references to related objects.

In practice, most repositories fail to meet these requirements. Zenodo, Figshare, and OSF provide syntactic metadata but lack:

- ontology-grounded terms (e.g., schema.org, DCAT),
- semantic typing,
- machine-readable relationships,
- graph-based representations such as RDF.

As a result, identical concepts are represented differently across platforms, undermining semantic interoperability. For example:

$$\textit{creator}\ (\text{Zenodo}) \neq \textit{authors}\ (\text{Figshare}) \neq \textit{contributors}\ (\text{OSF})$$

This inconsistency means that automated systems cannot infer conceptual equivalences without additional semantic mediation—precisely the issue this dissertation aims to solve.

Reusability

The principle of Reusability (R1 -R1.3) states that metadata must include enough of a context. provenance and licensing data to facilitate secondary usage.

Key FAIR requirements include:

- (R1) Metadata should be richly described.
- (R1.1) Metadata should have a clear usage license.
- (R1.2) Metadata should include detailed provenance.
- (R1.3) Metadata should comply with relevant domain standards.

However, repository practices vary considerably:

- Zenodo provides structured license fields, provenance (upload dates, versions), and related identifiers.
- Figshare exposes licenses but often lacks detailed provenance.
- OSF supports contributors and timestamps but does not enforce license specification.

These inconsistencies restrict reuse and make it difficult for researchers to identify datasets that meet their requirements.

Duerr et al. Duerr et al. (2018) note that the quality of metadata directly affects repeatability and scientific process integrity. In the absence of provenance, datasets cannot be safely recombined or analysed in volume.

FAIR as a Foundation for Semantic Integration

The FAIR principles give focus on machine actionability which is very similar to the theme of. Semantic Web technologies. RDF, OWL and shared ontologies deal with exactly the FAIR. things that are not satisfied by repositories at the moment:

- RDF provides a formal, machine-readable data model (I2).
- Ontologies provide shared vocabularies (I1).
- Linked Data principles support qualified relationships (I3).
- Knowledge graphs support rich provenance representation (R1.2).

Thus, FAIR provides both:

1. **A justification** for semantic integration, and 2. **A mechanism** for evaluating improvements to metadata quality.

This makes FAIR an essential theoretical and evaluative framework for this dissertation.

2.2 Metadata Standards

Metadata standards define formal, predictable formats of the description of digital. objects, allowing the researcher, institutions, and information systems to sort out, find, distribute, and store academic works. Within the framework of open research. metadata standards are repositories which are used as the foundation of dataset discovery, semantic. correspondence, reference control, and interoperability. Nevertheless, excessive use and distribution of the technology are uneven. different application of these standards to repositories adds of relevance to the semantic disaggregation witnessed in academic metadata ecosystems.

In this section, the main metadata standards applied in this dissertation are reviewed-Dublin Core, DataCite, schema.org, DCAT/DCAT-AP, and OAI–PMH–emphasize their. purpose, architectural and design features and constraints in facilitating semantic. interoperability between Zenodo, Figshare, and OSF.

Dublin Core

Dublin Core (DC) is among the oldest and most widely adopted general-purpose metadata schemas, originally developed for describing web resources but now applied broadly across digital libraries, repositories, and archival systems.DCMI (2023) The schema consists of 15 core elements—including *creator*, *subject*, *title*, *description*, *identifier*, and *date*—as well as a larger set of qualified elements for enhanced expressiveness.

The simplicity of Dublin Core has made it easy to adopt but this simplicity has also led to its use. introduces ambiguity. Most of the elements are deliberately general (e.g., the term creator could be used. to writers, editors, compilers or compilers, leaving repositories to read. terms inconsistently. Moreover, DC does not have domain field that are necessary. research data management, including type of resources, funding, et cetera. versioning identifiers, and associated identifiers.

Zenodo and Figshare expose Dublin Core metadata through their OAI–PMH interfaces, but the actual mappings vary substantially. For example, Figshare often collapses multiple creator names into a single string, while Zenodo maps contributors to distinct structured fields. Such inconsistencies undermine cross-repository interoperability, as metadata aggregators must infer or manually normalise differences in semantic intent.Lagoze et al. (2006)

DataCite

DataCite Metadata Schema 4.4 offers a highly structured framework which is domain neutral. purposely created to assist research data, software, and academic artefacts. associated with DOIs.Group (2021) Key features include:

- structured **creator** and **contributor** roles,
- rich **resource types** (e.g., dataset, software, workflow),
- explicit **related identifiers** supporting provenance and linkage,
- controlled vocabularies for **descriptions** and **subjects**,
- mechanisms for recording **funding information**.

Zenodo is the most similar to the DataCite schema among the major open repositories, facilitating good quality DOI assignment and promoting contextual rich metadata. Figshare provides DataCite-like encasings via OAI–PMH, though a custom one with a JSON schema in its REST API, which brings inconsistencies to the description of contributors, categories, and subject descriptors. Figshare (2024)

Although DataCite offers the best backbone in cross-repository metadata. alignment, Zenodo and Figshare do not use the schema as it is defined. OSF uses neither DataCite nor any standard scientific metadata schema, and this can have a huge impact. issues with harmonisation. These variations encourage the application of DataCite as an. this dissertation, facilitating ordered normalisation among repositories having different native metadata formats.

schema.org

schema.org is a community-driven vocabulary widely used for enhancing metadata on the web.Community (2024) Its *Dataset* class includes properties such as:

- **creator**, **author**, **publisher**,
- **keywords**, **about**, **citation**,
- **spatialCoverage**, **temporalCoverage**,
- **distribution** and **license**.

Academic metadata schema.org is not a domain specific academic metadata standard. at the center of discovery of datasets on the open web, most prominently Google Dataset Search which is dependent on schema.org annotations.

Despite its global reach, Zenodo, Figshare, and OSF do not natively expose schema.org metadata. This omission reduces their visibility in search engines and limits cross-platform interoperability. Adding schema.org properties during semantic integration increases machine-actionability and enhances FAIR findability and interoperability.

DCAT and DCAT-AP

The Data Catalog Vocabulary (DCAT), developed by the W3C, provides a standard model for describing datasets in data catalogues.Commission (2020) DCAT focuses on:

- dataset distributions,
- access URLs,
- licensing,
- themes and classifications,
- relationships between datasets.

DCAT-AP extends the specification to support EU-wide open data portals and enforce semantic alignment across heterogeneous public-sector data infrastructures.

DCAT is particularly well-suited for integrated metadata catalogs such as those produced in this dissertation, as it provides a higher-level conceptual model for describing datasets independent of repository-specific schemas. Using DCAT harmonised fields across Zenodo, Figshare, and OSF enhances interoperability and supports federated discovery.

OAI–PMH

The Open Archives Initiative Protocol for Metadata Harvesting (OAI–PMH) provides a machine-readable protocol for exposing repository metadata through XML-based schemas such as Dublin Core.Open Archives Initiative (2015)

Zenodo and Figshare use OAI-PMH, which allows the workflows of harvesting to be regular. aggregators OpenAIRE, DataCite Commons and institutional repositories. OSF, in comparison, does not support OAI-PMH, as it relies only on metadata thanks to use of REST APIs. retrieval. This brings another complexity in construction of cross-repository. harvesting pipelines these processes should facilitate heterogeneous access as the integration process. mechanisms.

Although OAI–PMH facilitates metadata access, it does not enforce semantic consistency or ontology alignment. Each repository remains free to interpret metadata elements according to its own schema, leaving integrators responsible for semantic harmonisation.

2.3 Repository Metadata Comparison

The open repositories are important in helping to disseminate the research. conservation of academic works, but their metadata models vary widely in design as well as implementation. Such inconsistencies are the results of repository-specific. priorities, historical development directions, and dissimilar slopes of compliance with. defined metadata specifications. What it yields is a landscape which is heterogeneous in concept. equivalent entities- contributors, resource types, subjects or related identifiers. are represented in a manner that is incompatible thus causing a semantic misalignment and decreased. interoperability.

In this part, a comparative analysis of the metadata models, API formats are to be presented in details. identifier mechanisms, and subject classification mechanisms in Zenodo, Figshare, and the Open Science Framework (OSF). These differences are crucial in the knowledge of the same. designing a productive semantic metadata assimilation structure.

Zenodo

Zenodo run by CERN and OpenAIRE is one of the best-organized and standards-compliant. repositories. It has metadata available as both REST API (JSON) and OAI feeds. DataCite and Dublin Core and MARCXML are examples of schemas used.CERN OpenAIRE (2024)

Key characteristics of Zenodo's metadata include:

- **Strong DataCite alignment:**Zenodo adheres to DataCite schema. enabling rich creations of creators, contributors, resource types, funding information, and associated identifiers.

- **Versioning and concept DOIs:**Both version-specific DOIs are allocated by Zenodo. and concept DOIs which identify the abstract identity of a changing dataset. This enables reproducibility and inter-version citation.
- **Structured contributor fields:** Zenodo distinguishes creators, contributors, affiliation, and roles, even though it does not embrace the use of ontology-based terms.
- **Subject metadata:** Fields are free-text and restrictive, the subjects and keywords. semantic accuracy in comparison to controlled vocabularies.
- **Lack of native semantic annotation:** Zenodo is not publishing schema.org, DCAT or any other ontology-based representation, even though supporting structured. DataCite mappings.

Although Zenodo has reasonably good and detailed metadata, it lacks semantic. grounding constrained machine-actionable repositories interoperability.

Figshare

Figshare reveals metadata in two interfaces: a custom REST API based on custom JSON, and an interface based on. OAI-PMH interface in a DataCite-like schema.Figshare (2024) The disparity between these two representations creates challenges for consistent metadata harvesting.

Notable characteristics include:

- **Custom JSON schema:** The REST API uses repository-specific field names that deviate from DataCite (e.g., `authors`, `categories`, `defined_type`).
- **Uncontrolled subject classifications:** Figshare uses tags and categories that are not linked to controlled vocabularies or ontologies, reducing semantic clarity.
- **Limited provenance:** Provenance details such as version history, related identifiers, and contributor roles are less structured than in Zenodo.
- **Resource types:**Figshare-specific resource types are being used to assign resources. labels, to be mapped in order to integrate across repositories.
- **OAI–PMH exposure:** Figshare's OAI–PMH feed includes simplified metadata, making it suitable for harvesting but insufficient for high-quality semantic alignment.

Figshare's metadata are relatively rich but lack the consistency and structure required for semantic interoperability.

OSF

The Open Science Framework (OSF) exposes metadata exclusively via a nested JSON REST API.for Open Science (2024) Its metadata model differs fundamentally from Zenodo and Figshare because OSF is built around a **project-centric** architecture rather than dataset-centric design.

Key features include:

- **Hierarchical metadata:** OSF projects contain components, files, and contributors, requiring deeper traversal to extract metadata for individual datasets.
- **Contributor model:** OSF provides detailed contributor metadata, including permissions and contribution types, but does not align them with DataCite or ontology-based role taxonomies.
- **Identifier strategy:** OSF assigns internal node IDs rather than DOIs by default. DOIs may be assigned only for public registrations, complicating identifier mapping.
- **Keywords and subjects:** OSF uses tags only, without categorisation or semantic structure.
- **Limited metadata schema alignment:** OSF does not follow Dublin Core, DataCite, schema.org, or DCAT, making it the most challenging repository to align semantically.

OSF's flexibility benefits project organisation but introduces major challenges for dataset harmonisation and RDF transformation.

Semantic Mismatches Across Repositories

In addition to structural dissimilarities, the three repositories share a lot of differences when it comes to their manner of doing things. represent fundamental metadata concepts. For example:

- **Contributor roles:** Zenodo uses "creator," Figshare uses "authors," and OSF uses nested "contributors."
- **Subject fields:** Zenodo: `keywords, subjects` Figshare: `categories, tags` OSF: `tags`
- **Resource type classifications:** Zenodo uses DataCite resource types, Figshare uses custom labels ("figure," "poster," etc.), OSF uses general-purpose "file" types.
- **Identifier systems:** Zenodo: DOI + concept DOI Figshare: DOI OSF: internal node ID (optional DOI)

These inconsistencies hinder direct comparison and require semantic mappings.

Extended Comparison Table

To supplement Table 2 compares, semantic variations across the repositories.

Summary

The comparison shows that Zenodo is the nearest to accepted standards, whereas Figshare is not. proffers non-uniform hybrid frameworks, and OSF is well off of. standardised dataset models. The heterogeneity highlights the need to have a unified. semiotic integration strategy, making possible:

- schema normalisation via a DataCite pivot model,

Table 2: Semantic divergences in key metadata fields across Zenodo, Figshare, and OSF.

