

## D7.2. SLICES Interoperability Framework and Integration with EOSC

Acronym	SLICES-PP
Project Title	Scientific Large-scale Infrastructure for Computing/Communication Experimental Studies – Preparatory Phase
Grand Agreement	101079774
Project Duration	40 Months (01/09/2022 – 31/12/2025)
Due Date	31 August 2024 (M24)
Submission Date	19 August 2024 (M24)
Authors	Yuri Demchenko (UvA), Shashank Shrestha (UvA), Panayiotis Andreou (UCLan)
Reviewers	Brecht Vermeulen (IMEC), Anna Brekine (IoTLab)



*This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101079774. The information, documentation and figures available in this deliverable, is written by the SLICES-PP project consortium and does not necessarily reflect the views of the European Commission. The European Commission is not responsible for any use that may be made of the information contained herein.*



## Executive Summary

---

Ensuring the reproducibility of experimental research is crucial for scientific integrity and reliability. This report outlines the data management principles necessary for achieving interoperability and integration of the SLICES Research Infrastructure (SLICES-RI) with the European Open Science Cloud (EOSC). The developments, design ideas, and ongoing research efforts summarized here adhere to the FAIR data principles and emphasize the importance of data management in archiving and sharing.

SLICES aims to align with the EOSC interoperability framework, facilitating seamless integration. This alignment ensures that data and services from SLICES can effectively interact with the broader EOSC environment, promoting data sharing and reuse across borders and disciplines. Efforts are being made to integrate SLICES with EOSC, leveraging the EOSC interoperability framework to enable automated and reproducible experimental research. This integration will enhance the ability of researchers to share and reuse experimental data efficiently.

The proposed Data Management Infrastructure (DMI) architecture is designed to support Experimental Research Reproducibility as a Service (ERRaaS). This architecture includes processes for data collection, archiving, and sharing to ensure that experiments can be replicated accurately by other researchers. ERRaaS involves several key processes: gathering data from various experimental sources, storing data in a structured and secure manner, and ensuring that data can be easily accessed and shared among researchers.

Ongoing work focuses on developing Metadata Registry Services (MRS) to enhance experimental data sharing. This involves creating standardized metadata models to facilitate interoperability and integration with EOSC. By implementing MRS, SLICES can ensure that metadata is consistently documented and accessible, which is essential for the reproducibility and reusability of experimental data.

To support these goals, RO-Crate is being utilized for data archiving and sharing within SLICES. RO-Crate is a method for packaging research data with rich metadata, ensuring that data is well-documented and easily accessible. Additionally, Data Version Control (DVC) is being implemented to track data lineage and support version control, enhancing the reproducibility of experiments. DVC ensures that researchers can trace the history of data and understand the changes made over time.

By adhering to the FAIR data principles and leveraging advanced data management tools and frameworks, SLICES-RI aims to enhance the reproducibility and integrity of experimental research. Integration with EOSC will further facilitate data sharing and reuse, contributing to the broader scientific community's efforts to achieve reliable and reproducible research outcomes.



## Table of content

---

<b>EXECUTIVE SUMMARY .....</b>	<b>2</b>
<b>TABLE OF CONTENT .....</b>	<b>3</b>
<b>TABLE OF FIGURES .....</b>	<b>5</b>
<b>TABLE OF TABLES .....</b>	<b>5</b>
<b>ACRONYMS.....</b>	<b>6</b>
<b>1. INTRODUCTION .....</b>	<b>7</b>
<b>2. SLICES INTEROPERABILITY FRAMEWORK .....</b>	<b>8</b>
2.1. EOSC DEVELOPMENT AND IMPORTANCE FOR EUROPEAN RESEARCH COMMUNITY.....	8
2.2. EOSC INTEROPERABILITY FRAMEWORK AND EOSC SERVICES .....	9
2.2.1. EOSC Interoperability Layers .....	10
2.2.2. EOSC interoperability recommendations .....	12
2.2.3. EOSC Interoperability and Composability Model .....	12
2.2.4. Conclusion of EOSC IF.....	14
2.2.5. Recently Developed EOSC Data and Metadata Management Services and Tools .....	15
2.3. FAIR DATA PRINCIPLES ADOPTION BY ESFRI .....	17
2.4. SLICES INTEROPERABILITY FRAMEWORK DEFINITION AND RECOMMENDATIONS .....	18
2.4.1. Requirements of SLICES-IF.....	18
2.4.2. SLICES-Interoperability Framework.....	20
<b>3. SLICES DATA MANAGEMENT INFRASTRUCTURE FOR REPRODUCIBLE EXPERIMENTAL RESEARCH .....</b>	<b>24</b>
3.1. EXPERIMENT AUTOMATION AND EXPERIMENTAL RESEARCH REPRODUCIBILITY IN SLICES .....	24
3.1.1. Experiment Reproducibility as a Service in SLICES.....	24
3.1.2. Experimental Data Management Stages .....	24
3.2. SLICES DMI ARCHITECTURE AND REQUIREMENTS TO SUPPORT EXPERIMENTAL DATA MANAGEMENT.....	25
3.2.1. SLICES DMI Architecture.....	25
3.2.2. Requirements to Support the Experimental Data Management.....	27
3.3. METADATA TO DESCRIBE INFRASTRUCTURE AND EXPERIMENT .....	27
3.3.1. General Metadata Definition and Services .....	27
3.3.2. Experiment Data Model and Required Metadata .....	28
3.3.3. SLICES Metadata Definition and Requirements .....	29
3.3.3.1. POS Experiment description and metadata .....	29
3.3.3.2. SLICES Blueprint Architecture for 5G/6G and related networking technologies.....	31
3.3.4. Experiment workflow management with POS and metadata extraction .....	31
<b>4. SLICES METADATA SERVICES FOR (EXPERIMENTAL) RESEARCH DATA SHARING .....</b>	<b>34</b>
4.1. SLICES METADATA DEFINITION AND REGISTRY .....	34
4.1.1. Metadata Design Objectives .....	34
4.1.2. SLICES FAIR Digital Object (SFDO) .....	35
4.1.3. Design Considerations.....	37
4.2. METADATA REGISTRY SERVICE (MRS) .....	38
4.2.1. Architecture.....	38
4.2.2. Repository .....	38
4.2.3. Backend.....	39
4.2.4. Web Portal .....	40
4.2.5. Metadata Crosswalks.....	41



- 5. SLICES SERVICES FOR INTEROPERABILITY AND INTEGRATION WITH EOSC ..... 42**
  - 5.1. USING RO-CRATE FOR RESEARCH DATA ARCHIVING AND SHARING..... 42
  - 5.2. USING DATA VERSION CONTROL FOR EXPERIMENTAL DATA MANAGEMENT ..... 42
- 6. CONCLUSION ..... 50**
- 7. REFERENCES ..... 51**
- APPENDIX A. FAIR DATA PRINCIPLES – THIS TEXT IS TAKEN FROM THE SLICES DMP ..... 53**
  - A.1. MAKING DATA FINDABLE..... 53
  - A.2. MAKING DATA ACCESSIBLE..... 54
  - A.3. MAKING DATA INTEROPERABLE ..... 54
  - A.4. MAKING DATA REUSABLE..... 55





## Table of Figures

---

- Figure 1. EOSC Interoperability and composability models [6]. ..... **Erreur ! Signet non défini.**  
Figure 2. SLICES Interoperability framework and its interaction with EOSC-IF. **Erreur ! Signet non défini.**  
Figure 3. SLICES Experiment lifecycle and data management stages and supporting infrastructure. .... **Erreur ! Signet non défini.**  
Figure 4. Architecture of the SLICES Data Management Infrastructure. **Erreur ! Signet non défini.**  
Figure 5. Generic experiment data model for reproducibility. .... **Erreur ! Signet non défini.**  
Figure 6. SLICES FAIR Digital Object (SFDO). .... **Erreur ! Signet non défini.**  
Figure 7. Metadata Registry System (MRS) Architecture. .... **Erreur ! Signet non défini.**  
Figure 8. Experimental data management architecture. .... **Erreur ! Signet non défini.**

## Table of Tables

---

- Table 1. SLICES Interoperability Framework Recommendations .....22



## Acronyms

---

- AAI** – Authentication and Authorization Infrastructure
- DMI** – Data Management Infrastructure
- DMP** – Data Management Plan
- DPO** – Data Protection Officer
- DQM** – Data Quality Management
- DVC** – Data Version Control
- EOSC** – European Open Science Cloud
- EOSC-IF** - European Open Science Cloud Interoperability Framework
- ERRaaS** – Experimental Reproducibility as a Service
- FAIR** – Findable, Accessible, Interoperable, Reusable
- GDPR** – EU General Data Protection Regulation
- MRS** – Metadata Registry System
- PID** – Persistent Identifier
- POS** – Plain Orchestration Service
- RI** – Research Infrastructure
- RO** – Research Object
- RO-Crate** - Research Object packaging format
- SLICES** – Scientific Large-scale Infrastructure for Computing/Communication Experimental Studies
- SLICES-DS** – SLICES Design Study
- SLICES-IF** – SLICES Interoperability Framework
- SLICES-PP** – SLICES Preparatory Phase
- SLICES-RI** – SLICES Research Infrastructure
- SLICES-SC** – SLICES Starting Community



## 1. Introduction

---

The objective of the SLICES Preparatory Phase (SLICES-PP) project is to build upon the experience of the SLICES Design Study (SLICES-DS) and SLICES Starting Community (SLICES-SC) projects, to finalize the technical design of the new leading-edge research infrastructure, and tackle all key questions concerning legal, financial, and technical issues leading to the establishment of the new SLICES Research Infrastructure (SLICES-RI).

Data management plays a critical role in ensuring the reproducibility of experimental research. Reproducibility is the ability of an experiment or study to be replicated by other researchers and yield the same results, which is a cornerstone of scientific integrity and reliability. In this report, the aspects of data management principles are discussed to achieve interoperability and integration of SLICES with EOSC.

This deliverable 7.2 summarizes all the developments, design ideas and ongoing research on the SLICES-RI interoperability and integration with EOSC to enable experiment research automation and reproducibility. The developments, designs and research adhere to the FAIR data principles and also highlight the recent implementations for data management including archiving, and sharing.

WP7 includes four tasks focusing on (1) interoperability framework and architecture, (2) SLICES DMI for experiment reproducibility, (3) Metadata sharing through Metadata Registry Services (MRS), and (4) interoperability with EOSC and integration with SLICES.

Section 2 provides an overview of SLICES interoperability with EOSC. It also gives details on the EOSC interoperability framework and how it can be integrated with SLICES.

Section 3 provides the details of the proposed DMI architecture and an overview on the processes involved in achieving experimental reproducibility. Experimental Research Reproducibility as a Service (ERRaaS) is also discussed. This section also gives an overview of POS 5G data and the current work on data modelling for metadata extraction, sharing, and experiment reproducibility.

Section 4 highlights the ongoing work on Metadata Registry Service (MRS) for experimental data sharing whereas Section 5 discusses and provides details on the ongoing work for data management. The use of RO-Crate for data archiving and sharing as well as DVC for version control, data lineage tracking, and experiment reproducibility are discussed.



## 2. SLICES Interoperability Framework

---

SLICES-RI Interoperability analysis and recommendations were one of the important topics studied in SLICES-DS project (2020-2022). It is reported in three deliverables of its WP4 where the general approach to defining the SLICES Interoperability Framework (SLICES-IF) was outlined and main interoperability requirements were proposed [1] [2] [3]. The proposed SLICES-IF is based on the EOSC Interoperability Framework defined at that time by the EOSC community and ongoing EOSC related projects.

This section provides a summary of the work done in SLICES-DS project and updates it with the recent developments in the EOSC community and related recent projects such as EOSC Future<sup>1</sup> and current wider adoption of the EOSC recommendations by the European research community<sup>2</sup>.

### 2.1. EOSC development and importance for European Research Community

EOSC is defined as European federated data infrastructure to support cross-domain (scientific and national) data sharing and exchange while being capable of new trends in data spaces management such as data sovereignty and trusted data processing platforms. EOSC EU Node<sup>3</sup> provides general information about current and future EOSC transformation.

EOSC EU Node is a platform that primarily supports multi-disciplinary and multi-national research promoting the use of FAIR (Findable, Accessible, Interoperable, Reusable) data and supplementary services in Europe and beyond. Within this environment, researchers can find easy-to-use tools and the much-needed support to both individually and collectively, plan, execute, disseminate, and assess their typical research workflows and outcomes across the EOSC ecosystem.

EOSC EU Node can be recognised as the first European-level node of the EOSC Federation promoted by the European Commission as an operationalised platform in production. Managed services provided by the EOSC EU Node have been procured by the European Commission, hosted and operated by third-party contractors, and made available to EOSC Stakeholders primarily gathered under the EOSC Association. The oversight of EOSC EU Node platform services is provided by the EOSC Tripartite Governance.

The European Open Science Cloud (EOSC) ultimately aims to develop a “Web of FAIR Data and services” for science in Europe upon which a wide range of value-added services can be built. These range from visualisation and analytics to long-term information preservation or the

---

<sup>1</sup> EOSC Future: <https://eoscfuture.eu/> (Accessed: 14 August 2024)

<sup>2</sup> It is important to mention that at the time of writing this deliverable (August 2024), the EOSC is going through an organisational transformation that includes some services provided by the EOSC Association. But presumably this will not affect technical frameworks and solutions proposed before 2024, in particular EOSC Interoperability Framework.

<sup>3</sup> <https://open-science-cloud.ec.europa.eu/about/eosc-eu-node>





monitoring of the uptake of Open Science practices. Future EOSC federated data infrastructure is expected to include a network of different domain specific EOSC Nodes that should comply with the specific requirements needed to support FAIR compliant data sharing<sup>4</sup>.

The EOSC enables further development for scientific communities and research infrastructures towards seamless access, FAIR data principles adoption and management, reliable reuse of research data and all other digital objects produced along the research life cycle (e.g. methods, software and publications).

The current version of the EOSC EU Node includes also Resource Hub<sup>5</sup> that includes the following resources: Publications, Data, Software, Services, Data Sources, Training, and Other Products, all of which provide links to the original locations of resources (services, repositories, and links).

## 2.2. EOSC Interoperability Framework and EOSC Services

Interoperability is an essential feature of EOSC ecosystem because a federation of services and data exchange is impossible without interoperability among different EOSC components. The meaningful exchange and consumption of digital objects is necessary to generate value from EOSC which can only be realised if different components of the EOSC ecosystem (software/machines and humans) have a common understanding of how to interpret and exchange them, what are the legal or regulation restrictions, and what processes are involved in distribution, consumption and production of them. To facilitate this, EOSC interoperability framework (EOSC-IF) is defined as a generic framework for all the entities involved in the development and deployment of EOSC. Core EOSC-IF defines the common requirements and challenges faced by the user communities targeted by EOSC, which can be used to develop potential technical solutions to meet these requirements to achieve interoperability at different levels and between different domains.

EOSC-IF is derived from the European Interoperability Framework [4] which defines the interoperability of an information technology system by four key elements, i.e., technical, semantic, organization, and legal interoperability, what makes EOSC-IF foundationally compliant with the European technical policy on digital infrastructures. However, EOSC-IF is providing a basis for the research community on smooth data sharing and FAIR data principles compliance. The first full definition of the EOSC-IF was published in February 2021 as an outcome of the EOSC-Hub project [5]. EOSC-IF has been further improved in the EOSC-A Task Force on “Technical Interoperability of Data and Services” deliverable [6] and in the technical paper produced by the EOSC Executive Board Working Groups and Architecture [7]. As

---

<sup>4</sup> Building the EOSC Federation: requirements for EOSC Nodes [draft v 24/5] - [https://eosc.eu/wp-content/uploads/2024/05/EOSC-A\\_GA8\\_20240527-28\\_Paper-G\\_Update\\_EOSC\\_Nodes\\_requirements-DRAFT-v240524.pdf](https://eosc.eu/wp-content/uploads/2024/05/EOSC-A_GA8_20240527-28_Paper-G_Update_EOSC_Nodes_requirements-DRAFT-v240524.pdf)

<sup>5</sup> <https://open-science-cloud.ec.europa.eu/resources/all>



practical recommendations, the newly defined EOSC-IF documents also include templates for EOSC-IF compliance declaration [provider, resource] that should be provided for future EOSC compatible services registration.

### **2.2.1. EOSC Interoperability Layers**

EOSC-IF defines four layers of interoperability supported by dedicated components, namely: technical, semantic, organizational and legal.

**Technical interoperability** is defined as the ability of different information technology systems and software applications to communicate with each other and seamlessly exchange data. The main challenges in the way of ensuring technical interoperability are:

- (i) Research data may be stored in different formats which are either general purpose (CSV, XML, JSON, etc.) or community specific (Darwin core, FITS, VOTable, VOResource, etc.) which are difficult to reuse across communities. To solve this, a common-minimum metadata model is needed to allow seamless discovery and reuse of data across multiple formats;
- (ii) Research data is often not available in multiple granularities. This makes it difficult to be found and reused by different scientific domains requiring different granularity of data. Thus, there is a requirement for research data to be stored at multiple levels of abstractions (fine grain and coarse grain) so that a wide variety of scientific and application domains can benefit from it;
- (iii) Generally, scientific communities employ community-specific persistent identifiers (PURL, IUPAC international chemical identifier, DOI, etc.) with a different set of policies. This sometimes results in identifiers which can be difficult to resolve. Therefore, a community-agnostic PID policy is required for a common understanding;
- (iv) Separate authentication and authorization are often required when accessing services across different infrastructures and communities, which generally requires transfer of personal information among identity and service providers. To address this issue, there is a need to develop an Authentication and Authorization Infrastructure (AAI) framework that is community independent and minimally obstructive.

**Semantic interoperability** refers to the ability that the exchanged data is understood well and have a common meaning across different entities of the EOSC ecosystem. The major obstacles to ensure semantic interoperability are:

- (i) Semantic artefacts are poorly documented and definitions of terms used are not precisely defined. This makes it difficult to be used across communities;
- (ii) Furthermore, common reference repositories/registries for semantic artifacts are not easily available or maintained for long enough;



- (iii) Also, there is a lack of common metadata schemas across communities. Different communities use different metadata schemas such as DarwinCore, RDA metadata, DCAT, DDI4, etc.

All these problems (i)-(iii) bring ambiguity in deriving and discovering logic, inference and knowledge from the shared data. To address these challenges, principled approaches for the creation and maintenance of ontologies and metadata schemas are required. Further, the metadata schemas should be extensible to allow for domain-specific attributes for harmonization across different scientific domains. It is important to mention that not only for research data but also the extensions should be available for scientific workflow, methods, software, hardware and experimental facility, and laboratory protocols to facilitate a truly domain-agnostic semantic interoperability.

**Organizational interoperability** is focused on the alignment of organizational policies, functions, responsible people, documentations, and processes across different EOSC service providers. The main emphasis is on defining a governance framework to achieve cross-organizations and cross-discipline interoperability. The main challenges include:

- (i) There is a lack of clear description of the “terms and conditions” and “acceptable use policies” that must be adhered by services provisioned by EOSC;
- (ii) Rules of participation do not provide details of how interoperability will be achieved across organizations and domains;
- (iii) Long-term availability of data, services and infrastructures is not always available (can be linked to Findable, Accessible parts of FAIR principles). Therefore, there is a need for defining a clear governance framework concerning the different functions and policies for participation in EOSC, documentation defining unambiguous terms and conditions and acceptable use policies are needed. Furthermore, interoperability certifications for service providers need to be developed so that users are aware of the interoperability levels of different services.

**Legal interoperability** primarily concerns data access governed by various forms of intellectual property rights (e.g., licensing, copyrights, etc.), general data protection regulation (GDPR), private and sensitive data and enabling legal instruments. Ensuring legal interoperability is very challenging due to the following:

- (i) Data reuse is difficult without clear information about the rights and legal conditions for reusing the data;
- (ii) Different datasets may have different licenses not compatible with each, thus making it difficult to combine and use them together;
- (iii) The scope of national copyrights may vary across jurisdictions, making it difficult for users to reuse the data;
- (iv) Separate licensing for different embedded objects in a dataset (e.g., photos in a dataset may have different licenses) can be confusing;



- (v) Users' rights for using the data (e.g., commercial use) may change with the passage of time;
- (vi) GDPR introduces strict constraints on sharing and processing of personal and sensitive data resulting in disproportionate costs on safeguarding personal data and obtaining individual consent for individual datasets;
- (vii) Each EU member state has a different guideline for data privacy impact assessment (DPIA) of high-risk personal data. In the next section, we summarize the key EOSC recommendations pertaining to the above four interoperability criteria.

Legal aspects need to be supported by ethical consent principles for specific research domains that should be implemented in real research projects. Ethical consent is especially important in life science and humanitarian research and less relevant in Computer Science research and projects.

### **2.2.2. EOSC interoperability recommendations**

**Technical:** (i) EOSC recommends that all the services should be using open specifications whenever possible; (ii) A common security and privacy framework including a common authentication, authorization mechanism should be used; (iii) A clear policy for persistent identifiers (PID) for research data, infrastructure and software should be defined; and (iv) data should be made available in a different format for ease of accessibility.

**Semantic:** (i) All the concepts, metadata and schemas should use clear, precise and publicly available definitions which are referenced with PIDs; (ii) a minimum metadata model should be used to describe all the research data for ease of discovery; (iii) metadata should have extensibility options for the inclusion of domain-specific information; and (iv) semantic artifacts should contain the associated documentation.

**Organizational:** Rule of participation should be clear for the resource/service providers with well-defined management functions.

**Legal:** (i) compatibility of EOSC licences with member state licenses should be ensured and there should be a clear alignment of the member state legislations with EOSC legislations; (ii) GDPR compliance of personal data should be adhered; and (iv) policy and guidelines w.r.t. patent filing, trade secret disclosure and data access restriction should be harmonized across participating member states.

### **2.2.3. EOSC Interoperability and Composability Model**

The EOSC Interoperability and Composability Model is defined in the EOSC Future project [6] that provides guidelines for implementing the general (high level) *EOSC Interoperability Framework*. This provides the vision for connecting different kinds of resources provided

across thematic domains and infrastructure boundaries together. Such model assumes that multiple components created by EOSC community may have their own technical implementation and the model provides a view how different independent systems, services, data and resources operated within different domains, can be composed to create a homogenous operational system (adopting a bottom-up approach rather than a top-down approach to interoperability).

The *EOSC-Core*, via the *EOSC Interoperability Framework*, aims to implement this vision by offering EOSC Providers a flexible framework to integrate with the EOSC itself and to describe the relationship between their resources and existing standards and guidelines, thereby becoming an enabler to mediate, bridge, and interoperate between different domains.

The *EOSC Interoperability Framework* also states that the EOSC IF will be composed of a wide range of policies and guidelines on standards and APIs which will be promoted within EOSC.

The wider EOSC IF implementation will provide guidelines for providers to connect resources to *EOSC-Exchange* but will also provide guidelines to be adopted within services made available through *EOSC-Core*<sup>6</sup>, supporting the composability and integration of resources across boundaries. The diagram below shows the different types of composition, integration, and interoperability that can be encountered in the EOSC landscape.