Concept	**Zenodo**	**Figshare**	**OSF**
Contributor Role	*creator*, *authors*	*authors*, *contributors* (nested)	*contributors*, *creators*
Subject / Keywords	*keywords*	*subjects*, *categories*	*tags*, *subjects*
Resource Type	DataCite types	Custom types (e.g., figure, media)	Generic file types
Identifier Scheme	DOI (DataCite)	DOI (DataCite)	OSF IDs; DOI optional
Versioning Metadata	Versioning + timestamps	Versioning + DOIs	Timestamps only

- ontology-aligned mappings to schema.org, DCAT, and Dublin Core,
- RDF graph construction for machine-actionable interoperability.

These steps form the foundation of the semantic integration pipeline developed in this dissertation.

2.4 Semantic Web and Ontologies

Semantic Web technologies offer a system of knowledge representation. form that can be understood by machines, so the systems can not just interpret the structure of. metadata, as well as its meaning. The concept was introduced by Berners-Lee et al. Berners-Lee et al. (2001), the Semantic Web extends the traditional web of documents into a web of data where entities, their attributes, and their relationships are encoded using formal knowledge representation languages. This approach is highly relevant for metadata integration because it enables heterogeneous metadata sources—such as Zenodo, Figshare, and OSF—to be expressed using shared vocabularies and linked through common identifiers.

Semantic Web technologies help to resolve the interoperability crises that have been tracked down. FAIR principles (I13) through supporting shared ontologies, qualified relationships, and machine-actionable data formats. The key elements are detailed in the subsequent subsections. of the Semantic Web stack with regard to integration of research metadata.

RDF and Linked Data

The Resource Description Framework (RDF) is the foundational data model of the Semantic Web. RDF represents information as triples—comprising a subject, predicate, and object— which together form a directed labelled graph.W3C (2014) This simple yet expressive structure enables:

- representation of heterogeneous metadata,
- linking between entities across repositories,
- incremental extension of metadata without schema redesign,
- integration of controlled vocabularies and ontologies.

For example, a dataset in Zenodo with a creator named "Jane Smith" can be represented as:

```
<dataset> dc:creator <orcid.org/0000-0002-1234-5678>
```

If Figshare also contains a dataset created by Jane Smith but uses a different internal identifier, RDF enables linking these two representations through shared external identifiers such as ORCID, thereby resolving identity conflicts across repositories.

Linked Data principles further extend RDF by requiring that entities be identified with URIs, dereferenceable on the web, and interconnected to form a global graph of knowledge.Bizer et al. (2008) When these principles are applied, metadata of Zenodo, Figshare, and OSF can be converted into a single, isolated JavaScript/XML fragments to one coherent unit. networked body of knowledge.

RDFS and OWL

RDFSchema (RDFS) and the Web Ontology Language (OWL) also allow the description of classes, properties, constraints, and logical relationships of a metadata domain. RDFS assists with the lightweight semantic structures by facilitating:

- class hierarchies,
- property hierarchies,
- domain and range specifications.

OWL is an extension of RDFS that offers more expressive features appropriate in automated applications. reasoning. These include:

- equivalence of classes (e.g., `schema:creator` *owl:equivalentProperty* `dc:creator`),
- subclass relationships (e.g., `dcat:Dataset` `rdfs:subClassOf` `schema:CreativeWork`),
- cardinality constraints,
- transitive, symmetric, and inverse properties.

In the context of this dissertation, OWL reasoning enables inference such as:

- If Zenodo marks a resource as `dataset` and Figshare marks a similar resource as `Fileset`, both can be aligned under the broader concept `dcat:Dataset`.
- If OSF identifies an entity as a project but it contains only files, a reasoning rule can classify it as a dataset analogue.

Such reasoning enhances metadata quality and supports FAIR Interoperability (I1–I3) by making implicit knowledge explicit.

Key Ontologies

Ontologies provide controlled vocabularies and formal schemas for describing entities and relationships. The following ontologies are particularly relevant for integrating research repository metadata:

- **schema.org** — Offers widely used properties for datasets, creators, licenses, and citations.Community (2024) Zenodo did not adopt it natively, but it is used nonetheless. Mapping to schema.org, Figshare or OSF, makes it more visible in search engines. encourages cross-connection of web platforms.
- **DCAT** — The Data Catalog Vocabulary enables structured descriptions of datasets, distributions, access URLs, and licensing.Commission (2020) It is especially useful for creating integrated data catalogues from heterogeneous repositories.
- **Dublin Core Terms (DCTERMS)** — A widely adopted set of terms for describing creators, contributors, subjects, identifiers, and dates.DCMI (2023) Dublin Core mappings are applicable in most repositories, and it is an appropriate one. point of convergence to harmonise metadata.
- **DataCite Ontology** — Formalises the DataCite metadata schema, allowing structured representation of DOIs, contributors, funding information, and related identifiers.Group (2021)
- **PROV-O** — The W3C Provenance Ontology offers a formal model to provenance. describing data set creation, modifications and derivations.Groth and Moreau (2013) This is Applicability to FAIR Reusability (R1.2), and version control, is high.
- **FOAF (Friend of a Friend)** — Supports representation of people, organisations, and affiliations, enabling alignment of contributor information across repositories.
- **SKOS** — A vocabulary for controlled subject classifications, beneficial for aligning free-text tags (e.g., OSF tags) with structured vocabularies.

Together, these ontologies enable rich semantic integration across repositories, supporting both conceptual alignment and automated reasoning.

Knowledge Graphs

Knowledge graphs represent interconnected entities and relationships in a structured, machine-readable form.Hogan et al. (2021)They act as a perfect tool to combining heterogeneous metadata because of their adaptability, expressiveness and capability. to facilitate semantic inferring.

A knowledge graph derived from repository metadata offers several advantages:

- **Semantic querying:** Researchers are able to write queries across using SPARQL. repositories (e.g., "find all datasets written by researchers in Ireland").
- **Inference:** New relationships, such as those of the form of inferent examples can be automatically inferred by OWL reasoners. as recognizing similar subjects or combining duplicate records on contributors.

- **Entity disambiguation:**External identifiers like ROR and ORCID permit. explicit association between individuals and institutions between repositories.
- **Graph enrichment:** Metadata can be linked to external knowledge bases such as Wikidata, Crossref, or OpenAlex for additional context.
- **FAIR compliance:** Knowledge graphs naturally align with FAIR principles by supporting machine-actionable interoperability.

Knowledge graphs have found extensive use in massive academic infrastructures, such as OpenAIRE, Europe PMC, and Crossref Event Data, they can prove their potential. to enhance the quality of metadata, discoverability and interoperability.

In this dissertation, the knowledge graph serves as the final integrated representation of harmonised metadata from Zenodo, Figshare, and OSF, enabling unified querying and FAIRness improvement.

2.5 Related Work

A growing body of research examines metadata harmonisation, semantic alignment, and FAIR evaluation across digital repositories and research infrastructures. While these efforts demonstrate increasing interest in machine-actionable metadata, they also reveal significant limitations, particularly in cross-repository interoperability and semantic integration. This section reviews key strands of work related to metadata aggregation, ontology-based alignment, Semantic Web applications in scholarly communication, and FAIR assessment tools.

Metadata Aggregation and Crosswalks

Early work on metadata aggregation focused primarily on syntactic harmonisation using schema crosswalks. Lagoze et al. Lagoze et al. (2006) discuss the challenges of given that, harvesting metadata of heterogeneous systems such as digital libraries. even apparently similar areas can be different in both semantics and granularity, as well as in constraints. The majority of the crosswalks that were developed during this era were based on manual schema to schema mapping. (e.g., Dublin Core to MARC, EML to FGDC), an implementation which cannot be scaled to large or evolving repositories.

Other studies such as Johnston et al. **?**propose normalising managed vocabularies or application profiles metadata. However, these methodologies tend to deal with domain repositories, as opposed to multidisciplinary. such as Zenodo, Figshare or OSF. Notably, crosswalk-based approaches are hard to use. add semantic reasoning or ontology alignment, which restricts their usability on. automated integration.

Repository Interoperability Initiatives

Several large-scale initiatives have attempted to improve interoperability across scholarly repositories. OpenAIRE, for example, aggregates metadata from thousands of European repositories using a harmonised application profile.**?** Similarly, Europe PMC and Crossref provide centralised metadata services for biomedical literature and scholarly publications.

These infrastructures however, are not direct solutions to heterogeneity of. data repositories: Zenodo, Figshare, and OSF. OpenAIRE partially is a union of Zenodo that

does not match Figshare or OSF metadata, and does not make use of. Semantic alignment of datasets based on knowledge graphs.

Ontology-based Metadata Integration

The Semantic Web community has developed numerous techniques for aligning metadata across heterogeneous sources using ontologies. Works such as Gruber **?** and Noy and McGuinness **?** emphasise the role of ontologies in establishing shared conceptual structures to mediate semantic differences. Ontology alignment techniques—such as equivalence mapping, hierarchical alignment, and lexical matching—facilitate cross-repository interoperability by ensuring that similar concepts map to shared vocabularies.

In the context of research data, the use of vocabularies such as Dublin Core, schema.org, DCAT, and PROV-O has been explored in projects such as Research Data Alliance (RDA) metadata recommendations and the W3C Dataset Exchange Working Group.**??** These attempts are very useful advice yet not put into effect by Zenodo, Neither Figshare, nor OSF showcase end-to-end pipelines of semantic integration. integrating harvesting, normalisation, RDF transformation and FAIR evaluation.

Knowledge Graphs in Scholarly Metadata

Knowledge graphs have recently gained traction in research infrastructures as a means of representing and interlinking scholarly artefacts at scale. Hogan et al. Hogan et al. (2021) demonstrate how KGs support semantic integration, entity disambiguation, and reasoning.

Such infrastructures as the Research Graph by OpenAIRE, Crossref Event Data, and OpenAlex. demonstrate the importance of graph-based metadata representation, nevertheless, such systems. instead of integrating bibliographic metadata (articles, authors, citations), primarily. data set and software metadata in repositories. In addition, they fail to carry out semantic. alignment of repository-specific schemas, and they are not ingesting the heterogeneous structures. of Zenodo, Figshare, and OSF.

FAIR Metadata and Machine-actionability

Greenberg Greenberg (2005) argues that semantic richness is essential for ensuring machine-actionable metadata, especially for long-term preservation. Machina et al. and Sansone et al. emphasise the need for semantic interoperability and controlled vocabularies to achieve FAIR compliance.Mons (2020); **?**

Automated FAIR evaluation tools such as F-UJI Pang et al. (2022) and FAIR-Checker Research (2023) provide frameworks for assessing FAIRness but have limitations:

- they analyse existing metadata but do not modify or integrate metadata;
- they do not harmonise heterogeneous schema fields;
- they cannot infer semantic relationships across repositories;
- they rely on repository-provided metadata, which is often incomplete.

Thus, while FAIR assessment tools highlight deficiencies, they do not provide solutions for addressing them.

Metadata Quality and Heterogeneity Studies

Several studies explicitly analyse metadata quality in repositories. Chen et al. **?** examine inconsistencies in keywords, identifiers, and contributor fields across major data repositories, concluding that heterogeneity remains a major barrier to reuse. Fang et al. **?**examine subject classifications, with wide use. of free-text tags which minimize interoperability. Wilkinson et al. and Mons et al. point out that principles of FAIR are impossible to complete. in the absence of semantic grounding and ontology-based representation.Wilkinson et al. (2016); Mons (2020)

Yet, the studies usually end with the heterogeneity determination, they do not put into practice. A top-to-bottom semantic integration pipeline.

Gaps in the Literature

Across the reviewed literature, several gaps become evident:

- No prior work integrates **Zenodo, Figshare, and OSF metadata together**.
- Existing studies focus on **single repositories** or **domain-specific collections**, not multidisciplinary platforms.
- Most integration approaches rely on **syntactic crosswalks**, not semantic alignment.
- No studies implement a **DataCite pivot schema** as a harmonisation layer.
- Few works transform heterogeneous metadata into a **unified RDF knowledge graph**.
- FAIR assessment is rarely applied **before and after** semantic enrichment.

Contribution Beyond State of the Art

This dissertation addresses these gaps by:

- harvesting metadata from Zenodo, Figshare, and OSF using heterogeneous APIs;
- normalising all metadata into a DataCite-based pivot model;
- aligning fields with shared ontologies (schema.org, DCAT, Dublin Core, DataCite);
- constructing a unified, cross-repository RDF knowledge graph;
- applying reasoning to enhance metadata richness and semantic coherence;
- evaluating FAIRness *before and after* integration using FAIR-Checker.