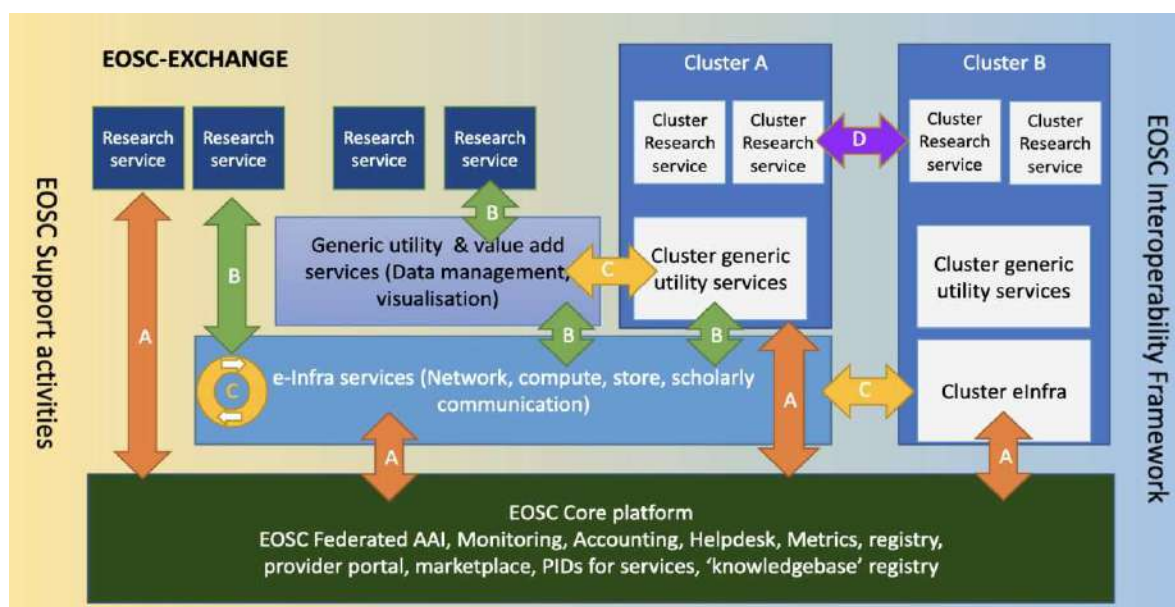


Figure 1. *EOSC Interoperability and composability models* [6].

The diagram on Figure 1 shows the elements of *EOSC-Core* and *EOSC-Exchange*, connected and supported by the *EOSC Interoperability Frameworks* and Support activities. Vertical arrows



represent **vertical integrations** (integrating a resource with more basic and/or common resources and functions), while horizontal arrows represent **horizontal integrations** (connecting peer resources) to add value. These two categories can be further divided into subcategories represented with the letters A, B, C and D (actually defining necessary APIs):

#### **Vertical integrations:**

**Type A:** Support composability of a resource with resources from the *EOSC-Core* to make the resources interoperable in EOSC (e.g., make resources discoverable via the *EOSC-Exchange*, integrate with the order management system and helpdesk to lower the barrier of access and to provide support to the users). It adds significant value for users and providers, as it makes the user experience more coherent, and for providers it adds value without them having to do further development or saves effort on developing the functionality themselves. Type A is related to the *EOSC-Core*; work to enable composability of Type A connections are realised by defining necessary API using standard OpenAPI approach.

**Type B:** Support composability of a resource with *Horizontal Services*, typically provided by e-Infrastructure providers and services, to enrich the resource with additional features and easy/elastic/on-demand access of EOSC resources (e.g., a materials science service from a Science Cluster is integrated with a horizontal cloud computing service from an e-Infrastructure).

#### **Horizontal integrations:**

**Type C:** Support composability of the same type of services or resources that belong to different e-Infrastructure domains or clusters (e.g. a horizontal data management service from an e-Infrastructure is integrated with data management functions and data from a cluster, or integration between e-Infrastructure services from different organisations).

**Type D:** Support composability of cross-domain resources to create added-value solutions to handle complex scientific problems (e.g., integration of different experimental facilities or application data processing for different interconnected tasks). Type D composability will require more solutions at the semantic, organisational and possibly legal layers.

#### **2.2.4. Conclusion of EOSC IF**

Here, we summarize the key features of the EOSC interoperability framework as the following:

- (i) EOSC interoperability framework is a set of *not-so-specific* guidelines to ensure smooth integration of infrastructure services and seamless exchange of research data across the EOSC ecosystem. EOSC-IF is derived from the European Interoperability framework which defines interoperability of an information



technology system by four key elements, i.e., technical, semantic, organization and legal interoperability;

- (ii) The FAIR principles, federated resource and user management and legal compliances pertaining to privacy, licensing and governance by European Commission are at the core of the EOSC-IF framework;
- (iii) In terms of implementation, EOSC FDO is defined as the basic building block for describing EOSC services and data, which should embed all four layers of interoperability;
- (iv) EOSC does not (re)invent any new mechanisms and techniques to ensure technical (service specifications, AAI, PID, etc.) and semantic (semantic artifacts, metadata models) interoperability, but rather, it provides a framework that advocates to utilize the open specifications for implementation of various building blocks to facilitates technical and semantic level interoperability. For example, EOSC intends to use the popular AARC-BPA for AAI, provides a minimum metadata model that is derived from the existing metadata models such as Dublin core and recommends CrossRef XML schema for the storage and distribution of scholarly publications.

From the SLICES point of view, EOSC-IF provides guiding principles that would be highly useful in defining the SLICES interoperability framework to ensure its integration with EOSC for the publication of SLICES services and enable the research data exchange.

### ***2.2.5. Recently Developed EOSC Data and Metadata Management Services and Tools***

Data and metadata management services and tools are essential resources that will enable building interoperable and FAIR compliant Data Management Infrastructure. FAIRCore4EOSC project<sup>7</sup> developed valuable tools important to FAIR compatible Data Management Infrastructure<sup>8</sup>. This includes:

**Compliance Assessment Toolkit (CAT).** The Compliance Assessment Toolkit will support the EOSC PID policy with services to encode, record, and query compliance with the policy. To do so, a wide range of compliance requirements (TRUST, FAIR, PID Policy, Reproducibility, GDPR, Licences) will be evaluated as use cases for the definition of a conceptual model. At the same time, vocabularies, concepts, and designs are intended to be reusable for other compliance needs: TRUST, FAIR, POSI, CARE, Data Commons.

---

<sup>7</sup> <https://faircore4eosc.eu/>

<sup>8</sup> <https://faircore4eosc.eu/eosc-core-components>



**EOSC Data Type Registry (DTR).** DTR enables the registration of PID metadata elements. This will allow a machine actionable standardisation of PID metadata.

**Metadata Schema and Crosswalk Registry (MSCR).** The MSCR allows registered users and communities to create, register and version schemas and crosswalks with PIDs. The published content can be searched, browsed and downloaded without restrictions. The MSCR also provides an API to facilitate the transformation of data from one schema to another via registered crosswalks.

**EOSC PID Graph (PID Graph).** PID Graph support the harvesting of the PID Graph metadata. The API service and data dumps are made available for the community to ingest and reuse the metadata seamlessly. In addition, the component will focus on the interoperability framework for graph data exchange.

**EOSC PID Meta Resolver (PIDMR).** The PID Meta Resolver is a generalized resolver for mapping items into records. PIDMR will support researchers in their daily work so they can easily make use of the PIDs (resolution, metadata).

**Research Activity Identifier Service (RAiD).** The RAiD provides persistent, unique and resolvable information for research projects. The EOSC RAiD will mint Persistent Identifiers for research projects, which will allow users and services to manage information about project-related participants, services, and outcomes.

**EOSC Research Discovery Graph Service (RDGraph).** The EOSC Research Discovery Graph Service (RDGraph) delivers advanced discovery tools across EOSC resources and communities. The RDGraph builds upon the EOSC catalogue's content, extending it with additional entities like the Research Activity Identifiers (RAiDs).

**EOSC Research Software APIs and Connectors (RSAC).** RSACs ensure the long-term preservation of research software in different disciplines. The component will improve interoperability between various infrastructures catering to research software.

**EOSC Software Heritage Mirror.** The EOSC Software Heritage Mirror (SWHM) is a copy of the Software Heritage universal source code archive, operated in agreement with, but independently from the Software Heritage organization.

The following tools are identified as a current preparatory stage of the SLICES-RI development: Metadata Schema and Crosswalk Registry (MSCR), EOSC PID Graph (PID Graph), EOSC PID Meta Resolver (PIDMR), EOSC Research Software APIs and Connectors (RSAC), that will provide a platform for internal SLICES interoperability and future integration with EOSC services and infrastructure, and potentially with other Ris adopting EOSC Interoperability Framework.





### 2.3. FAIR Data Principles Adoption by ESFRI

FAIR data principles are well developed wide research community and created a basis for EOSC development as an environment and platform to support European research and for effective research data sharing. Recently published ESFRI Opinion paper on FAIR data principles implementation by RIs [8] provided an expert view (by ESFRI EOSC Task Force and Steering Board expert group) on the role of FAIR data principles for the enhancement of transparency in the research process, improvement of reproducibility of results and a novel opportunity for reuse of data, software and analysis of results as well as allowing/facilitating transdisciplinary and interdisciplinary research, that can be achieved with using EOSC.

The paper defined the Quality Assessment FAIR Data (QFAIRD) that includes the following key aspects:

1. The current FAIR data implementation level that is generally related to newly produced data and research projects and activities connected to EOSC linked projects and activities, also adopted by data drive research domains and RIs such as ELIXIR, ENVRI, LHC, EUDAT, and others.
2. The goal of ideal FAIR data productivity and quality control is relevant to individual RIs or specific clusters. This includes the definition of the metadata commons that will be required for data preservation (collection, curation, archiving) for at least 10 years. EOSC should play an important role in connecting researchers across thematic and national domains and providing necessary tools for FAIR data management, supporting the whole research data lifecycle.
3. The bottlenecks that should be addressed to increase FAIR data productivity including the following aspects: insufficient and non-coordinated training of data curators and stewards to support the whole FAIR value-chain, resistance to the adoption of the FAIR data principles starting from the research planning and design, as well as limited specialist support and infrastructure resources to support data management and sharing. Full FAIR data principles implementation will require analysis and assessment of both RIs and e-Infrastructure (European and national).
4. The needed EOSC services to improve FAIR data principles implementation and productivity. The paper notes that the EU EOSC Node shall provide core services to enable the FA (Findable and Accessible) data/objects from the production side to become progressively IR (interoperable and reusable) for general users. Thematic and national resources (such as RI or organisational nodes) shall find in the EOSC the complementary services to give full value to the data produced, curated, and archived also progressively aligning to good practices and reference standards.



The paper also presents the current view on the data management aspects facilitated and enabled by the growing use of AI and ML in modern and future research and possible improvement of the data management process with AI and ML services. AI enabled research will require new data management procedures and infrastructure services that can be achieved by interplay of FAIR and AI.

## 2.4. SLICES Interoperability Framework definition and recommendations

This section will refer to SLICES Interoperability Framework initially proposed by SLICES-DS and provide updates based on recent SLICES, EOSC and ESFRI documents.

### 2.4.1. Requirements of SLICES-IF

Since SLICES aims to provide a distributed pan-European experimental research platform by jointly utilizing the geographically dispersed computing, storage and networking RIs, it is highly important that the different nodes interacting in the experimental workflow are interoperable with each other. For example, considering a mobile edge computing use case, compute, storage and networking resources from different nodes would be used. In such a scenario, it is necessary that resource description, availability, execution and data exchanges are smooth. This can only be assured if a common interoperability framework is adopted across SLICES so that different subsystems have a common understanding of resources, data/metadata and are on the same page with respect to the licensing, copyright and privacy requirements.

The SLICES-IF mainly targets four communities:

- **Users(research) community:** The SLICES-RI would provide a rich set of infrastructure services (e.g., 5G/6G, IoTs, etc.) and research data that can be accessed by research communities and industries across the European research area. For researchers accessing SLICES, it is important that smooth navigation across different SLICES resources and integration of resources is possible to accomplish the research/experimental workflow. Further, it is important to ensure that the research data (produced as a result of experimentation by the SLICES user or existing data from other SLICES users) conforms to the FAIR principles;
- **EOSC:** EOSC being the flagbearer of future European open digital sciences is a top priority for SLICES due to: (i) publishing of SLICES RIs, services and data through the EOSC portal for a wider reach; and (ii) SLICES users will benefit from the rich set of EOSC services that are expected to even grow more in the future;
- **External RIs:** It is pragmatic to think that some large-scale complex experiments may require SLICES resources to be clubbed with the external RIs such as the public clouds. This means the SLICES-IF should ensure that the state-of-the-art and widely accepted approaches for technical and semantic layers implementation are adopted to ensure a wider interoperability.
- **SLICES consortium partners:** Members of SLICES consortium will provide different resources (hardware and software services) pertaining to computing, data,



networking, storage and domain specific resources. The interaction of these resources is needed to execute complex scientific experiments. Therefore, a common framework for interoperable resource description, communication protocols, resource integration, semantics and data/metadata models are required.

To achieve this, likewise the EOSC-IF, the SLICES-IF will be built upon the foundations led by the European Interoperability Reference Architecture (EIRA), where interoperability is classified at four layers, namely: (i) technical, (ii) semantic, (iii) organizational; and (iv) legal. Although the target audience for EIRA (governance and administration) was very different of what compared to SLICES, core principles and objectives remain the same. Additionally, the different components (in particular technical and semantic) of SLICES-IF would be chosen in such way that SLICES is fully interoperable with EOSC for uninterrupted data exchange pertaining to the use of EOSC services and research data by SLICES as well as to enable the publications of SLICES infrastructure, services and data through EOSC portal. Considering EOSC, the following types of data would be exchanged *to-and-fro* between SLICES and EOSC:

- **EOSC services data:** EOSC provides a plethora of digital tools and services targeting various scientific domains ranging from data processing to biological science and environmental sciences to astronomy. These tools can be highly useful for many future SLICES users. Therefore, it is important that EOSC services can be smoothly integrated into the SLICES workflow to reap the benefits of EOSC services;
- **SLICES resources data:** EOSC being the leading initiative on European digital sciences, SLICES aims to publish its services through EOSC portals due to its wider reach across research communities. To ensure this, SLICES service specifications should conform to the service specifications implemented by EOSC;
- **EOSC Research data:** EOSC ecosystem is expected to produce a huge amount of research data from varieties of scientific domains. This research data can be utilized by SLICES users for several purposes including conducting new experiments, reproducing the existing scientific experiments and for algorithmic benchmarking, etc. In order to consume the research data produced by the EOSC experiments and allow a smooth integration with the SLICES experimental workflow, SLICES need to adopt data models that are compatible with the EOSC data models;
- **Research data and publications produced by SLICES users:** Research publications are important outcomes of scientific experiments. In many cases, research publications are accompanied by models and datasets forming research artifacts that are quintessential for reproducibility of scientific experiments. The large-scale scientific experiments performed through SLICES will generate a wealth of research data/models that can be further used by other academic and industry researchers/ scientists. To maximize the reusability and extensibility of this data, it is important that the data should be stored in formats that can be easily accessed and processed by researchers and developers;
- **Metadata:** Metadata is key to ensure the discovery/findability of scientific resources and research data. Also, metadata contains vital contextual information about the scientific experiments such as the environment in which the experiment was conducted such as the information of the hardware used, software environment and dependencies, etc.



Based on the above discussion, the key requirements of SLICES-IF can be summarized as the following:

1. SLICES-IF is required to be fully compatible with EOSC-IF for research data exchange. This requires well defined semantic and metadata catalogues adhering to EOSC specifications. This should be also supported by federated AAI used by EOSC services.
2. It is important that SLICES adopt open specifications for the representation of services and tools. This is necessary to ensure that tools and services available under the EOSC ecosystem are easy to integrate within the SLICES experimental workflow.
3. A common PID framework for SLICES resources (infrastructure, tools, models, catalogues and data) is required.
4. Easy to use APIs for access through the web as well as command line interfaces are required for user and resource management.
5. The organizational framework should ensure clear rule of governance and participation.
6. The legal framework must provide unambiguous rules for licensing, data use, copyrights and privacy.

#### **2.4.2. SLICES-Interoperability Framework**

Similar to EOSC, FAIR digital object would constitute the basic building block of SLICES-IF with provisions for interoperability at all the layers. Figure 2 below shows the SLICES-IF in the context of EOSC-IF and FAIR digital object guidelines. To achieve interoperability at each layer of technical, semantic, organizational and legal, corresponding enablers fulfilling the concerned requirements will be adopted. For example, to achieve semantic interoperability, SLICES-IF would clearly define the semantic artifacts and metadata following the FAIR principles. A clear hierarchy for semantic artifacts will be developed starting from the less formal representation such as XML schemas and UML models to the highly formal representations such as ontologies. For example, an object class represented in a UML model (less formal) can be linked to an ontology (more formal) which defines the concept of that particular class.



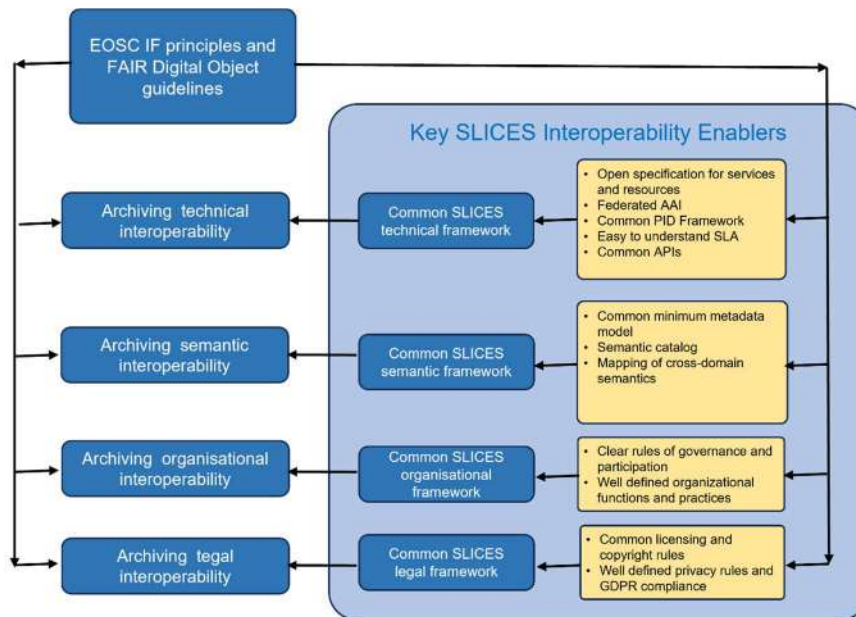


Figure 2. SLICES Interoperability framework and its interaction with EOSC-IF.

SLICES-IF defines four groups of interoperability recommendations in compliance with the EOSC Interoperability framework:

- **Technical interoperability** is defined as the ability of different information technology systems and software applications to communicate with each other and seamlessly exchange data;
- **Semantic interoperability** refers to the ability that the exchanged data to be understood well and have a common meaning across different entities of the EOSC ecosystem;
- **Organizational interoperability** is focused on the alignment of organizational policies, functions, responsible people, documentation and processes across different EOSC service providers. The main emphasis is on defining a governance framework to achieve cross-organizations and cross-discipline interoperability;
- **Legal interoperability** primarily concerns data access governed by various forms of intellectual property rights (e.g., licensing, copyrights, etc.), general data protection regulation (GDPR), private and sensitive data and enabling legal instruments.

A detailed list of all recommendations is provided in Table 1, which is derived from the EOSC IF recommendations, however at current stage of the SLICES-RI development, we primarily focus on the technical and semantic interoperability that are relevant to experimental infrastructure and data management services.





**Table 1 SLICES Interoperability Framework Recommendations**

Layer	Recommendation
Technical (services but not yet infra)	<ul style="list-style-type: none"><li>• Open specifications for SLICES services published on the SLICES Portal (adopting and ensuring interoperability with EOSC services description).</li><li>• A common security and privacy framework (including Authorisation and Authentication Infrastructure).</li><li>• Easy access to data sources available in different formats.</li><li>• Coarse-grained and fine-grained dataset (and other research objects) search tools.</li><li>• Interoperability and integration with the EOSC PID infrastructure, compliance with EOSC PID policy.</li></ul>
Semantic (Metadata)	<ul style="list-style-type: none"><li>• Clear and precise, publicly-available definitions for all concepts, metadata and data schemas.</li><li>• Semantic artifacts, preferably with open licenses.</li><li>• Associated documentation for semantic artifacts.</li><li>• Repositories of semantic artifacts, rules with a clear governance framework.</li><li>• A minimum metadata model (and crosswalks) to ease discovery over existing federated research data and metadata.</li><li>• Extensibility options to allow for disciplinary metadata.</li><li>• Clear protocols and building blocks for the federation/harvesting of semantic artifacts catalogues.</li></ul>
Organisational	<ul style="list-style-type: none"><li>• Interoperability-focused rules of participation recommendations.</li><li>• Usage recommendations of standardised data formats and/or vocabularies, and with their corresponding metadata.</li><li>• Clear management of permanent organisation names and functions</li></ul>
Legal	<ul style="list-style-type: none"><li>• Standardised human and machine-readable licenses, with a centralised source of knowledge and support on copyright and licenses.</li><li>• Permissive licenses for metadata (and preferably for data, whenever possible), where is CC0 preferred over CC BY 4.0.</li><li>• Identification of different parts of a dataset with different licenses.</li><li>• Compatibility with EOSC-recommended licenses and hosting national regulations.</li><li>• Tracking of license evolution over time for datasets.</li><li>• Harmonised policy and guidance to dealing with cases where patent filing or trade secrets may be compromised by disclosure.</li><li>• GDPR-compliance for personal data.</li><li>• Alignment between Member States national legislations and EOSC.</li></ul>



The proposed the SLICES Interoperability Framework enables the following features:

- I. Support integration into a distributed pan-European research infrastructure that shall be accessible remotely.
- II. Allow deploying and operating large scale multi-site experiments on SLICES infrastructure, including external services and interaction with other RIs.
- III. Use well defined, where possible standard, interfaces/API that support both services/resources interaction and data access and sharing (e.g., research data, computing and storage modules, providers and users).
- IV. Comply with the FAIR data principles where interoperability is essential to ensure that the SLICES data can be integrated with the scientific workflows for analysis, storage and processing.
- V. Interaction with EOSC and other RIs to allow external services to be used in SLICES-RI and SLICES-RI services are accessible by external RIs and researchers; the goal is to consolidate the scattered experimental facilities and European research infrastructures.



## 3. SLICES Data Management Infrastructure for Reproducible Experimental Research

---

### 3.1. Experiment Automation and Experimental Research Reproducibility in SLICES

#### 3.1.1. Experiment Reproducibility as a Service in SLICES

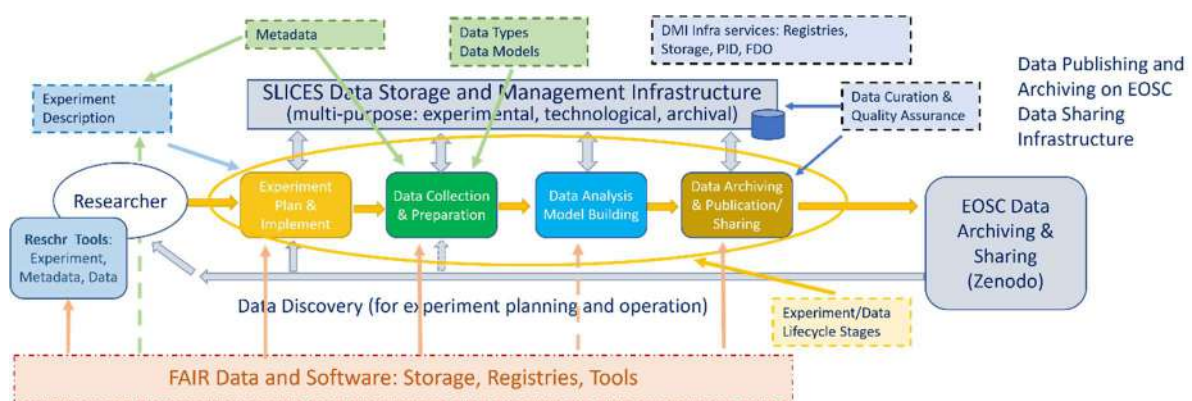
SLICES is dedicated to support experimental research reproducibility as one of the core principles of Open Science. The primary focus will be on the **repeatability** and **reproducibility** with the future support of **replicability** and shared distributed experiments orchestration. Reproducibility of experimental research imposes additional requirements on the reproducible experiment setup, including resource provisioning, experiment environment setup, and experiment and data lifecycle management. The following aspects will be addressed:

- Documenting all relevant parameters and environment for experiments;
- Automate the documentation of experiments; a well-structured experiment workflow may serve as documentation;
- Offering Experimental Research Reproducibility as a Service (ERRaaS) will be beneficial to the research community by:
  - Reducing the amount of work for experimenters to create reproducible experiments;
  - Diminishing the load for other **researchers** to recreate and re-run experiments;
  - Decreasing the overall energy consumption and environmental impact of large-scale and complex experiments;
- Automating the entire experiment (setup, execution, evaluation), including energy optimization;
- Making reproducibility an integral part of the experiment design will serve another purpose of documenting infrastructure design and usage patterns that can be re-used by other RIs intending to use new DI technologies in their research.