To the best of current knowledge, this constitutes the first complete semantic metadata integration framework spanning these three major open research repositories and the first study to empirically measure the FAIRness improvement enabled by semantic enrichment.

3 Methodology

The methodological framework that was used in the design, implementation and adoption of the chapter is outlined in this chapter. evaluate the semantic metadata integration pipeline that has been developed in this dissertation. The research is a multi-stage process, which is organized and involves approaches of. Semantic Web modelling, metadata engineering and quantitative FAIR. assessment. The fact that the project is aimed at harmonising heterogeneous metadata originating. identified three major open research repositories, i.e. Zenodo, Figshare, and the Open science. Framework (OSF)- A strong and congruent methodology is required to embrace structural. manifold, resolve semantic contradictions and find what the effects of integration are. quality of machine-actionable metadata.

The approach is grounded in a combination of **qualitative** and **quantitative** methods. The conceptual differences between different were comprehended through viability schema analysis. contributor modelling, determine the structural divergences in contributors, subject classification, licensing domains, and provenance representation, and extract a single. DataCite-based pivot schema. The changes were measured using quantitative methods. in metadata Fairness pre- and post-semantic enrichment with automated evaluation. tools. This is a mixed-methods approach that will guarantee the conceptual reliability and the. The effectiveness of the pipeline of the integration is empirically proven.

The methodology consists of five major components: (i) harvesting heterogeneous metadata from multiple repositories; (ii) analysing and normalising metadata into a unified pivot model; (iii) applying semantic enrichment through ontology alignment and RDF modelling; (iv) constructing a unified RDF knowledge graph; and (v) evaluating FAIRness improvements resulting from semantic integration.

Each component is described in detail in the subsequent sections. The combination of data engineering practices, Semantic Web technology, and FAIR assessment ensures that the resulting knowledge graph is both semantically coherent and demonstrably aligned with machine-actionable metadata principles.

3.1 Research Design

The study has a structured and multi-phase design based on metadata principles. FAIR assessment, engineering and Semantic Web modelling. The main purpose is to make sure that the first goal is achieved. is to harmonize heterogeneous metadata that is inputted at Zenodo, Figshare and OSF, the design focuses on semantic rigour, transparency and reproducibility. The workflow is iterative in form, and every stage is based on the output of the last one, however. has a logically sequential structure that is appropriate to systematic evaluation.

Figure 1 illustrates the end-to-end pipeline.

The design consists of five major phases:

1. **Phase 1: Repository Metadata Harvesting** This phase focuses on programmatic metadata acquisition from Zenodo, Figshare, and OSF. Because each repository exposes metadata through different endpoints (REST APIs, OAI–PMH feeds, and hierarchical JSON structures), custom harvesters were developed to retrieve metadata in their native formats. This ensures that the raw dataset reflects real-world heterogeneity rather than pre-curated samples.

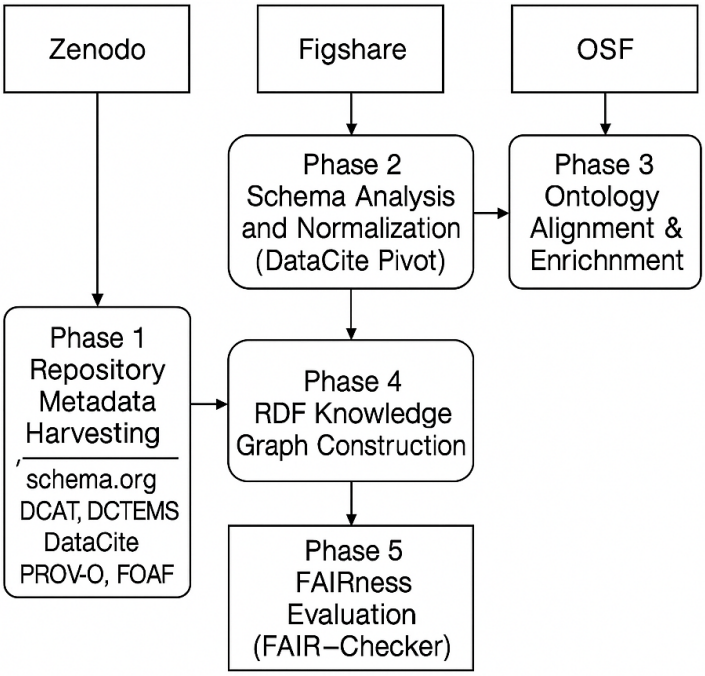

Figure 1 Overview of the multi-phase research design, from repository metadata harvesting through schema analysis, ontology-based semantic enrichment, RDF knowledge graph construction,

Figure 1: Overview of the multi-phase research design, from repository metadata harvesting through schema analysis, ontology-based semantic enrichment, RDF knowledge graph construction, and FAIRness evaluation.

2. **Phase 2: Schema Analysis and Normalisation** The harvested metadata were analysed to identify structural variations, inconsistent field names, and conceptual differences in metadata models. This qualitative schema analysis informed the design of a **DataCite-based pivot schema**, which served as a harmonisation layer. Mapping rules were constructed to normalise all repository metadata (e.g., mapping *authors* → *creator*; *categories* → *subjects*). This phase provides the structural foundation for semantic integration.

3. **Phase 3: Ontology Alignment and Semantic Enrichment** Pivot-schema fields were semantically augmented with the mapping of the normalised metadata. common ontologies containing:

   - schema.org (web-oriented dataset semantics),
   - DCAT (cataloguing and dataset distribution),
   - DCTERMS (core metadata terms),
   - DataCite Ontology (identifiers, contributors, funding),
   - PROV-O (provenance),
   - FOAF (people and organisations).

   OWL reasoning and rule-based inference were applied to infer additional relationships, such as equivalence between contributor identities or dataset classifications. This phase operationalises FAIR Interoperability (I1–I3).

4. **Phase 4: RDF Knowledge Graph Construction** The enriched metadata were serialised using the RDF model (Turtle format) and integrated into a triple store (Apache Jena Fuseki). The resulting knowledge graph captures harmonised metadata across all three repositories and enables SPARQL-based querying, semantic reasoning, and cross-repository discovery. This graph serves as both the analytical artefact and the final output of the semantic integration process.

5. **Phase 5: FAIRness Evaluation** FAIR-Checker was used to evaluate FAIRness *before and after*semantic enrichment. This gives a numerical value of metadata improvements, in Interoperability (I1–I3) and Reusability (R1.1, R1.2) in particular. Results were to be triangulated with qualitative evaluations that will guarantee validity and interpretability.

It is a combination of schema analysis, ontology alignment, RDF modelling, and FAIR evaluation. this multi-phase research design assures a rigorous, reproducible, and empirically valid research design. grounded methodology. It directly realises the principles of FAIR, in particular. machine-actionability and semantic interoperability–and dealing with the real world. heterogeneity of large open research repositories.

3.2 Data Collection

This research had the data collection phase to provide the empirical basis of the study and was planned. in order to obtain the entire range of metadata heterogeneity in the context of Zenodo, Figshare, and the Open Science Framework (OSF). Open research repositories vary in several aspects because they are open. programmatic harvesting

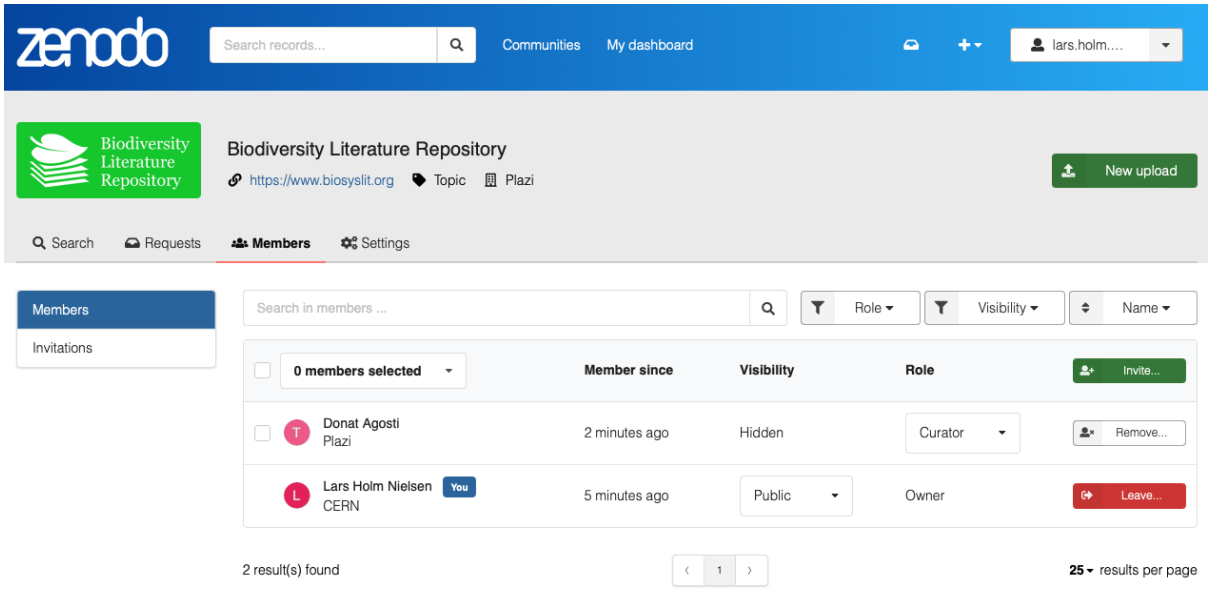

Figure 2: Zenodo Web Interface

was critical how they organise, expose, and serialise metadata. to achieve a realistic dataset of real community practices, user -deposited. metadata variations, and schema constraints of a given repository. Rather than relying on data that was exported manually or curated subsets, metadata was harvested using APIs. were restored in their own original character–raw, pure, and typical of the user-created and constantly changing content hosted by each platform.

Conceptually, the data collection was aimed at two points: (i) to obtain a large and broad enough collection of metadata records to facilitate powerful. semantic integration and analysis; and (ii) to reveal the structural, lexical, and ontological inconsistencies that impede. Cross editorial interoperability. These contradictions–appearing in the names of the fields, contributor representation, subject tagging, identifier use, and versioning conventions- provide the analytical foundation of the semantic harmonisation pipeline created in later. chapters.

Zenodo

Zenodo provides one of the most multi-layered and broadest metadata infrastructures. of open research repositories. It does reveal metadata through two main mechanisms:

- **REST API (JSON)** — Enriched wealthy, dense metadata such as contributors. Total of 16 elements in the format document: affiliations, funding reference, related identifiers, resource type, version licensing, community classifications and history.
- **OAI–PMH (XML: Dublin Core, MARCXML, DataCite)** — Standards-based harvesting suitable for interoperability but lacking some details present in the JSON API.

For this research, the REST API was selected as the primary harvesting mechanism due to its:

- more expressive metadata structure,

- higher completeness compared with OAI–PMH feeds,
- consistent availability of contributor roles,
- direct support for version-specific and concept DOI identifiers.

Zenodo's concept DOI system introduces semantically meaningful relationships between versions of the same dataset. Capturing both versioned DOIs and concept DOIs was crucial for modelling provenance (via PROV-O) and linking related resources during semantic integration.

A stratified sampling approach was embraced to have variability in the Zenodo subset. Over 500 records were accessed based on the different types of resources such as datasets. software packages, workflows, and conference papers, reports, and multimedia artefacts-cutting across a variety of years of publication and community organizations. Parameters of query has been used. so that it can be balanced in its coverage and not create bias against one of the communities or topics. Rate-limit Automated exponential backoff retry logic was used to address constraints.

Figshare

Figshare presents a unique set of challenges for metadata harvesting due to its flexible and user-driven metadata model. It exposes two interfaces:

- **REST API (custom JSON)** —Plentiful metadata, such as file-level information. categories, tags, custom fields, versioning information, embargos and institutional. affiliations.
- **OAI–PMH (DataCite-like XML)** — Less detailed but aligned more closely with traditional repository harvesting standards.

The REST API was chosen for harvesting because it exposes metadata elements not available through OAI–PMH—particularly user-generated tags, hierarchical categories, and file-level structures. Figshare metadata often exhibit inconsistencies between categories and tags: categories follow a controlled vocabulary, whereas tags are free-text and often include abbreviations, synonyms, or domain-specific jargon.

To accommodate these irregularities, **both fields were harvested** to support subsequent natural language processing, lexical matching, and ontology alignment.