#### 3.1.2. Experimental Data Management Stages

Management of experimental data is a key aspect of SLICES-RI, and it includes several services that must support all stages of the experimental data lifecycle. As illustrated in Figure 3, SLICES-RI will operate distributed federated Data Storage and Management Infrastructure to support activities typical for experimental research, such as experiment planning and deployment (as explained in the previous sections), the discovery of data from internal data archives and external data sources and data publication.





**Figure 3. SLICES Experiment lifecycle and data management stages and supporting infrastructure.**

Each data lifecycle stage, i.e., experiment setup, data collection, data analysis, and finally, data archiving, typically works with its own datasets, which are linked and their transformation must be recorded in the process that is called lineage (which can also be extended to provenance for complex linked scientific data). All staged datasets need to be stored and possibly re-used in later processes.

SLICES DMI establishes policy for data governance and management, including data security and quality assurance (data curation), that are supported by corresponding infrastructure tools. Figure 3 also illustrates stages and activities where the FAIR-compliant metadata must be applied.

Many experiments may require already existing datasets that will be available in the SLICES data repositories or can be obtained/discovered in EOSC data repositories as illustrated by links to EOSC data services.

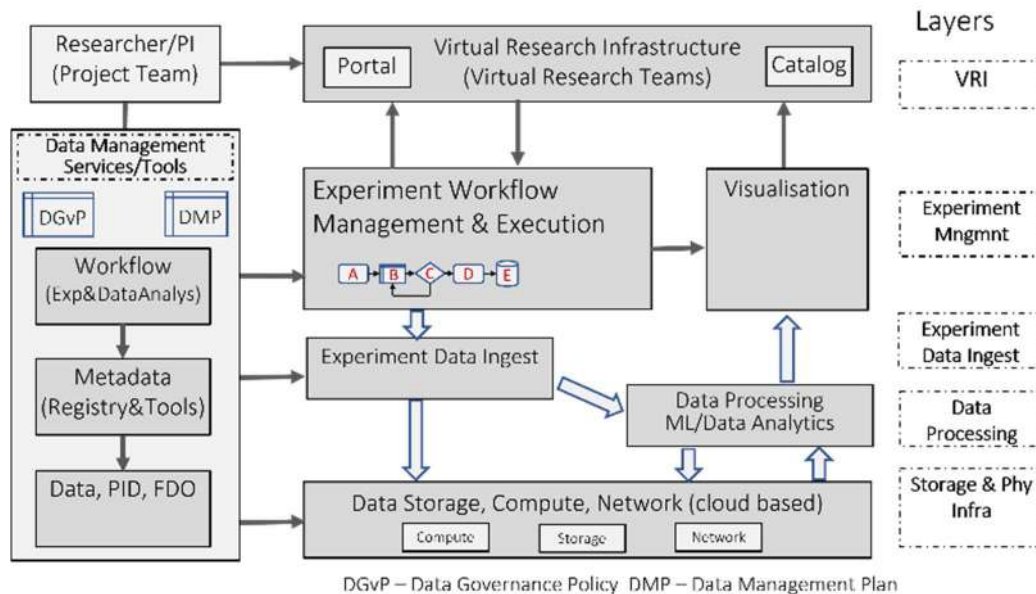
## 3.2. SLICES DMI Architecture and Requirements to Support Experimental Data Management

### 3.2.1. SLICES DMI Architecture

The consistent definition of DMI will impose specific requirements to the SLICES Reference Architecture and will require the implementation of special services to support data collection, data management, and data sharing at all functional layers of the SLICES infrastructure.

DMI creation will have a staged process starting with the bottom-up data and metadata services integration with the existing SLICES infrastructures and experimental sites, delivering a Minimum Viable Product (MVP). DMI will follow the SLICES-RI evolution and incorporate new data and metadata tools development, primarily coordinated and facilitated by EOSC. The long-term vision for DMI should incorporate all these factors by adopting a sustainable architecture design principles. The proposed SLICES DMI Architecture (see ) has been

discussed at SLICES technical meetings and presented at the recent GLOBCOM2024 Conference [9].



**Figure 4. Architecture of the SLICES Data Management Infrastructure.**

DMI Architecture definition includes hierarchical service layers (allowing horizontal and vertical composition and integration) and cross-layer services defined as planes. Such architecture definition allows separating data management and governance functions, concerns, and actors/roles. The following layers and planes are defined:

- **Layer 5** - Virtual Research Environment (VRE) and researcher portal or dashboard.
- **Layer 4** - Experiment configuration and management.
- **Layer 3** - Experimental data collection/recording that applies data models and metadata for experimental data.
- **Layer 2** - Data processing that performs data analysis, allows ML models building for processes and systems-under-test, and ensures the computation workflow scalability and portability.
- **Layer 1** - Data Storage, Archiving, Exchange that represents the physical or virtual infrastructure resources for data or metadata storage, archiving and publication. This layer supports FAIR Digital Object (FDO), PID registries and gateway/proxy.

The Data Management Plane includes Data Management Services and Tools that can be used by each of the DMI layers:

- Data Management Plan and Data Quality Assurance, FAIR compliance;
- Metadata registries and tools;
- Data Governance Policy, Data Security, GDPR compliance.





VRE and user portal may benefit from implementing the Platform RI as a Service (PRIaaS) architecture that is compliant with the TeleManagement Forum Digital Platform Reference Architecture for telecom system [10].

### ***3.2.2. Requirements to Support the Experimental Data Management***

Data Management is an essential component of the SLICES-RI infrastructure that includes data collection from experiments (including experiment description and measurement data), data storage, data preparation, data lineage and quality assurance, data publication, and data sharing.

The following are requirements for SLICES DMI for experimental data issued from best practices and use cases analysis in the SLICES-DS project:

- **RDM1** - Distributed data storage and experimental data(set) repositories should support common data and metadata interoperability standards, in particular, common data and metadata formats. Outsourcing of data storage to the cloud must be protected with appropriate access control and compliant with the SLICES Data Management policies.
- **RDM2** - SLICES DMI should support the whole research data lifecycle. It should provide interfaces to experiment workflow and staging.
- **RDM3** - SLICES DMI shall provide PID (Persistent Identifier) and FDO (FAIR Digital Object) registration and resolution services to support linked data and data discovery that should be integrated with EOSC services.
- **RDM4** - SLICES DMI must support (trusted) data exchange and transfer protocols that allow policy-based access control to comply with the data protection regulations.
- **RDM5** - SLICES DMI must enforce user and application access control and identity management policies adopted by the SLICES community that can be potentially federated with the EOSC Federated AAI.
- **RDM6** - Procedures and policies must be implemented for data curation and quality assurance.
- **RDM7** - Certification of data and metadata repositories should be considered at some maturity level following certification and maturity recommendations by RDA.

SLICES DMI will be designed in such a way that would allow integration with the EOSC federated data infrastructure and services to allow a hybrid data management infrastructure that may include both its own data storage, as part of the private cloud, and external data storage offered by EOSC and EGI communities. The use of public cloud storage and file sharing services will be regulated by data management policies.

## **3.3. Metadata to Describe Infrastructure and Experiment**

### ***3.3.1. General Metadata Definition and Services***



Metadata are an important component of DMI that provide a basis for services interoperability, experimental research reproducibility, effective data sharing and discovery. Effective and consistent metadata management is the foundation of the FAIR data principles implementation. All data are defined by the data models, metadata, data formats and data types. Metadata are defined as part of the data model.

For SLICES as a RI for experimental studies in digital technologies and ICT, metadata includes three main areas:

- General services description: metadata profiles and metadata will be used for publishing SLICES services in EOSC Catalog and the SLICES services catalog;
- Description of data collected, produced and handled in SLICES-RI that include experimental data, staged/processed data, archival data, publications, reports, and management data. Additional data categorization is required;
- Experiment description that includes all necessary information for experiment reproducibility and deployment.

Two other categories of metadata may be required to support SLICES experiments include:

- Infrastructure descriptions that are required for infrastructure management and monitoring (network devices, network traffic, status and events). This type of metadata is well supported by existing network and service management standards (SNMP MIB-II, DMTF CIM and CIMI);
- Metadata for data processing and lineage, in particular, for data used in ML and AI processes.

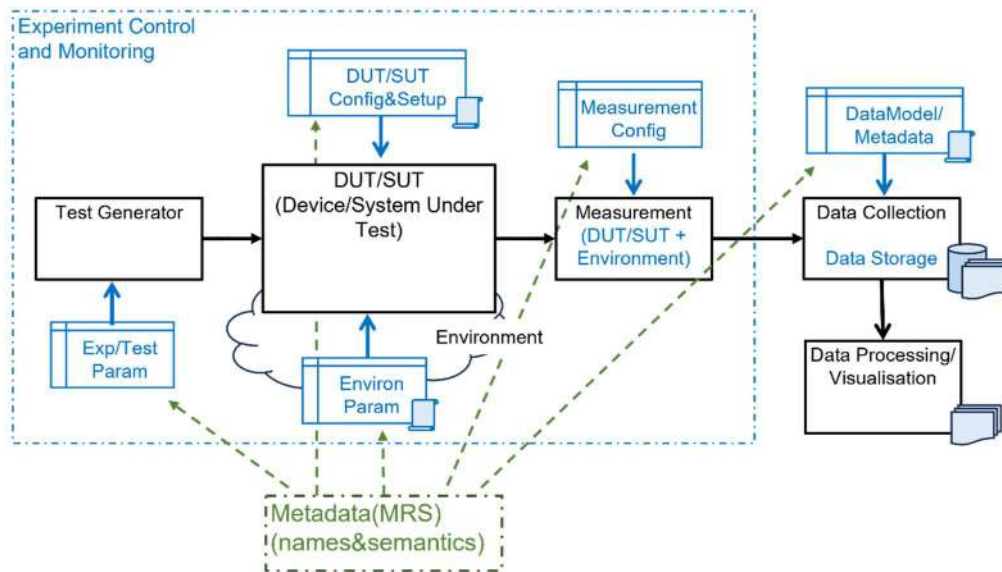
Defining domain-specific metadata requires the definition of the metadata schema and namespaces that create a basis for unique metadata elements identification and consequently discovery, sharing and integration.

### **3.3.2. Experiment Data Model and Required Metadata**

To provide a full experiment description and achieve the experiment reproducibility all experiment setup, control and execution must be supported by the consistent experiment model and documented at the time of execution and during the whole research lifecycle. All documented data must be FAIR compliant: Findable and Accessible after publishing to ensure research sharing, and further Interoperable and Reusable to achieve full reproducibility.

Figure 5 illustrates the generic experiment data model including the following components:

- Device/System under Test (DUT/SUT) model (variables, parameters, environment);
- DUT/SUT Configuration&Setup;
- Test/Stimulus Variables& Parameters, possibly also including environment parameters;
- Measurement (instruments) configuration.



**Figure 5. Generic experiment data model for reproducibility.**

For consistent and reliable experiment description and documenting, the properly defined metadata must be defined and applied for all experiment components and stages. Additional aspect in supporting experimental data collection is the proper data model selection and corresponding database support. The relational data model and corresponding database platform are considered as the most relevant taking into account that the experiment description and setup will use multiple tables (for setup and test generation). This work is undergoing the practical implementation in the project.