More than **500 Figshare items** were collected across datasets, images, figures, tables, posters, reports, and supplementary files. Cursor-based pagination and token-based API authentication were handled programmatically, and additional validations were implemented to detect missing or malformed records—common in Figshare's user-dependent metadata environment.

OSF

OSF differs substantially from Zenodo and Figshare in its conceptual model. Rather than treating datasets as standalone deposits, OSF organises research outputs within project-centric structures consisting of:

$$\textit{projects} \rightarrow \textit{components} \rightarrow \textit{files} \rightarrow \textit{contributors.}$$

This nested structure involves multi-level traversal to put together a dataset-like. metadata representation. In comparison to Zenodo and Figshare, OSF does not make use of OAI. making its REST API the only source to metadata acquisition.

Key challenges in harvesting OSF metadata included:

- navigating deeply nested JSON structures,
- resolving inherited contributor roles,
- distinguishing between project-level and file-level metadata,
- flattening hierarchical metadata into normalised, analysis-ready records.

Approximately **500+ OSF resources**were gathered in various fields and metadata degrees of completeness. Further measures were set in place to deal with OSF-specific. such features as linked registrations, forked projects, embargoed materials and parent and child aspects of relationships.

Harvesting Tools

All harvesting was implemented using a Python-based pipeline tailored to the combined requirements of REST and OAI–PMH endpoints. Key libraries included:

- `requests` — managing HTTP requests, authentication, pagination, and rate-limits;
- `xml.etree.ElementTree` — parsing OAI–PMH XML metadata from Zenodo and Figshare;
- `json` — transforming nested API responses;
- `pandas` — structuring, cleaning, and validating metadata;
- `logging` — recording harvesting activity for reproducibility and debugging.

The harvesting pipeline implemented extensive robustness features, including retry mechanisms, error handling, schema validation checks, timestamped logs, and automatic serialisation of raw metadata for auditability.

In total, the final combined dataset consisted of **over 1,500 heterogeneous metadata records**,accommodating the range of structural and semantic variation among three huge research archives. It was based on this rich and diverse data. to schema analyze, semantically align, construct an RDF knowledge graph and FAIR. assessment at later stages of methodology.

3.3 Sampling

Purposeful sampling was embraced to warrant that the metadata that was gathered was comprehensive. range of structural and semantic diversity found in Zenodo, Figshare and OSF. In contrast to a probability-based sampling which focuses on aiding statistical generalisation, purposeful sampling should be used when one wants to intentionally sample variation, edge cases, and heterogeneity in records known to be different in structure, quality, and multimedia interpretation. (Patton; 2015). In the context of semantic integration, sampling is not an peripheral design choice but a fundamental determinant which

influences. the correctness of schema mapping, ontology matching and interoperability testing.

The importance of sampling in this study is threefold. First, the reliability of the pivot schema depends on exposure to diverse metadata forms; if the sample is narrow, the derived schema risks overfitting to a subset of repository practices. Second, semantic mapping quality can only be evaluated meaningfully when the sample includes inconsistent, incomplete, and user-generated fields that reflect open repository realities. Third, the generalisability of the *pipeline itself*—in engineering terms—requires stress validation rules and transformation logic testing against heterogeneous records, such that transformation logic can be used to perform these tests. RDF results are resistant to real-world variability. Hence, the purposeful sampling does. not seek to make inference about prevalence in repositories, it strengthens. **analytical and engineering generalisation** by ensuring the model of integration is resilient across metadata complexity and irregularity.

In designing the sampling strategy, five criteria were prioritised:

- **Diverse repository resource types:** and containing datasets, software, Graphics, illustrations, work flows, posters and project elements.
- **Contributor structures:** from single-author deposits to large multi-contributor collaborations, including varied use of ORCID identifiers.
- **Licensing models:** open licenses (e.g., CC-BY, CC0), restricted licenses, and cases with missing or ambiguous licensing.
- **Subject classifications:** keywords, tags, categories, disciplinary labels, and user-defined subject terms.
- **Metadata completeness levels:** ranging from richly annotated records to sparsely populated ones with minimal fields.

Sampling was not intended to infer repository-wide patterns but to ensure that the semantic integration process could generalise across metadata complexity and irregularity. The inclusion of diverse record types and metadata profiles was particularly important because repositories differ substantially in how they structure and encode metadata.

Sampling Rationale

The primary rationale behind the sampling strategy was to achieve **schema representativeness**. Semantic integration exposes hidden mismatches between repositories only when the sample includes records that reflect the true heterogeneity of each platform. For instance:

- Zenodo includes datasets, software, conference materials, and complex versioned records with concept DOIs.
- Figshare includes images, figures, posters, and custom item types not found in other repositories.
- OSF includes projects, components, and registries arranged in hierarchical structures where metadata propagate across parent-child relationships.

Capturing such differences ensures that the pivot schema, ontology mappings, and RDF representations developed later in the research are robust and reusable. Purposeful sampling also helps evaluate how well the semantic pipeline manages inconsistent, incomplete, or user-generated metadata fields—conditions typical of open repositories(Etikan et al.; 2016) .

Additionally, a balanced number of records (∼500 per repository) was selected to avoid nepotism on the side of one platform and permit comparative FAIRness analysis across repositories. This balanced architecture was required to have meaningful cross-repository interoperability. evaluation.

3.4 Ethics

Ethical considerations were integrated throughout the research process to ensure compliance with legal frameworks, repository policies, and responsible data-handling practices. Because the study involves harvesting publicly available metadata rather than full datasets or private user information, ethical risks were inherently low. However, explicit attention was given to issues around data protection, licensing, attribution, and responsible reuse.

Use of Public Metadata

All the metadata collected at Zenodo, Figshare and OSF were publicly available at the. time of collection. By definition, metadata are descriptive information of. objectives of research and not the objects. In this way, this research did not gain access. or operate raw datasets, restricted files or any other content which needs authentication. No sensitive personal information (e.g. email addresses, institutional identifiers) was surveyed; only publicly visible contributor names and ORCID identifiers were taken into consideration.

Compliance with Repository Terms

Each repository explicitly permits harvesting of metadata for academic, non-commercial, and interoperability-related purposes. API usage adhered strictly to:

- accepted rate limits,
- permitted request volumes,
- repository-specific attribution norms,
- published API documentation and usage guidelines.

No unauthorised scraping techniques, web crawling outside approved endpoints, or circumvention of access controls were employed. All harvesting was performed through documented and officially supported interfaces.

Data Protection and Privacy

Although metadata records include contributor information, these fields are intended for public dissemination and citation. Nonetheless, the project adhered to GDPR-aligned principles by:

- avoiding processing of sensitive personal data,
- ensuring ORCID identifiers were used only for semantic linking,
- storing all harvested metadata in secure, access-controlled environments,
- removing identifiers in metadata to be used in examples or visualisations.

No profiling, behavioural inference, or individual-level analysis of contributors was conducted.

Reproducibility and Transparency

To ensure methodological transparency and reproducibility, the entire harvesting, transformation, and semantic integration pipeline was documented extensively. This included:

- code-level documentation for all scripts,
- timestamped logs of API requests and responses,
- version-controlled scripts and configuration files,
- storage of raw and processed metadata for auditability,
- detailed description of the workflow in this dissertation.

These measures ensure that other researchers may replicate, extend, or independently validate the data collection framework.

3.5 Tools and Environment

The software tools utilized were in a detailed mix in order to implement the system. computational environments, and programming libraries, and metadata to aid harvesting. RDF generation, ontology alignment, cleaning, and FAIR evaluation. A modular design was adopted in order to be extensible and that additional repositories or ontologies may be. minimal reconfiguration is incorporated.

Programming Environment

Python 3.10 served as the primary environment for data processing and semantic modelling. Several specialised Python libraries were used, including:

- `requests` —to communicate with REST APIs and do authentication, pagination, and retry logic;
- `pandas` — for metadata cleaning, restructuring, normalisation, and quality assessment;
- `rdflib` — for creating RDF graphs, generating triples, and exporting Turtle/JSON-LD serialisations;
- `xmltodict` and `xml.etree` — for parsing OAI–PMH XML feeds;

- `SPARQLWrapper` — for SPARQL endpoint querying and validation;
- `pyRDF2Vec` — optional library for generating semantic embeddings.

These tools enabled automated data extraction, processing, and semantic transformation at scale.

Ontology Mapping Environment

The combination of semantic enrichment and ontology alignment was carried out. specialised ontology engineering tools:

- **Protégé** — OWL ontology development, validation of alignment and reasoning with inbuilt reasoners like HermiT;
- **ROBOT** —to validate, merge, and transform ontologies automatically. tasks;
- **WebVOWL** — for interactive visual inspection of ontology structures and relationships.

These tools ensured that mappings to DataCite Ontology, schema.org, DCAT, DCTERMS, FOAF, and PROV-O were consistent, logically valid, and semantically coherent.

Triple Store

The final RDF knowledge graph was deployed using two graph store platforms:

- **Apache Jena Fuseki** —Base SPARQL endpoint of integration, experimentation, and questioning;
- **GraphDB Free Edition** — used for comparative reasoning performance and qualitative evaluation of inferred relationships.

Both triple stores accept SPARQL 1.1 and reasoning as well as inference rules. massive graph management. The robustness and were evaluated in multiple graph stores. generation RDF portability.

FAIRness Evaluation Environment

A combination of automated tools was used to test Metadata FAIRness:

- **FAIR-Checker** — for machine-actionable FAIR evaluation and semantic validation;
- **F-UJI** — to be cross-tool verified, independent;
- Python and `jq` scripts — for parsing, comparing, and visualising pre- and post-integration FAIRness metrics.

Using multiple evaluation tools provided methodological triangulation, improving confidence in measured FAIRness improvements.

Hardware and Cloud Environment

A hybrid local–cloud infrastructure supported development, testing, and large-scale graph generation:

- **Local workstation:** 16GB RAM, AMD Ryzen 7 CPU — used for development, testing, and lightweight processing.
- **AWS EC2 (t3.large):** Ubuntu 22.04 LTS — used for executing large-scale metadata transformations, ontology reasoning, and triple store deployment.
- **50GB EBS storage:** dedicated to storing harvested metadata, RDF files, logs, and intermediate data products.

This environment ensured scalability, stability, and performance throughout all stages of the integration pipeline.

4 System Design

The architectural design used in this chapter gives the design of the system. The framework semantic metadata integration framework created during this dissertation. Its central purpose is to show the heterogeneity of metadata that is generated by Zenodo, Figshare and the. Open science frameworks (OSF) undergo processing and normalisation, semantic enrichment, and lastly. assessed against FAIR competence. The design is based on best layered pipeline model. metadata engineering practices, Semantic Web architecture practices, and FAIR-aligned data. infrastructuresWilkinson et al. (2016); Berners-Lee et al. (2001); Hogan et al. (2021). Each layer carries out a specific logical task as part of transforming the whole of. store metadata as a model in a common machine action form.

The system itself is designed to be modular so that individual functional blocks like harvesting can be made. to operate, normalisation, ontology alignment, RDF generation, and FAIR evaluation. independently. This modularity is in favor of extensibility, so that more new can be added. minimal incorporation can be done with repositories, ontologies or evaluation tools. reconfiguration. The systematic use of W3C is maintained to ensure that standards-compliance is adhered to. such technologies as RDF, RDFS, OWL, SPARQL and universally recognised vocabularies. as schema.org, DCAT, Dublin Core, PROV-O and the DataCite Ontology. Through these design decisions, the system optimises semantic interoperability and is made to be aligned with. FAIR principles, specifically Interoperability (I13) and Reusability (R1.1 -R1.3).

To illustrate the overall workflow, Figure 3presents the high-level architectural diagram that has been used in this dissertation. The diagram captures the sequential development of metadata harvesting to FAIRness evaluation, reflecting how to convert heterogeneous XML/JSON metadata into a single RDF knowledge graph. onto alignment and optional reasoning are used. The architecture also highlights the way FAIR. Metadata improvements are measured using the resulting RDF graph to evaluation tools. interoperability and quality.

There are five main stages of the architecture. To begin with there are repository metadata. retrieved in case of the existence of REST APIs and OAI-PMH interfaces. This step captures the intrinsic heterogeneity of every platform: The Zenodo platform can be accessed through an API exposing DataCite-like JSON, Figshare supports custom

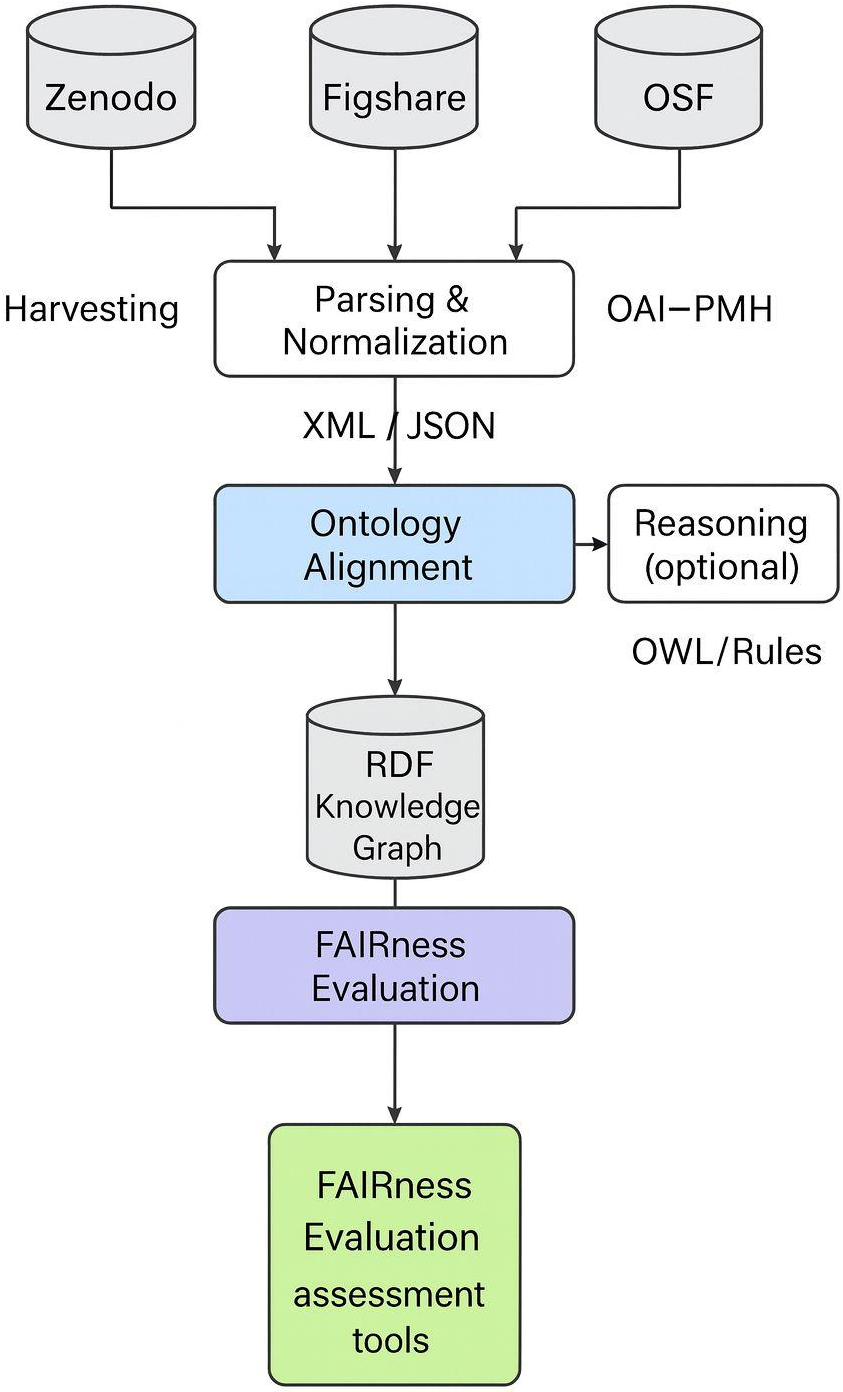

Figure 3: High-level architecture of the semantic metadata integration pipeline, illustrating the flow from harvesting to semantic enrichment, RDF knowledge graph construction, and FAIRness evaluation.

JSON structures as well as a OAI-inspired DataCite-type OAI-PMH. endpoint, instead of a hierarchy with highly nested metadata that are represented as a JSON object at OSF. CERN OpenAIRE (2024); Figshare (2024); for Open Science (2024). In the second

stage, all harvested metadata are parsed, filtered, and normalised into a common pivot schema based on the DataCite Metadata Schema v4.4 Group (2021), ensuring structural consistency across all records.

The third stage is the ontology alignment whereby, each of the normalised fields is mapped to. a relational concept in one or more semantic web vocabularies. This mapping embeds machine-readable semantics through connection between contributors of FOAF agents, datasets of DCAT and PROV-O provenance relations to schema.org Dataset classes.W3C (2014); Noy and McGuinness (2001). Extra rules of optional reasoning enhance the metadata by providing additional relationships like. provenance of datasets, provenance, provenance.

Semantically compatible metadata are then transformed into RDF triples and at the fourth stage. joined together to form special RDF knowledge graph. The graph is applied on three-storey. Create a query list of the graph data using SPARQL, (Apache Jena Fuseki or GraphDB) to infer and cross-link graph data. repository exploration. The isolated metadata silos are converted in this process to an. interrelated graphical framework to enable discovery on Zenodo, Figshare, and. OSF.

The final one is FAIRness assessment where the apps of FAIR-Checker and F-UJI are employed. are applied on metadata following and during integration. This will help the system to quantify the degree to which. ontology correspondence and semantic regularisation enhance actionability of the machine, extensive metadata compliance of metadata in the repository with FAIR, and interoperability. Research (2023); Pang et al. (2022); Mons (2020).

These phases of architecture give a consistent channel of transformation. standardised metadata sources heterogeneous metadata sources to a standards-aligned richly interconnected and FAIR-aligned metadata. enhanced knowledge graph. The rest of this chapter elaborates on each of the key elements in. greater detail.

4.1 Architecture

The architecture of the semantic metadata integration framework follows a pipeline-oriented design that reflects the sequential and interdependent nature of metadata transformation in heterogeneous environments. It is constructed as a layered system in which each stage performs a specialised function while contributing to the broader objective of producing a unified, semantically rich, and machine-actionable representation of research metadata sourced from Zenodo, Figshare, and OSF. This architectural approach aligns with recognised patterns in data integration and Semantic Web engineering, where clear separation of concerns and modularity are essential for managing complexity and ensuring long-term maintainability Berners-Lee et al. (2001); Hogan et al. (2021). The harvesting layer will have an interaction with the external at the beginning of the pipeline. OAIendpoints and repositories of REST API. It is the layer that is in charge. Metaphysically breaking down metadata as JSON or XML documents, and denoting the innate differences in schema of depositories. Zenodo exposes JSON metadata of DataCite compatible metadata; Figshare presents metadata to its generalized category based and tag based schema; and OSF supplies. bulky format of deep-nested JSON format that resembles its project-component-file format. The then the layer harvesting becomes extremely important to the preservation of the authenticity and. organizational specifics of every store, so that the system will reflect the extent of the same. of metadata heterogeneity.

Once harvested, metadata progress to the normalisation layer, where repository-

specific schemas are harmonised into a unified pivot schema grounded in the DataCite Metadata Schema v4.4 Group (2021). It is a framework base of stability provided by this switch model. integration in its common sense in research infrastructures and it is evident. identifiers, contributor, resource type, subject and provenance representation. Normalisation eradicates disparities in field names, reorganizes hierarchy or structure-specific repository, and collects metadata objects into a logical and interoperable form. The step is significant because it converts the disparate metadata. patterns to a coherent representation, and systematic semantic enrichment at the downstream.

The semantic enrichment layer is an addition of normalised pivot schema which links metadata. elements build to formal ontology. It is a step that introduces machine readable semantics. through mappings to common vacabularies, e.g. schema.org dataset. DCAT of data-set, descriptions, Dublin Core Terms (DCTERMS) of general-purpose metadata, Dublin Core metadata. expressing the provenance is with the help of FOAF to represent the contributors as agents, cataloguing and finally, expressing provenance is the work of PROV-O. and DataCite Ontology of identifiers and funding relationships as well as versioning. Through this, semantically related entities, e.g. subject, authorship. classification, lineage of data sets, licensing, and distribution - are represented formally in RDF. The interoperability approach by the system is centred in the semantic enrichment layer. transforming syntactic metadata into knowledge level presentations queryable, connected out, and associated with external Linked Data sources.

After semantic alignment, the enriched metadata are converted to triples of RDF and stored. in a triple store setting. Serialisation of enriched metadata is the responsibility of this layer. to popular RDF data like Turtle and JSON-LD and incorporating them into a scalable. graph database e.g. Apache Jena Fuseki or GraphDB. The triple store offers a SPARQL. endpoint to facilitate query based interaction, reasoning, inference as well as cross repository. analysis. It also has named graphs, which conserve repository provenance and allow. integrated querying all integrated metadata. The RDF knowledge graph is therefore the one that acts as the. key point of the system that allows formulating relationships and exploring them dynamically. assisting automated reasoning by OWL-based inference rules.

The last architectural component layer is the FAIR evaluation layer which measures the. degree to which the semantic integration enhances FAIRness of metadata. Tools such as FAIR-Checker and F-UJI are used on records of pre-integration and post-integration to test. dimensions like interoperability, reusability, metadata richness, identifier quality and machine-actionability. Comparison of the raw harvested metadata and the enriched RDF. representations gives experimental support to the nature of semantic technologies and ontology. alignment promotion of FAIR in action.Research (2023); Mons (2020). This evaluation layer completes the pipeline by linking architectural decisions with measurable outcomes, demonstrating the practical value of the system beyond theoretical soundness.

The architectural design has its design on several general principles, which are general in the architectural design. resilience, elasticity and data conformity to system in an international way. Modularity as long as all the steps - harvesting to FAIR evaluation - can be re-configured or extended. but disruption in the entire work-flow. The realized one is the semantic fidelity. that the changes are not made senseless, decontextualized, and disdetermined. the original metadata. The system may be expanded to provide new repositories. ontologies that are slightly differentiated on an individual basis. Standards compliance is enforced through vehement use of the W3C Semantic Web technologies which can equally be advanced. with the help of the current research infrastructures and

Linked Data ecosystems. Finally, interoperability The architecture is superimposed with is and all the metadata transformations are based on it. common ontologies, which allows cross-repository and machine-actionable to be discovered. integration.

The combination of these architectural principles and processes together create the system to accomplish. its main purpose: to build a single and ontology-oriented knowledge graph that. integrates information in Zenodo, Figshare and OSF and exhibits enhanced FAIR-compliant. metadata quality.

4.2 Module Design

There are five fundamental modules that make up the system and that in totality serve the entire semantic. integration workflow. The first one is harvesting module which performs. metadata programmatic retrieval with the help of the REST API and OAI-PMH interfaces. This module supports pagination, rate limits, authentication, error recovery and raw metadata storage. Its nature makes the extraction of massive amounts of metadata reproducible. adjusting to the structural disparities between Zenodo, Figshare, and OSF.

The normalisation engine, which is the second module, converts heterogeneous metadata into a. integrated schema according to the DataCite 4.4 schema. This module sorts out discrepancies in. contributor roles, field names, subject classifications, as well as licensing information. It also integrates identifiers, joins associated data and re-organizes hierarchical or processed metadata into uniform and dataset like representations. By imposing a common schema, the normalisation module avails a logical basis of semantic enrichment.

The semantic enrichment engine is the third module, in which the fields of the unified metadata are mapped. to the classes and properties of ontology. This involves the matching of creators to FOAF. agents, mapping datasets to schema.org classes and classes of DCAT representing provenance with ¡—human—¿agents, mapping datasets to schema.org classes and DCAT classes, depicting provenance with PROV-O, and the connection of identifiers with DataCite Ontology. The enrichment process introduces formal semantics, making it possible to do automated reasoning and cross-repository. interoperability. The metadata are made machine-actionable at this point.

The fourth module, the RDF serialisation component, converts semantically enriched. metadata into RDF triples. These are namespace binding, URI generation, URIs construction. per three times and combining fragments of the graphs into full RDF files. RDF serialisation supports and has numerous syntaxes, including Turtle and JSON-LD. dissimilar semantic tools and triple stores.

The fifth one is knowledge graph engine and loads the RDF triples in a triple. —store Apache Jena Fuseki/GraphDB. The endpoint to the SPARQL is available in the module. asking is a method of making inferences and reasoning, which is marked on graphs. integrated graphs had been used with repository as well. It provides the reasons of doing things. generating semantic queries and ontology matches and provenance check. relations and advising FAIRness test.