### **3.3.3. SLICES Metadata Definition and Requirements**

This section describes data used and collected from the two experimental platforms in SLICES: Plain Orchestration Services (POS) and Blueprint which are in the process of integration for running the experiments. This integrates the experiment management but from the data management point of view it will require smooth data integration addressing common and interoperable metadata definition and correspondingly defining the consistent experiment data model.

#### *3.3.3.1. POS Experiment description and metadata*

Consistent/full experiment description and corresponding metadata must ensure experiment/experimental research reproducibility and FAIR data sharing.

The following data types and metadata are considered essential for consistent experiment description (based on POS implementation and suggestions):

- Experiment abstract model with parameters, input variables and variables under test (as it is known at the beginning);





- Experiment setup/infrastructure, including network equipment and the network topology, including VMs/containers:
  - Hardware: list of major hardware components (e.g., CPU, network cards, memory, storage);
  - Firmware: version numbers of the installed firmware, e.g., BIOS version, CPU microcode version, firmware version of the network card;
  - Software: version of the investigated software and its installed dependencies, version of the installed OS (including version of OS kernel), software installed on the OS.
- Configuration of all infrastructure components, deployment sequence (presumably Ansible playbook, Terraform plan, or Jupyter Notebook);
- Test generators, measurement equipment and sensors (and corresponding infrastructure points):
  - Specification of the generated traffic (the content of the traffic);
  - Specification of the patterns used for the generated traffic (e.g., a distribution of the inter-packet gaps or traffic bursts).
- Environment description (hardware/software);
- Experiment workflow (the usage of pos ensures reproducibility of experiment workflow);
- Data ingest process, data preprocessing and assessment;
- APIs for experiment setup, monitoring, and data collection.

Data models and metadata must be defined for all types of data describing the experiment. For some well-established experiments, data models may be defined for the specific data storage and database type, such as data lakes, SQL database, key-value, document based, or triple storage (for semantic data). Experiment as a Research Object must be assigned a unique identifier and experiment/object type, optionally registered schema and domain namespace.

**Goal of reproducibility:** The goal of reproducible experiments is the reproduction of key performance descriptors for a specific experiment. The exact reproduction of these key performance descriptors may depend on specific hardware and software that may change over time. We rather aim for recording a complete picture of the environment an experiment is conducted in. The recorded information is the foundation to find differences between experiments that may not be obvious during the initial experiment.

The usage of pos ensures the collection of many aspects mentioned before. To generate the necessary information for the experiment description, pos relies on standard Linux tools. Typical tools to automate the hardware description would be `lshw` that lists all hardware (and some of the firmware versions). Additional, more specialized tools may be used for other hardware components, such as `ethtool` for network cards. The service itself may provide



information about the topology, that should be detected regularly (e.g., using tools for topology detection such as `lldp`).

We propose to record the typical format provided by the mentioned tools. A service should provide an abstraction layer to access the mentioned information in a more generalized format to simplify further processing. Tools such as `pos` already provide a common data format (JSON) to unify the output of the different tools.

### *3.3.3.2. SLICES Blueprint Architecture for 5G/6G and related networking technologies*

SLICES Blueprint defines a basic/core/instant infrastructure setup that can be used for running experiments on 5G/6G and related networking technologies. Blueprint is defined in a modular way that allows defining basic building blocks or design patterns that can be composed in a different way to create a platform for running different experiments.

To achieve effective composability and flexible customization of the SLICES experimental setups, the following data and metadata should be defined (similar for the general experiment setup as described above):

- Services deployed in the Blueprint and corresponding APIs;
- Input and output data or signals;
- Configuration of all elements, including RAN (RU – Radio Units, UE – User Equipment), core 5G network, dedicated network (VPN or VPC), network switches, servers;
- Computational nodes/instances type and configuration: hardware (AMD/Intel, RAM, OS, firmware);
- Operational environment that may need to be documented for experiments;
- Infrastructure design patterns or templates (in the form of Ansible playbooks or Terraform plans);
- Monitoring or measurement point and corresponding API;
- Experiment-specific or other data collected in the infrastructure.

### **3.3.4. Experiment workflow management with POS and metadata extraction**

#### 3.3.4.1. Plain Orchestration Services

POS is a test controller tool that aims to make network experiments portable and reproducible. By installing the POS daemon for an infrastructure service, experiments become portable - they should work exactly the same on any given infrastructure. This is, of course, given that the same hardware and software is available on all the clusters that are being used to run experiments. Simultaneously, experiments become easily reproducible as researchers can take any experiments and run it on any infrastructure and verify the results.



The POS daemon should be installed on the management node of cluster and manages the following within that cluster:

- Infrastructure devices;
- Experiment orchestration;
- Collection of experiment data.

In turn, researchers can interact with the POS daemon through the API, which is made available on the management node as a command line interface through a Python package "poslib".

The setup of experiments using POS can vary quite a bit, depending on preferences and toolings used by researchers and their experiments. To understand what an experiment using POS would look like, as well as what metadata could be present, we can explore the pos-artifacts repository.

The pos-artifacts repository contains multiple directories and the README explains what data is in there:

- **\_includes:** part of the website generator;
- **data:** data files for figures;
- **experiment:** contains experiment scripts;
- **figures:** plots;
- **plots\_scripts:** scripts for plotting;
- **results:** contains measurement results from experiment runs;
- **template:** part of the website generator;
- **web:** part of the website generator;
- **.gitignore:** list of files that should not be pushed to git;
- **README.md:** Readme file that contains a description of the contents of the repo;
- **\_config.yml:** part of the website generator;
- **index.html:** part of the website generator;
- **publish.py:** part of the website generator.

From this list we deduce that the metadata about the experiment could largely be found within the experiment directory, so we will dive into that directory:

- **dut:** folder with config/script for the device under test;
- **loadgen:** folder with config / script for the load generator of the experiment;
- **experiment.sh:** main experiment script;
- **global-variables.yml:** global variables for the experiment;
- **loop-variables.yml:** loop variables for the experiment.





After analysing the (meta)data in the POS artifacts repository and the generic experiment model, a relational or hierarchical data model can be implemented. Further works are going on to explore these options and how it can assist the recent development of MRS and inclusion into the proposed DMI.



## 4. SLICES Metadata Services for (Experimental) Research Data Sharing

To support the operation of the future SLICES-RI, the project will design a flexible metadata model covering all data produced in the projects from individual experimental research to the final research objects to be published on the recognized platforms, in particular, EOSC operated and EOSC recognized. The intended data model will consist of compulsory metadata attributes that are domain-agnostic (e.g., Persistent Identifier, Creator, Name, Description) and can describe any digital object besides data, such as software, tools and services, ensuring that it conforms to FAIR principles and beyond. Where appropriate, SLICES will support additional optional metadata attributes accompanied by their metadata model to further enhance the description of the object (e.g., data size, duration and format for datasets and user manual, access policy for software). Consequently, the metadata will comprehensively describe data objects and support a plethora of functions to query and retrieve them. The metadata model will allow for easy addition of new attributes or new types/categories.

### 4.1. SLICES Metadata Definition and Registry

#### 4.1.1. Metadata Design Objectives

SLICES aims to facilitate the management and sharing of the research data intuitively and safely, respecting the creators' rights, while ensuring compliance with open access and FAIR principles. The previous section provided several design principles that should be considered to design a robust, efficient, and effective metadata model. These principles were transformed into appropriate objectives that must be met to ensure that the metadata models act as foundational components in the SLICES research ecosystem, enabling the replication and provenance of experiments.

- **Flexible Hierarchical Metadata Model (Domain-agnostic and Domain-specific):** Since there is no “one-size-fits-all” metadata standard to address all current and future requirements, SLICES considers the utilization of a hierarchical model consisting of compulsory metadata attributes that are domain-agnostic and can describe any digital object (e.g., data, services) ensuring that it conforms to FAIR principles and beyond. Where appropriate, SLICES supports additional optional metadata attributes accompanied by their metadata model to further enhance the description of the object. Consequently, the model comprehensively describes data objects and supports a plethora of functions to query and retrieve them. The model is flexible and open, allowing extensions with new attributes or new types/categories as well as with new additional hierarchy levels.
- **Hybrid Metadata Production (human and machine-generated):** SLICES streamlines metadata production using appropriate tools that can automatically generate/extract metadata (e.g., date of creation). Manual human-based metadata creation is supported (e.g., description, keywords) to enable user-defined metadata.



Furthermore, predefined input validation workflows are employed to improve the quality of data.

- **Wide-ranging Interoperability:** Metadata should include attributes to support different levels of interoperability: *semantic*, which allows internal and external systems to discover and understand what the underlying object is; *legal*, which describes the restrictions in the data; and *technical*, which enables systems to communicate effectively using appropriate catalogues and services. Furthermore, although metadata are stored in the SFDO core metadata format, SLICES must ensure interoperability with other systems (e.g., EOSC, aggregators) by supporting the transformation of the data to other metadata formats (e.g., DublinCore, OpenAIRE). As such, SLICES-RI provides appropriate interoperability services to enable researchers/practitioners, content providers, funders and research administrators to collaborate or utilize an existing platform to do so.
- **Long-term Reusability:** It is expected that through the lifetime of the SLICES project, the metadata model will evolve due to internal (e.g., support for new use cases) or external factors (e.g., introduction of new standards, communication protocols). The metadata model must incorporate appropriate elements to describe the metadata model itself, including specific attributes, such as the metadata version, and utilize services where systems or users can obtain this info. Vocabularies and Standards utilized by specific attributes should also be versioned. This will also enable mappings between different versions of the same metadata records further enhancing reusability and interoperability.
- **Enhanced Discovery:** Discovery should be further supported with descriptors created manually (e.g., keywords) but also automatically. Automatic descriptors may come from automatic metadata data extraction (e.g., creation date, file size) but also using built-in data analysis functions (e.g., term document frequency).
- **Metadata Quality Assurance:** Metadata quality is important for ensuring reusability and interoperability with other infrastructures/platforms and applications that may want to consume data from SLICES. A Metadata Quality Assurance Framework should be put in place as part of the Data Governance Group with appropriate metrics to assess the quality of metadata, its FAIRness, and other objectives.
- **Metadata Governance:** The members of the Data Governance Group should have the knowledge and authority to make decisions on how metadata are maintained, what format is being utilized, and how changes are authorized and audited.

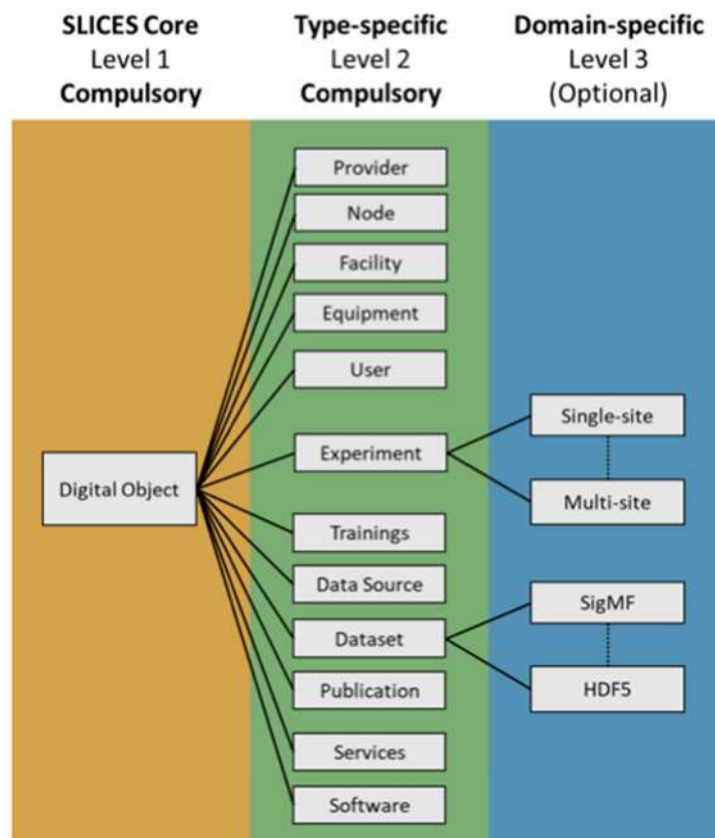
Using the above objectives, we provide the definitions for the metadata profiles in the next section.