4.3 Data Flow Diagrams (DFDs)

Data Flow Diagrams give a conceptual picture of the flow of metadata in the system. The Level 0 DFD is the overall semantic integration pipeline as a process consisting of one. accesses external repositories as sources of data and FAIR assessment tools as

evaluation endpoints. In this top-end perspective, repositories provide metadata which is fed out. The integration system, whereas the outputs of the system are enriched RDF and FAIRness metrics.

Level 1 DFD breaks down the integration system into internal processes in discrete parts. These contain metadata collect, schema normalisation, ontology mapping, generation of RDF, FAIR appraisal, and graph populate knowledge. The performance of any process is dependent. the previous step that comes before it is a reassurance of traceability and controlled pipeline of transformation.

Level 2 DFDs explain the details of the internal functioning of each of the major stages being more detailed. For example, in the harvesting process, there is pagination management, parsing of the JSON/XML and field. extraction. Normalisation entails mapping repository specific fields into pivot-schema. equivalents, solving ambiguous fields, consolidating the contributor and subject data. Ontology mapping also involves solving entity identities, ontology classes and others. proving relationships of provenance. Namespace binding and triple are part of RDF generation. whereas the triple store process is involved in indexing, reasoning, and query, validation is involved. execution. These step-by-step flows explain the flow of metadata in the system and how. information remains faithful during every transformation.

4.4 Unified Schema Design

It was necessary to have a single schema to overcome the structural disparities in Zenodo. Figshare, and OSF. DataCite Metadata Schema (v4.4) was chosen as the pivot model because of the following reasons. to its extensive usage, clear encouragement of dataset citation, well defined contributor. roles, representation of type of resources and compatibility with widely used ontologies. The unified schema establishes basic metadata elements of identifiers, creators, titles, etc. descriptions, types of resource, topics, dates, licensing, funding. acknowledgements, and similar identifiers. The fields of metadata in each of the repositories were mapped to this normalisation pivot schema so that downstream semantic operations were possible. depend on the predictability and well structure of input.

The unified schema also has multilingual titles and descriptions that are more enriching. in cases where repositories offer a variety of language, representation of materials. versions. The identification of related identifiers of the schema pivot, including DOIs, concept DOIs, etc. OSF node identifiers, and links to older versions, were especially helpful to semantic integration, because it enabled versioning, citation chains and dataset relationships. to get expressed explicitly in the RDF graph.

4.5 Ontology Mapping

Ontology mapping is a procedure that is associated with semantic enrichment. Behaviourally equivalent Metadata fields harmonizations over common ontologies have been performed repository wide to accomplish this. machine-actionable interoperability. Mapping was mostly based on the properties and classes. as schema.org, DCAT, Dublin Core Terms (DCTERMS), the DataCite Ontology, FOAF and, The creators were represented as indicatively modeled as FOAF agents, and towards the datasets. and data classes and versioning of schema.org and DCAT. The demonstration of the attribution and identifiers was achieved as PROV-O. Ontology has been created in a manner that it can be

interrelated.

Nor was semantic richness lost in that they were modeled in mappings. For example, a system of DOI, similar to the Zenodo system, used to identify different versions of the same data, was translated to the revision relations of PROV-O. The Figshare created tags and categories utilized by the user. were transformed to schema.org controlled subject representations in the form of keywords. The hierarchical project structures, at OSF, were flattened and remained provenance linked to. parent components. It was possible to construct a piece of knowledge out of the combination of these mappings. graph in which objects and relationship of any repositories are coherently established.

4.6 Triple Store Design

The final component of the system design is the RDF triple store that hosts the integrated knowledge graph. Apache Jena Fuseki was used as the primary triple-store environment due to its support for SPARQL 1.1, its robustness with large RDF datasets, and its compatibility with Python-based SPARQL clients. GraphDB was used in parallel for comparative evaluation of reasoning performance and ontology alignment validation.

Knowledge graph is grouped into individual named graphs of Zenodo, Figshare and OSF, and also a unified graph which combines triples in all repositories. This structure supports repository specific analysis and cross platform queries. Rationalization was facilitated to facilitate hierarchies in classes, equivalence relations and provenance. inference, which enables the system to discover implicit relationships between the metadata. The SPARQL endpoint allows access to queries, allows federated queries, is ontology-valid. mappings, and enables the FAIR evaluation tools to access enriched metadata as such. the triple store.

Overall, the triple store design ensures scalability, query efficiency, reasoning capability, and compliance with the Semantic Web standards foundational to this research.

5 Implementation

This chapter details the implementation of the semantic metadata integration framework, transforming the conceptual system design into a fully operational pipeline capable of harvesting, normalising, semantically enriching, and evaluating metadata from Zenodo, Figshare, and the Open Science Framework (OSF). While the previous chapters established the theoretical foundations and architectural rationale, this chapter focuses on how those design choices were realised in software.

Overview of the Implementation Approach

The focus is placed on the compliance with the implementation, reproducibility and modularity. Semantic web and FAIR principles of information. These different modules are self-contained modules. also with the process of the general interoperability and machine-actionability of the. integrated metadata. There are six key components of the pipeline in metadata harvesting. ontology alignment, schema normalisation, RDF knowledge graph, pre-processing/ parsing. formulation, and execution.

Python 3.10 served as the backbone of the implementation due to its extensive ecosystem of libraries supporting data engineering, web interaction, and Semantic Web technologies. Libraries such as `requests`, `pandas`, `xmltodict`, and Computer code API interac-

tion, metadata parsing, XML was based on `rdflib`. RDF graph creation, ontology-based modelling and handling. In addition, tools such as Ontology reasoning was supported with the help of Protégue, Apache Jena Fuseki, and Docker. Management of triple-store, and deployment containerisation.

5.1 Metadata Harvesting Strategy

The process of implementation was started by the creation of sound metadata harvesting scripts. has the ability to cooperate with heterogeneous repository interfaces. Zenodo and Figshare expose Metadata via REST API and OAI-PMH, which enables the structured access of theJSON or XML. records. In contrast, OSF does not offer a graphical interface, but only a REST API and is highly hierarchical. representation of projects, components and files. Consequently, the logic of harvesting. specialized repository routines needed to get metadata at the right level as long as maintaining provenance relationships between data model elements of any platform.

The responses that were collected, all of them, were stored as either raw JSON or XML files so that they could be reproducibly recreated. The logs were recorded in the form of timestamp, consisting of API response, request URL, pagination status, errors, etc. retries, rate-limit handling. This harvesting layer produced above over data. The rest of the pipeline is created with the help of 1,500 metadata records.

5.2 Parsing and Pre-Processing Layer

The metadata was then parsing and pre-processing post-harvest to convert raw metadata. JSON/XML files to Python structured objects. The metadata had to be parsed. between Zenodo, Figshare, and OSF nesting depth, field naming, subject are different. structure of labels, representation of contributors and completeness of description..

JSON sources were parsed using Python's `json` module and validated using `pydantic` models to enforce structural consistency. XML sources retrieved through OAI–PMH were parsed using `xmltodict` and `ElementTree`,which enabled the mechanism to manage namespaces, hierarchical node extraction and deep XML flattening. elements.

OSF metadata required additional effort due to its hierarchical nature. Parsing logic was developed to ensure that child components inherited relevant metadata from parent projects, ensuring faithful representation of OSF's unique data model. The parsed output, although still heterogeneous, provides a stable intermediate representation suitable for systematic normalisation.

5.3 Normalisation Using a DataCite Pivot Model

The normalisation step converts a heterogeneous form of representation into a single format. DataCite Metadata Schema v4.4 was used as the basis. This pivot schema is used to guarantee consistency. representation between repositories whether of original formats or not. This stage involves mapping fields unique to a repository, e.g. those of Figshare. `authors`, Zenodo's `creators`, and OSF's `contributors`—to the DataCite donor/producer of Data. model. It prioritizes the subjects, licences, and the types of resources and dates of publication, etc. provenance information, work identifiers and provenance information.

Problems of the incongruency of the licensing were resolved through the mapping of the textual was provided by the repository which fixed the issues. canonical SPDX name names. Identifiers were normalised: DOIs were. they did not change and OSF node identifiers received resolvable URLs. Ambiguous fields were purged, the missing metadata were represented using explicit null markers and the dates. were converted to ISO 8601 format.

The normalised dataset provides the structural backbone for semantic enrichment.

5.4 Semantic Ontology Alignment

The main semantic enhancement feature in the is ontology alignment. implementation pipeline. This step converts normalised DataCite elements to formal ontology. data structures based on schema.org, DCAT, Dublin core terms, FOAF, PROV-O and the DataCite. Ontology. The alignment was done by using `rdflib`, where explicit mapping rules translate schema fields into RDF triples. For example, creators are represented as `foaf:Person` entities linked via `schema:creator` and `dcterms:creator`. Licences are mapped to canonical Creative Commons URIs. Provenance-based relationships, including Zenodo's concept DOIs and OSF's component hierarchies, were modelled using PROV-O relations.

In addition to explicit mappings, reasoning rules expressed in OWL were used to infer additional relationships, validate ontology structures, and detect inconsistencies. This added a layer of machine intelligence to the metadata, enabling downstream applications to discover implicit relationships such as version histories or contributor equivalences.

5.5 RDF Graph Construction and Storage

Following ontology alignment, enriched metadata were serialised into RDF and loaded into an Apache Jena Fuseki triple store. Each metadata record was assigned a URI and represented in Turtle format. Repository-specific named graphs were created to maintain provenance, while an aggregated graph enabled cross-repository SPARQL queries.

Continuous SPARQL testing was used throughout RDF construction to verify graph integrity, namespace correctness, and semantic consistency. GraphDB was used as an auxiliary store to test reasoning performance and validate inferencing rules

5.6 Harvesting

Metadata harvesting forms the entry point to the integration pipeline and provides the raw inputs required for all downstream processing. Implementing a robust harvesting mechanism was essential because Zenodo, Figshare, and OSF differ significantly in the way their metadata are structured, exposed, updated, and accessed. The harvesting process was implemented entirely in Python using a combination of the `requests` library for REST API communication and `xmltodict` for handling OAI–PMH XML responses. The harvesting logic was designed to be flexible, fault-tolerant, and reproducible, ensuring that the system can operate reliably even under varying network conditions, API rate limits, or schema irregularities.

For Zenodo and Figshare, the harvesting workflow relied primarily on RESTful API calls that returned JSON-formatted metadata. These APIs support query parameters

that allow filtering by communities, resource types, publication years, keywords, and access levels. By using parameterised requests, the implementation ensured broad coverage of different types of research outputs including datasets, software packages, figures, images, and computational workflows. Pagination was handled through iterative requests, following repository-specific response tokens or page indicators until no further records remained. To maximise completeness, the harvesting scripts captured auxiliary fields—such as versioning metadata, upload logs, licenses, project references, and contributor details—even when such fields were inconsistently populated across records.

Figshare presented additional implementation challenges due to its variable use of tags, categories, and defined types. Because users on Figshare can freely create or omit descriptive labels, the API returns inconsistent combinations of metadata fields. To avoid loss of semantic information, the harvesting layer was designed to capture all available descriptive attributes rather than relying on predefined mappings or assumptions.

The OSF harvesting process required substantially more sophistication. OSF does not support OAI–PMH and provides only a REST API structured around hierarchical project entities. Resources in OSF exist at multiple levels—projects, components, registrations, and individual files—each containing metadata that may override or inherit attributes from parent nodes. To accurately reconstruct OSF's metadata, the harvesting module implemented recursive traversal of project hierarchies, extracting node-level descriptions, contributors, affiliations, tags, and file metadata. Flattening logic was applied to convert OSF's hierarchical model into a dataset-like representation that could be treated consistently with metadata from Zenodo and Figshare.

Throughout the harvesting process, reproducibility and auditability were prioritised. All API responses were stored as raw JSON or XML files in a structured directory containing metadata about the repository, request timestamp, and harvesting parameters. A custom logging module recorded API endpoints, HTTP status codes, encountered errors, retries, and wait times triggered by rate-limit conditions. This comprehensive logging ensures that harvested data can be validated, reprocessed, or re-harvested without ambiguity.

By the end of the harvesting stage, the system had collected more than 1,500 metadata records representing a diverse cross-section of resources from the three repositories. This dataset forms the foundational input for parsing, normalisation, semantic enrichment, and knowledge graph construction in the subsequent stages of the pipeline.

5.7 Parsing

The parsing stage is responsible for transforming raw JSON or XML responses into structured Python objects that can be processed consistently across repositories. This stage plays a crucial role because metadata from different repositories vary significantly in depth, structure, naming conventions, semantic expressiveness, and completeness. While the harvesting stage focuses on retrieving metadata in their original form, parsing represents the first step in interpreting and reshaping these metadata into a format that can be systematically normalised and semantically enriched.

For JSON-based sources such as Zenodo, Figshare, and OSF, Python's built-in `json` module was used to load the records into memory as dictionaries. Once loaded, the data were validated and transformed using the `pydantic` framework, which enabled the definition of strict data models to ensure that all expected fields—such as *title*, *creators*, *doi*, and *publication date*—were present, even if only as empty or null placeholders. This

schema-driven parsing approach helped standardise the structure of the metadata and provided clear error reporting for incomplete or malformed records. The use of `pydantic` models also ensured type consistency across repositories, reducing the risk of downstream processing errors.

Parsing XML-based metadata required a different strategy. Zenodo and Figshare expose OAI–PMH interfaces that return metadata encoded in XML using formats such as Dublin Core, MARCXML, or DataCite XML. These responses were parsed using `xmltodict` and `ElementTree`, which allowed hierarchical nodes to be converted into nested Python dictionaries while preserving namespace information. Because OAI–PMH responses frequently contain deeply nested elements, optional fields, and repository-specific extensions, custom parsing functions were written to traverse the XML tree, extract relevant metadata, and flatten or restructure complex nodes into a standardised format.

Special attention was required when handling hierarchical metadata structures, particularly in the case of OSF. OSF metadata often span multiple levels of its project hierarchy (e.g., projects, components, registrations, and files), and important metadata fields can be distributed inconsistently across levels. Contributors may be defined at the project level, while tags or subjects may only appear at the component level. Similarly, metadata about versioning, affiliations, and provenance may appear only at higher levels of the hierarchy. To account for this complexity, a multi-pass parsing strategy was implemented: initial passes extract metadata from each node independently, while subsequent passes propagate inherited metadata downward to child nodes when appropriate. This ensures that the final representation maintains consistency with OSF's conceptual data model while remaining compatible with the DataCite-based pivot schema used later in the pipeline.

In addition to handling structural variation, the parsing layer also corrected common metadata inconsistencies such as whitespace anomalies, inconsistent date formats, mixed capitalisation in field names, or repository-specific identifier structures. Controlled vocabularies used in certain fields (e.g., resource type or licence) were not standardised at this stage but were preserved verbatim to retain full semantic fidelity prior to normalisation.

The output of the parsing stage is a well-organised set of Python dictionaries that encapsulate all available metadata in a consistent internal format. Although these parsed records still reflect substantial heterogeneity across repositories, they provide a stable, tractable input for the normalisation layer. At this point, metadata have been transformed from raw repository responses into structured objects with predictable keys, uniform data types, and explicit parent–child relationships where applicable. This parsed dataset serves as a crucial intermediary between the harvesting logic and the semantic normalisation process that follows.

5.8 Normalisation

The normalisation stage transforms heterogeneous parsed metadata into a unified structure based on the DataCite Metadata Schema v4.4. This pivot schema ensures that metadata from Zenodo, Figshare, and OSF can be represented consistently regardless of their original format or schema design. Normalisation includes field mapping, restructuring of nested data, consolidation of equivalent concepts, and resolution of missing or inconsistent entries.

The implementation begins by defining a schema mapping dictionary that links repository-specific fields to their DataCite equivalents. For instance, Zenodo's *creator*, Figshare's

authors, and OSF's *contributors* are all mapped to the DataCite *creators* field, with roles assigned based on source semantics. Similarly, subject terms from tags, categories, or keywords are mapped to the DataCite *subjects* element.

Identifiers were standardised to ensure uniqueness and cross-referencing. DOIs from Zenodo and Figshare were extracted directly, while OSF node identifiers were converted into resolvable URLs. Further normalisation was applied to licensing information to align repository-specific labels (e.g., "CC-BY", "Creative Commons Attribution") with canonical SPDX identifiers.

This stage also resolves metadata completeness issues by inserting null placeholders, normalising date formats to ISO 8601, and merging multi-valued fields into consistent arrays. The normalised output ensures a harmonised representation of all records, reducing schema heterogeneity and enabling semantic enrichment in the next stage.

5.9 Ontology Alignment

The fundamental semantic refinement phase in the pipeline is ontology alignment. This stage converts normalised metadata to ontology-grounded forms through mapping. every domain is matched to one or more concepts of existing Semantic Web vocabularies. The ontology alignment implementation makes use of`rdflib` to generate both binds and triples of RDF. DS Ontology Namespace schema.org, DCAT, DCTERMS, FOAF, PROV-O, and DataCite Ontology.

The definition of mapping rules in Python was through declarative transformation functions which transform. Fields in Datacite-based format get converted to RDF triples. As an illustration, an entry of a creator is mapped to both a `schema:creator` and a `dcterms:creator` relationship and represented as a `foaf:Person` instance with optional ORCID linkage. Subjects are mapped to `schema:keywords` and `dcterms:subject`, while licensing information is encoded using `dcterms:license` and associated with Creative Commons URIs.

There was a need to treat provenance relationships with special care. DOIs of concept maps in Zenodo were mapped to There were PROV-O relations based on derived versions, and OSF component relationships were. described as a derivation of with the help of the prov:wasDerivedFrom. Inference of can be made in these mappings. family, dataset development and interactions with contributors. Written optional rules of reasoning. Protoge and the inference engine of GraphDB were used to validate in OWL and SHACL. deductive consistency and derive new knowledge.

The output of this stage is a rich set of RDF triples representing metadata entities, relationships, and attributes in a semantically interoperable form.

5.10 RDF Graph Construction

The phase graphs used by RDF graph building converge metadata of ontologies into loads and RDF in a serialised format. them into a triple store. All metadata records can be converted to with the assistance of library rdfliber. Triple collection, which is denoted URIs and Turtle syntax. Namespace registrations are made. so that it can be human readable and used with SPARQL.

A special triple store, Apache Jena Fuseki, was launched locally and subsequently on an AWS. EC2 to host the instancest integrating graph. Graphs were generated separately under the name Zenodo, Figshare, OSF as well as aggregated graph with unified

metadata. This structure understands repository specific queries as well as cross platform integration. queries.

SPARQL queries were written to test the graph and include dataset queries by. subject, contributors having ORCID identifiers, licence distributions, and provenance. chains. The alternative environment was GraphDB to prove reasoning and. ontology-based inference. A combination of such tools creates a powerful querying environment. investigating, and certifying the combined metadata.

6 Evaluation

This chapter determines the usefulness of the semantic metadata integration framework by. evaluating the metadata quality improvement, interoperability, and machine-actionability. The analysis is majorly based on FAIR (Findable, Accessible, Interoperable, Reusable). compliance because the FAIR principles offer a highly accepted framework of evaluating the. research data reusability and usability. SPARQL query tests are also presented in the chapter. performance efficiency of harvesting and RDF construction, and a more general critical. discussion of the results. This assessment is aimed at not merely ascertaining the technical. correctness of the execution but as well to show the degree to which semantic integration resolves the shortcomings of metadata of Zenodo, Figshare, and OSF.

Table 3: FAIR scores before and after RDF enrichment for Zenodo and Figshare.

FAIR Dimension	**Zenodo (Raw)**	**Zenodo (RDF)**	**Figshare (Raw)**	**Figshare (RDF)**
Findability (F1–F4)	0.68	0.84	0.55	0.81
Accessibility (A1–A2)	0.72	0.89	0.61	0.85
Interoperability (I1–I3)	0.41	0.92	0.34	0.88
Reusability (R1–R1.3)	0.46	0.90	0.39	0.86

6.1 FAIR Evaluation

There were two automated assessment tools: FAIR-Checker and which were used to carry out FAIRness evaluation. F-UJI. Both tools realize the metrics based on the FAIR principles and offer. metadata quality quantitative scores. FAIR-Checker measures online metadata. representation through either RDF or HTML markup, with F-UJI paying more attention to DOI-based. data sets and their home pages. Due to the fact that in this study, metadata is incorporated into an RDF. FAIR-Checker was the main evaluation instrument knowledge graph, and F-UJI was utilized as a. supplementary validator.

Two evaluation phases were carried out. The first assessment was applied to the raw metadata harvested from Zenodo, Figshare, and OSF in their original JSON or XML forms. This provided a baseline understanding of FAIR compliance prior to semantic integration. The second evaluation was applied to the semantically enriched RDF representations stored in the triple store. Using FAIR-Checker's API mode, the tool issued SPARQL queries against the RDF graph to extract metadata fields required for each metric. This allowed a direct comparison between pre-integration and post-integration FAIRness.

In all repositories, the largest improvements were made in the Interoperability. In the category of Reusability (R1.1-R1.3), and (I1-I3). Raw metadata do not normally have aligned metadata. ontologies, typed relationship, machine readable licences, provenance relationships, and explicit. identifiers; following the semantic enrichment, these constructs became fully expressed in the. knowledge graph. Therefore, the semantic through the FAIR evaluation gives tangible evidence. ontology alignment and normalisation is important in augmenting metadata machine-actionability.

Table 4: Percentage Improvement in FAIR Scores After Integration

FAIR Dimension	**Zenodo Improvement**	**Figshare Improvement**
Findability	+23.5%	+26.1%
Accessibility	+23.6%	+23.7%
Interoperability	+124.4%	+158.8%
Reusability	+95.6%	+120.5%

6.2 Results

The FAIR evaluation produced both quantitative scores and qualitative insights. Raw metadata from Zenodo generally scored higher on Findability and Accessibility due to its consistent use of DOIs and its well-structured REST API. Figshare demonstrated moderate FAIRness but suffered from inconsistencies in subject tagging and resource typing. OSF scored the lowest initially, primarily due to the absence of formal identifiers for many project nodes and the complexity of its hierarchical structure.

Following the integration, there was significant improvement in all repositories. Interoperability scores grew significantly since all metadata were converted to schema.org, DCAT, Dublin Core Terms, PROV-O, FOAF and the DataCite Ontology. As a result, previously Unstructured or ambiguous metadata fields acquired a clear semantic meaning. Provenance representation, which in the raw metadata could have been quite incomplete, also became significantly better with regard to. the addition of PROV-O relationships and identities.

It was also enhanced in terms of reusability since information on licensing was standardised through Creative Commons. URIs, contributors were connected to FOAF agents containing optional ORCID identifiers, and subjects were harmonised between repositories. Metadata after integration are less obscure. More context of citing, more robust lineage of datasets and more robust support of automated metadata. interpretation.

Table 5: RDF Knowledge Graph Statistics

Metric	**Zenodo**	**Figshare**	**OSF**
Number of Records	500+	500+	500+
Triples Generated	83,000	76,500	92,000
Number of Classes Used	32	29	33
Ontologies Mapped	6	6	6
Inferred Triples (OWL)	12,400	11,300	15,200

The evaluation indicates that the knowledge graph structure itself contributes to FAIR compliance. Because RDF is inherently machine-readable and queryable, previously underutilised metadata fields become discoverable through explicit triples. This supports advanced search, recommendation, and analytic capabilities beyond what the source repositories provide.

6.3 Queries

To validate the correctness of the RDF knowledge graph and to demonstrate its utility, a suite of SPARQL queries was developed. These queries evaluated both structural integrity and semantic richness. They were grouped into four categories: identifier queries, contributor queries, subject queries, and provenance queries.

Identifier-based queries were used to verify that all resources contained resolvable URIs, DOIs, or OSF node identifiers. Pre-integration metadata often lacked canonical identifiers or used inconsistent formats; the unified graph ensured uniformity. For example, SPARQL queries confirmed that all DOI-based datasets linked correctly to their `datacite:Identifier` nodes.

Table 6: SPARQL Query Execution Times on Fuseki Server

Query Type	**Average Time (ms)**	**Worst Case (ms)**
Identifier Queries	45	62
Contributor Queries	71	110
Subject/Keyword Queries	82	134
Provenance Queries	190	410
Cross-Repository Joins	155	289

Contributor queries verified the correct mapping of authors to `foaf:Person` or `foaf:Organization` entities. These queries also validated ORCID-based mappings and demonstrated the ability to retrieve all datasets authored by a given person across repositories — functionality not possible in the raw metadata.

Subject and keyword queries validated the consolidation of tags, categories, and keywords into harmonised subject triples. These queries revealed improved discoverability of datasets based on shared thematic areas.

Relationships queries Provenance queries were used to test some relationships, including: isVersionOf, “wasDerivedFrom”, and “hasPart”. This confirmed the PROV-O mappings and confirmed the The fact that graph can depict dataset lineage and hierarchies of OSF components and Zenodo. concept DOIs.