#### 4.1.2. SLICES FAIR Digital Object (SFDO)

SLICES implements a flexible hierarchical metadata model consisting of three levels as illustrated in Figure 6. The first level consists of compulsory domain-agnostic information that can describe any digital object (e.g., data, services, publications, tools), ensuring that it



conforms to FAIR principles and beyond. It includes basic information, such as identification, description and its resource type. Management information is also included such as version and metadata profile used.



**Figure 6. SLICES FAIR Digital Object (SFDO).**

Additionally, SLICES employs a second level of compulsory metadata attributes that are type-specific to enhance machine-actionability for specific commonly used types of digital objects, such as data, services, and software. For example, a dataset may have start and end date, a facility may have an address.

Finally, the third level incorporates optional domain-specific attributes to further enhance interoperability for specific communities. For example, the SigMF standard is designed to record signal (e.g. wireless radio transmissions) data and includes some predefined metadata attributes, such as `core:sample_rate`, `core:datatype` and `core:hw` (hardware that was used for signal recording). This information is exposed as Level 3 SFDO attributes to facilitate enhanced discoverability across different such datasets.



#### 4.1.3. Design Considerations

Metadata design and implementation are mission-critical, core activities contributing to the efficient and effective sharing and reuse of data within and outside SLICES. As such, high-quality metadata is as important as data, and adequate resources should be allocated to cater for metadata operations. This means that appropriate protocols and procedures should be put in place on metadata operations (i.e., creation and operational workflows) ensuring proper management and stewardship. These protocols should also incorporate automated metadata production (e.g., metadata extraction) whenever possible and appropriate, decreasing both human effort and errors (e.g., use of predefined lists from established vocabularies) thus further contributing to enhancing the quality of data.

Metadata management should also be considered as an evolutionary process. Metadata attributes may be transformed, adapted, or enhanced over time to cater for new or revised requirements. As such, metadata should be flexible and open, allowing for enrichment. Furthermore, metadata change management should be part of data governance's procedures utilizing appropriate mechanisms, such as versioning.

The following list summarizes the key metadata design principles and considerations that influenced the design of SFDO:

- **Requirements Analysis:** Metadata models need to accommodate the current and future needs of the research communities the RI aims to serve. This includes understanding the needs and requirements of all related stakeholders and ensure the model is relevant. Areas of improvement from current standards should also be identified.
- **Design Objectives:** Clear and attainable design objectives should be set supporting the needs of the RI. This includes decisions on the structure, openness, flexibility, extensibility, and interoperability of the metadata.
- **Development and Documentation:** The development needs to describe the metadata elements, the rationale behind their inclusion, and their relationships. Documentation should be available in human, and machine, readable and actionable formats.
- **Compliance:** The RI should provide appropriate tools and services to enable users to correctly define metadata values, ensuring compliance and quality.
- **Interoperability:** The RI should provide services to make the metadata interoperable with other standards, such as developing convertors and crosswalks.
- **Evolution:** The metadata model should be periodically reviewed and adapted/enhanced to identify any issues, improve the model's applicability, and ensure long-term and broader acceptance by the research community.

## 4.2. Metadata Registry Service (MRS)

This section presents the Metadata Registry System, which realizes the SFDO metadata model and provides access and management operations for digital objects that adhere to the SFDO structure.

### 4.2.1. Architecture

MRS provides access and management services to SFDOs using three components as illustrated in Figure 7. First, the metadata persisted in a repository, implemented as a Postgres database. Second, a backend is responsible for exposing the repository as a REST API while providing authentication/authorization, backward compatibility, backward maintenance, and other functionality. Finally, a web portal is provided to facilitate human interaction with MRS.

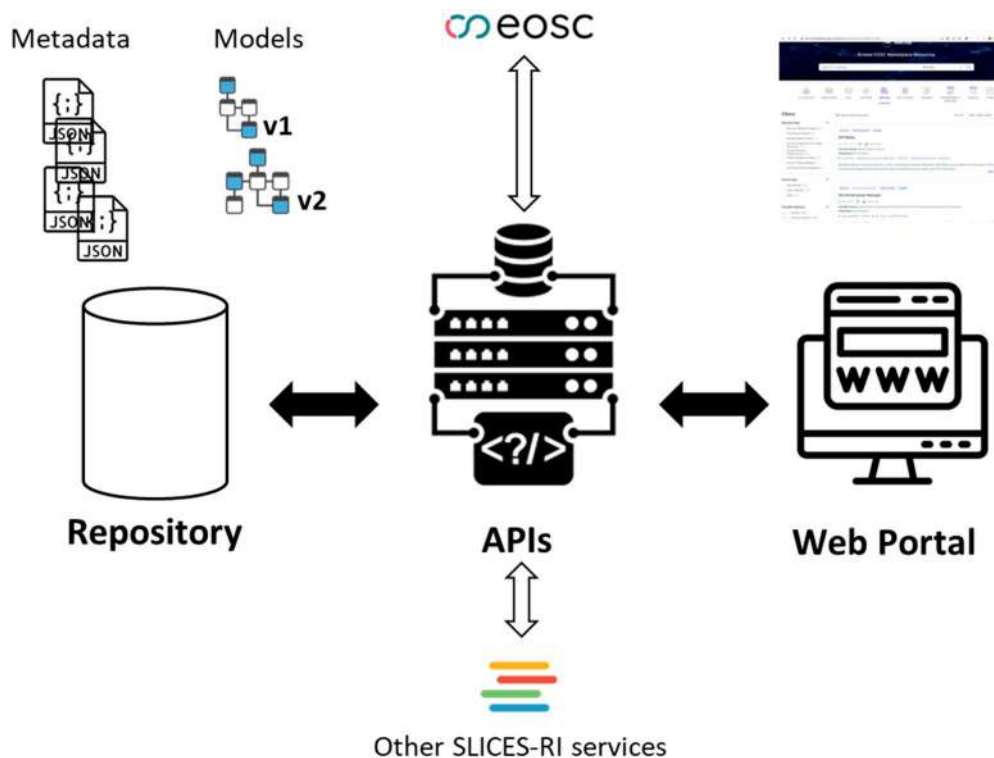


Figure 7. Metadata Registry System (MRS) Architecture.

### 4.2.2. Repository

MRS uses a dedicated repository for storing the metadata models and the digital objects. An important requirement for metadata persistence was to use an open-source solution. PostgreSQL was selected due to its extensive feature set, reliable track record, and widespread familiarity. However, this component can be easily replaced in the future based on new requirements, e.g. scaling, replication, etc. For example, given the nature of MRS





storage and access patterns, the storage could be switched for NoSQL document and graph database engines.

SFDOs are stored using a Table per Hierarchy (TPH) method, i.e. SFDOs of different Level 2 types share the same database table. To accommodate for this, Level 2 attributes (columns) are set as nullable, even if they are required. As the database is only ever accessed by the backend (see next section), the correct schema is still enforced before data has persisted. If the database was accessed directly by multiple systems, check constraints could be used to enforce non-nullability based on the discriminator column.

### **4.2.3. Backend**

The backend is the core of the system. It is implemented using ASP.NET Core Web API, Entity Framework Core and several other widely adopted open-source libraries, such as NSwag (OpenAPI spec generator), Asp.Versioning (for API versions), QueryKit (for the advanced search), lowering the barrier for contribution. SFDO and the attributes are defined as C# classes (an abstract class for Level 1 and one class per Level 2 type, inheriting the L1 abstract class).

The backend exposes the metadata to the rest of the world in the form of a REST API. The REST API is described by an OpenAPI 3.0 specification which is automatically generated from C# class and method definitions. OpenAPI (also known as Swagger previously) is a widely adopted modern standard for API description. Tools exist for consuming OpenAPI-documented REST APIs in nearly all modern programming languages and frameworks, enabling very low-barrier interaction with MRS for any potential consumer.

Besides exposing the metadata to the world, the backend provides additional functionality, such as authentication and authorization. Authentication is done via validating Bearer JWT (JSON Web Tokens) passed as a header in the HTTP request. The JWT is issued and signed by the SLICES Open ID Connect STS (Secure Token Service).

Using this approach, MRS, as a component of SLICES-RI, uses the same AAI as the rest of RI, ensuring a smooth user experience across the entirety of RI services.

At the moment the authorization layer operates in a binary mode - authenticated consumers can perform mutating operations (e.g. create, update, and delete an SFDO); unauthenticated users are not allowed. As the SLICES-PP project progresses and the SLICES AAI (Authentication and Authorization Infrastructure) is developed, the MRS authorization will be inherited from the SLICES AAI. This will provide support for cases such as access control based on the projects that the user is participating in.

All APIs exposed by the backend are versioned to provide backward compatibility. The version is specified as part of the URL path, thus ensuring that any future usage by a specific client



will remain consistent. The backend handles any required translation, allowing the underlying storage format to be decoupled from any client interaction. This allows for scenarios such as:

- New attribute is added. Older clients do not receive this attribute. When older clients create a new SFDO, the new attribute is set to its default value (e.g. null) or a predefined placeholder (e.g. "N/A"). When older clients apply a differential update, the attribute's value is not changed.
- New level 2 type is added. Clients using older versions are either not able to see SFDOs of the new type or a generic "other" type is returned.
- A single value item is promoted to an array. Older clients receive only the 1st element (or null if no elements).

Any change to the metadata is accompanied by an increase in the metadata/API version number. While this approach may appear to add a lot of overhead for MRS maintainers, metadata is not expected to change often; likewise, backward compatibility of a core service is very important to ensuring the longevity of the platform. Currently, the versioning scheme uses "v{major}.{minor}" format. From v1.0 onwards, minor versions are restricted to only adding new attributes and SFDO types.

The API paths for SFDO CRUD (Create, Read, Update, Delete) operations are as follows:

- **Create:** POST /v[apiVersion]/digital-objects, Body: attribute values;
- **Read:** GET /v[apiVersion]/digital-objects/{id};
- **Update:** PUT /v[apiVersion]/digital-objects/{id}, Body: attribute values;
- **Delete:** DELETE /v[apiVersion]/digital-objects/{id}.

For example, to read an SFDO with ID of 13 using metadata version 0.1, the following request can be used: GET /v0.1/digital-objects/13.

While the backend is designed to be as independent as possible, as mentioned above the authentication is delegated to SLICES-global Authentication and Authorization Infrastructure (AAI) to ensure uniform user access across the entire infrastructure.

#### **4.2.4. Web Portal**

The web portal allows user-friendly access to the MRS functionality. Users can search for objects, manage objects (if they have permissions), and access reporting facilities, such as dashboards. The web portal is accessible to all registered users.

The web portal is implemented as a single-page application (SPA) built using Angular 16. It interacts with the backend using the same REST JSON APIs as any other client would. Likewise, the web portal redirects the user to the global SLICES AAI in order to obtain the access token (JWT) to add as a header to the backend API calls.





#### **4.2.5. Metadata Crosswalks**

Metadata transformation enables translation from one metadata format/standard to another when exchanging data. This enables the SLICES metadata format to be transformed, initially to a select set of predominant metadata formats, such as Dublin Core, DataCite, DDI, and ISO19115, to further enhance interoperability. Additionally, it can transform the metadata attributes to specific application profiles that are used by different platforms and services, such as search engines (e.g., Google Scholar) and metadata aggregators. The crosswalks of SFDO were influenced by Metadata Crosswalk mapping model<sup>9</sup>, which provides crosswalks among the most commonly used metadata schemes and guidelines to describe digital objects in Open Science, including RDA metadata IG recommendation of the metadata element set, EOSC Pilot - EDM1 metadata set, Dublin CORE Metadata Terms, Datacite 4.3 metadata schema, DCAT 2.0 metadata schema and DCAT 2.0 application profile, EUDAT B2Find metadata recommendation, OpenAIRE Guidelines for Data Archives and many more.