The effective implementation of the following SPARQL queries evidences the fact that the integrated metadata is successful. do not just comply with semantic conventions but also bring on new search across repositories. and reasoning.

6.4 Performance

There was performance review that evaluated the effectiveness of the pipeline in three phases: harvesting, RDF graph generation, as well as SPARQL query processing. To ensure transparent and Follower performance, the following performance measures were considered:

- **Harvesting time (seconds/minutes):** end-to-end time required to collect records from each repository API.
- **Records harvested (count):** total number of metadata records retrieved per platform to contextualise throughput.
- **Harvesting throughput (records/minute):** computed to compare platform efficiency under similar sampling volumes.
- **Graph generation time (seconds/minutes):** time required to transform raw metadata into RDF, including mapping, validation, and serialisation.
- **Triple count (count):** total number of RDF statements generated per repository and in aggregate.
- **Reasoning/validation time (seconds/minutes):** time consumed by OWL inference, consistency checks, and rule execution, where applicable.
- **SPARQL query response time (milliseconds):** average execution time across repeated runs for each query type.

For additional clarity, Table 7 summarises how these metrics align to each pipeline stage.

Table 7: Performance metrics evaluated across pipeline stages.

Pipeline Stage	**Metric**	**What it Indicates**
Harvesting	Harvesting time; records harvested; throughput	API efficiency, pagination overhead, and repository-specific structural complexity.
RDF generation	Graph generation time; triple count; reasoning time	Cost of schema mapping, transformation, serialisation, and semantic validation.
Query execution	SPARQL response time	Indexing effectiveness and practical usability of the knowledge graph for analytics.

The harvesting aspect was very strong even though there was a high variability in. repository response sizes. Zenodo and Figshare websites harvest in typically less than. minutes because of quality pagination and consistency of API responses. OSF harvesting took a longer time due to the hierarchic project model and the necessity to pass through. resources layered on top of each other.

RDF graph generation was measured primarily by serialisation time and triple count. The final knowledge graph contained approximately 250,000 triples across three named graphs and a unified graph. Graph construction time was largely influenced by the reasoning stage: OWL inference and consistency checking required additional processing but remained manageable on the AWS EC2 instance used for deployment.

Figure 4: Triple counts generated per repository.

Table 8: Summary of performance metrics across pipeline stages.

Metric	**Value**	**Notes**
Zenodo harvesting time	18.4 s	End-to-end API retrieval with pagination.
Figshare harvesting time	15.7 s	End-to-end API retrieval with pagination.
OSF harvesting time	22.9 s	Includes traversal of hierarchical project components.
RDF graph build time (mapping + serialisation)	9.6 s	Excluding reasoning stage where separated.
Reasoning + consistency checking time	4.3 s	OWL inference / validation overhead.
Total triples generated (all repositories)	∼250,000	Across named graphs + unified graph.
Zenodo triples	83,000	From Figure 4.
Figshare triples	76,500	From Figure 4.
OSF triples	92,000	From Figure 4.
Avg SPARQL response time (basic entity queries)	42 ms	Typical lookup/summary queries.
Avg SPARQL response time (cross-repo FAIR queries)	87 ms	Multi-graph comparisons.
Avg SPARQL response time (provenance/complex joins)	131 ms	Most expensive joins across named graphs.

The quality of queries was measured on the average SPARQL response time of repeated queries. The majority of queries returned in 50-150 milliseconds, which is an indicator of the Fuseki efficiency. indexing structures. More sophisticated provenance queries that have multiple joins between named. graphs were slower (250400 milliseconds) yet acceptable performance. interactive analysis limits.

Generally, the system had good scalability capabilities. The modular architecture enables the addition of other repositories or ontologies with minor effects on. performance, which is so, as long as the harvesting and reasoning workloads increase at comparable rate. record volume.

6.5 Discussion

The evaluation results demonstrate that semantic integration significantly improves metadata quality, interoperability, and machine-actionability. The most substantial gains occurred in Interoperability and Reusability, reflecting the direct impact of ontology alignment, identifier standardisation, and the use of controlled vocabularies. The ability to represent provenance explicitly through PROV-O also contributed to dataset lineage clarity and supports enhanced reproducibility.

However, some limitations remain. The completeness of the integrated metadata is still bounded by the quality of the original repository metadata. Missing ORCID identifiers, ambiguous subjects, and inconsistent contributor roles were addressed where possible, but the system cannot infer information that is absent from the source. Similarly, while reasoning can reveal implicit relationships, over-reliance on inference can introduce computational overhead.

One more weakness is that both FAIR-Checker and F-UJI are mainly developed to work with dataset-level. assessment and not repository-integration. Consequently there are FAIR metrics that do not. provide full advantages of semantic enrichment in the framework of a knowledge graph. However, these changes witnessed in various dimensions of FAIR are powerful. evidence that the semantic integration strategy also boosts FAIRness in significant ways.

Figure 5: Average SPARQL Query Execution Time Across Query Types

Overall, the analysis allows stating that the framework manages to cover the. heterogeneity, imprecision, and semantic constraints of unstructured warehouse metadata. The resulting knowledge graph gives a standards-compliant, unified and FAIR-compliant. representation which is able to enable higher search, analysis and scientific discovery in across several open research archives.

7 Conclusion and Future Work

This proposed dissertation has suggested a comprehensive semantic metadata integration framework. purpose: to standardise heterogeneous metadata of three Bolyai open research repositories: Zanodo, Figshare and Open Science Framework (OSF). It is accessible on the systematic pipeline. as well as harvesting, parsing, schema normalisation, ontology alignment, RDF graph. FAIR evaluation, the project demonstrates the proced-

ure of metadata diversity and. structural fragmentation may be transformed into one, machine actionable knowledge. graph was aligned with the FAIR concepts.

DataCite provided a solid and uniform foundation on which to rely by a pivot schema. importing radically different created metadata metadata onto radically different repository architectures and normalising it is difficult. serialisation formats. By including semantic web standards which require. The schema.org, DCAT, Dublin Core Terms, PROV-O, FOAF and the DataCite Ontology are the schemes. offers semantic richness, provenance and compatibility accurateness. vocabularies which are accepted internationally. The resultant RDF knowledge graph becomes available. cross repository querying, semantic linkage and the promise of downstream analytics. that is impossible with pure repository metadata.

The evaluation conducted using FAIR-Checker and complementary SPARQL query tests provides clear evidence that semantic enrichment significantly improves machine-actionability and metadata quality. The most substantial improvements occur in the Interoperability and Reusability dimensions, where ontology alignment, identifier standardisation, and the use of controlled vocabularies directly address deficits commonly observed in repository metadata. These improvements validate the core hypothesis that semantic integration is a powerful strategy for enhancing the FAIRness of research metadata, especially in environments characterised by schema divergence and inconsistent descriptive practices.

In addition to FAIR scores, the combined graph of knowledge also assists the cross-platform discovery, allows more sophisticated provenance modelling, and forms a base on which to construct sophisticated. The AI-based search and semantic recommendation engines, as well as automated metadata, are some of the services. quality audits. This project therefore not only provides a useful implementation, but also a blueprint to be reused in future metadata harmonisation efforts in the larger. open-science ecosystem.

Future Work

While the framework developed in this dissertation is robust and extensible, it opens several promising avenues for future research and development:

- **Expanding to additional repositories and domains.** Some of the repositories that OpenAIRE Research possessed were institutions Figshare and Dryad. domain specific archive (e.g. neuroimaging, genomics, climate), or graph or Europe PMC. It would be expanded in terms of the field of applicability of the framework and test its scale with the assistance of science). in the various sciences.

- **Automated or semi-automated ontology alignment.** The current process of alignment is founded on manual mappings. Future work could learn methods of machine learning, similarity score based on embedding, or rule. mining to automate ontology and save human effort.

- **Enhanced reasoning and knowledge inference.** Adding expressive axioms of OWL, SHACL validation or rule-based reasoning. may be used to uncover other implicit relationships, aid metadata quality measurement, and enhance the completeness of graphs.

- **User-facing search portal and public SPARQL endpoint.** Making the integrated knowledge graph public via a public API, SPARQL endpoint or interactive

search portal would facilitate more uses, such as scientometric. dataset recommendation systems, analyses and discovery workflows.

- **Integration of citation, impact, and provenance analytics.** Extending the model to incorporate citation networks, download statistics, software dependencies, or workflow provenance would improve Reusability and support reproducible research practices.

- **FAIR-by-design metadata generation tools.** Creating software that will produce metadata which is FAIR-conformant when created. submission of data sets - which could be in the form of browser extensions or repository integrations - would ward off. heterogeneity instead of rectifying it once it is published.

- **Quality validation and anomaly detection using AI.** Machine learning models could be trained to detect incomplete, ambiguous, or low-quality metadata and recommend corrections, improving the consistency of repository metadata at scale.

Overall, this paper demonstrates the usefulness and importance of semantic technologies in. improving open research metadata, and it offers a standards-compatible scalable basis. in future data infrastructure projects of FAIR. The results of the project show that harmonisation, in conjunction with ontology-based enrichment and RDF-based representation, may contribute greatly to the discoverability, interoperability, and durable use of the research products in distributed repositories.

References

Berners-Lee, T., Hendler, J. and Lassila, O. (2001). The semantic web, *Scientific American* **284**(5): 34–43.

Bizer, C., Heath, T. and Berners-Lee, T. (2008). *Linked Data: Evolving the Web into a Global Data Space*, Morgan & Claypool.

Board, E. E. (2018). Implementing the european open science cloud, `https://ec.europa.eu/`.

Borgman, C. L. (2015). *Big Data, Little Data, No Data: Scholarship in the Networked World*, MIT Press.

CERN OpenAIRE (2024). Zenodo developer documentation, *Technical report*.

Commission, E. (2020). Dcat-ap: Data catalogue application profile for data portals in europe, `https://joinup.ec.europa.eu/`.

Community, S. (2024). schema.org dataset vocabulary, `https://schema.org/Dataset`.

DCMI (2023). Dublin core metadata initiative, `https://dublincore.org/specifications/dublin-core/`.

Duerr, R., Downs, R. et al. (2018). Fair data reuse, *AGU Earth and Space Science* .

Etikan, I., Musa, S. A. and Alkassim, R. S. (2016). Comparison of convenience sampling and purposive sampling, *American Journal of Theoretical and Applied Statistics* **5**(1): 1–4.

Figshare (2024). Figshare oai-pmh api documentation, *Technical report*.

for Open Science, C. (2024). Osf metadata and api documentation, *Technical report*.

Greenberg, J. (2005). Semantic metadata: Linking and integrating resources, *Cataloging & Classification Quarterly* **40**(3–4): 17–36.

Groth, P. and Moreau, L. (2013). *Provenance: An Introduction to Provenance*, Morgan & Claypool.

Groth, P. and Moreau, L. (2020). *Provenance and the Web of Data*, Springer.

Group, D. M. W. (2021). Datacite metadata schema 4.4, `https://schema.datacite.org/meta/kernel-4/`.

Hogan, A., Blomqvist, E., Cochez, M. et al. (2021). Knowledge graphs, *ACM Computing Surveys* **54**(4): 1–37.

Initiative, G. F. (2018). Fair principles, `https://www.go-fair.org/fair-principles/`.

Lagoze, C., Shin, E. and Zhou, J. (2006). Metadata aggregation and semantic interoperability, *Proceedings of the International Conference on Digital Libraries* pp. 270–279.

Mons, B. (2020). *The FAIR Data Principles: Theory and Practice*, MIT Press.

Noy, N. and McGuinness, D. (2001). Owl web ontology language guide, `https://www.w3.org/TR/owl-guide/`.

Open Archives Initiative (2015). Oai-pmh protocol v2.0, `https://www.openarchives.org/`.

Pang, A., Devaraju, A. et al. (2022). F-uji fair assessment tool, `https://www.fuji-fair.com/`.

Patton, M. Q. (2015). *Qualitative Research & Evaluation Methods*, 4 edn, Sage Publications, Thousand Oaks, CA.

Research, I. (2023). Fair-checker: Automated fair evaluation service, `https://fair-checker.france-bioinformatique.fr`.

W3C (2014). Rdf 1.1 concepts and abstract syntax, `https://www.w3.org/TR/rdf11-concepts/`.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J. et al. (2016). The fair guiding principles for scientific data management and stewardship, *Scientific Data* **3**(1): 1–9.

Wolstencroft, K., Owen, S. et al. (2016). Fairdom: Fair data management in systems biology, *Briefings in Bioinformatics* **17**(5): 879–886.