---

<sup>9</sup> Crosswalk of most used metadata schemes and guidelines for metadata interoperability, <https://fairsharing.org/CrosswalkOfMostUsedMetadataSchemesAndGuidelines>



## 5. SLICES Services for Interoperability and Integration with EOSC

---

### 5.1. Using RO-Crate for Research Data Archiving and Sharing

RO-Crate (Research Object Crate) is a community-driven effort to create a lightweight approach to packaging research data with rich metadata<sup>10</sup>. It is designed to facilitate the archiving, sharing, and reuse of research data by providing a standardized, extensible format for describing datasets, workflows, and other research outputs. RO-Crate consists of three main components : Research Object (RO), Crate and metadata.

A Research Object (RO) within a crate aggregates diverse digital assets, including datasets, documents, software, and workflows, and is described using JSON-LD (JavaScript Object Notation for Linked Data) to ensure compatibility with web standards. The process involves identifying the research data to be included, collecting or creating comprehensive metadata that describes the data and provides contextual and provenance information, and organizing the data into a structured directory with a metadata file, typically named `ro-crate-metadata.json`. Tools provided by the RO-Crate community can be used to validate the metadata file to ensure it meets the specifications and guidelines. Once validated, the RO-Crate can be shared through repositories, institutional archives, or data sharing platforms, making it accessible and citable. The rich metadata enhances discoverability, enabling other researchers to understand, cite, and reuse the data easily. RO-Crate supports interoperability, standardization, and sustainability in research data management, ensuring that research outputs are well-documented and preserved for long-term use, thereby enhancing the overall quality and impact of scientific research.

### 5.2. Using Data Version Control for Experimental Data Management

Data version control (DVC) is a critical practice in the realm of experimental data management, offering robust solutions for tracking changes, ensuring reproducibility, and managing collaboration<sup>11</sup>. In scientific research and data-intensive projects, the ability to track the evolution of datasets and code is essential. DVC provides tools and methodologies that enable researchers to version control their data in much the same way they version control their code. By creating a system where every change to the dataset is logged and can be reverted, DVC helps maintain the integrity and reliability of experimental results.

---

<sup>10</sup> Information on RO-Crate: [https://www.researchobject.org/ro-crate/about\\_ro\\_crate](https://www.researchobject.org/ro-crate/about_ro_crate) (accessed 29 July 2024)

<sup>11</sup> Information on DVC: <https://dvc.org/doc> (accessed 29 July 2024)

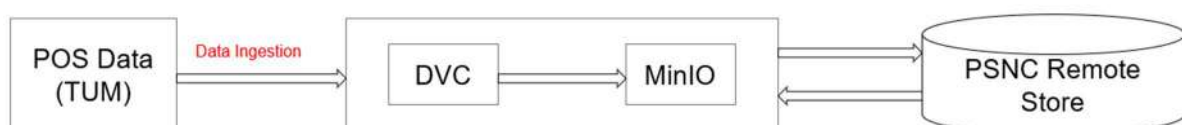
One of the primary benefits of DVC is its support for reproducibility. Scientific experiments often require detailed records of how data was processed and analysed. With DVC, every modification to the data and its associated processing scripts are versioned. This means that any collaborator can recreate the exact environment and data state that were used in a previous analysis, ensuring that results can be consistently replicated. This is particularly important in complex experiments where even minor changes can significantly impact outcomes.

By providing a clear and comprehensive history of data and its transformations, DVC ensures that all steps in the experimental process are transparent and reproducible.

DVC also enhances collaboration among researchers. In multi-person projects, coordinating changes to datasets can be challenging without a structured system. DVC integrates with traditional version control systems like Git, allowing teams to manage data alongside their code within a unified framework. This integration facilitates seamless collaboration, as team members can pull the latest data changes, push their modifications, and resolve conflicts efficiently. Additionally, DVC supports remote storage options, enabling teams to share large datasets without the limitations of traditional version control systems.

Furthermore, DVC addresses the challenges of managing large datasets. Traditional version control systems are not optimized for handling big data, but DVC uses a smart approach by storing only the changes (deltas) rather than entire datasets. This significantly reduces storage requirements and improves performance. DVC's ability to work with cloud storage solutions also means that researchers can leverage scalable infrastructure for their data needs, ensuring that storage and accessibility issues do not hinder the progress of their projects.

For the purpose of testing DVC with POS data, a MinIO instance with S3 object storage was created. The remote storage is provided by PSNC. The general objective of the test was to verify the functionality and ease of implementation with integrating DVC on POS data. The architecture of the integration of POS data with data versioning and MinIO storage is presented below in Figure 8.



**Figure 8. Experimental data management architecture.**



DVC allows seamless and easy integration like with any cloud storage service. It has support for S3 object storage for AWS and AWS like S3 object storage as MinIO.

It can be integrated by using simple commands on the terminal once you have installed DVC on your system. The commands are Git-like so anyone having knowledge of Git can use DVC following instructions and demos available on their website or on the web. The commands for integrating and DVC and MinIO can be done as:

```
dvc remote add -d minio s3://<bucket-name>/<project-name> \  
  --endpoint <minio-endpoint-url> \  
  --access-key <minio-access-key> \  
  --secret-key <minio-secret-key>
```

The test was conducted to verify data reproducibility, efficiency in storage and for data versioning and lineage tracking. The scalability to include large volume of data and performance were also verified.

### 5.2.1 Setting up Git and initializing DVC

**Step 1:** Create a code repository in [github](#), [gitlab](#) or any specific hosting platform.

**Step 2:** In your preferred python environment, install dvc with s3 object storage dependencies.

```
$ pip install dvc[s3]
```

**Step 3:** Cloning your directory and locating yourself into it

```
$ git clone <your working directory>
```

```
$ cd <directory>
```

**Step 4:** Initializing data version control

```
$ dvc init
```



```
PS C:\Users\sshrest\Documents\slicesmd> dvc init
Initialized DVC repository.

You can now commit the changes to git.

-----
DVC has enabled anonymous aggregate usage analytics.
Read the analytics documentation (and how to opt-out) here:
<https://dvc.org/doc/user-guide/analytics>
-----

What's next?
-----
- Check out the documentation: <https://dvc.org/doc>
- Get help and share ideas: <https://dvc.org/chat>
- Star us on GitHub: <https://github.com/iterative/dvc>
```

A message on the terminal can be seen as above if the dvc is setup properly.

**Step 5:** Commit the files created by dvc init

```
$ git add .
$ git commit -m "initialized dvc"
```

All the necessary config files are created after initializing dvc, so it is important to add the files using git commands.

## 5.2.2 Setting up DVC remote storage

After connecting to the MinIO server a S3 bucket was created. It is important to note that minio server uses the port 9001 and the S3 API uses the port 9000. The object storage can be used as remote store using following steps:

**Step 1:** Adding the storage bucket

```
$ dvc remote add -d s3-data-storage s3://slicesexp/
Setting 's3-data-storage' as a default remote.
```

**Step 2:** Adding the endpoint url

```
$ dvc remote modify s3-data-storage endpointurl http://127.0.0.1:9000
```

**Step 3:** Configuring and adding remote access credentials

```
$ dvc remote modify --local s3-data-storage access_key_id <your_username>
$ dvc remote modify --local s3-data-storage secret_access_key <your_password>
```

**Step 4:** Committing the files created by dvc init



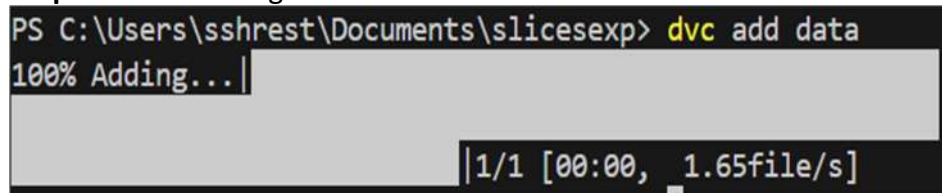


```
$ git add .  
$ git commit -m "setup of dvc remote storage"  
[main 4acd2e0] setup of dvc remote storage  
1 file changed, 5 insertions(+)
```

### 5.4.3 Adding the data to dvc tracking

After successful configuration of remote storage, the next step is to add the data which needs to version controlled and tracked for the purpose of reproducibility. This can be done in following steps:

**Step 1:** Start tracking the desired data



```
PS C:\Users\sshrest\Documents\slicesexp> dvc add data  
100% Adding... |  
|1/1 [00:00, 1.65file/s]
```

DVC will guide you through how to add tracking of files in Git. Instead of adding the whole **data** directory to the repository, you only add the **data.dvc** file. You can check that **data** directory has been added to **.gitignore**.

**Step 2:** Add the changed files not tracked by dvc

```
$ git add .  
$ git status  
On branch main  
Your branch is ahead of 'origin/main' by 2 commits.  
(use "git push" to publish your local commits)
```

```
Changes to be committed:  
(use "git restore --staged <file>..." to unstage)  
    modified:   .gitignore  
    new file:   data.dvc
```

**Step 3:** Committing the changes

```
$ git commit -m "Control the data directory with DVC"  
[main 5edab45] Control the data directory with DVC  
5 files changed, 109 insertions(+)  
create mode 100644 data.dvc
```

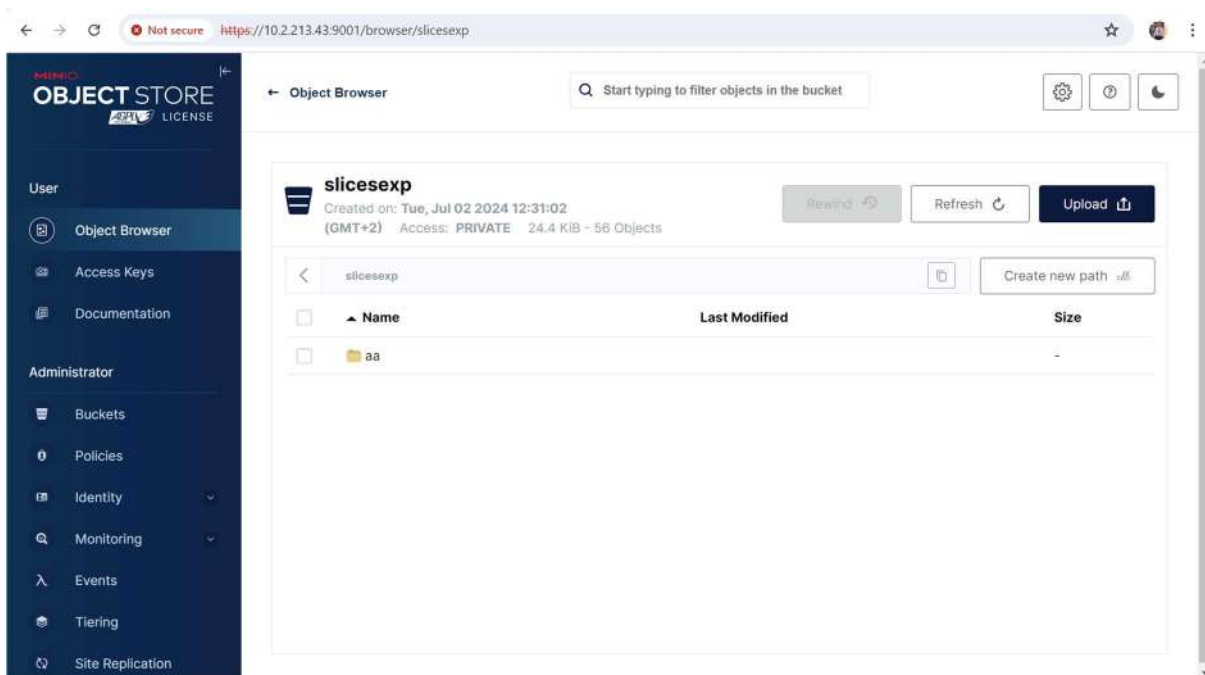


**Step 4:** Pushing the data to git and dvc

```
$ git push
```

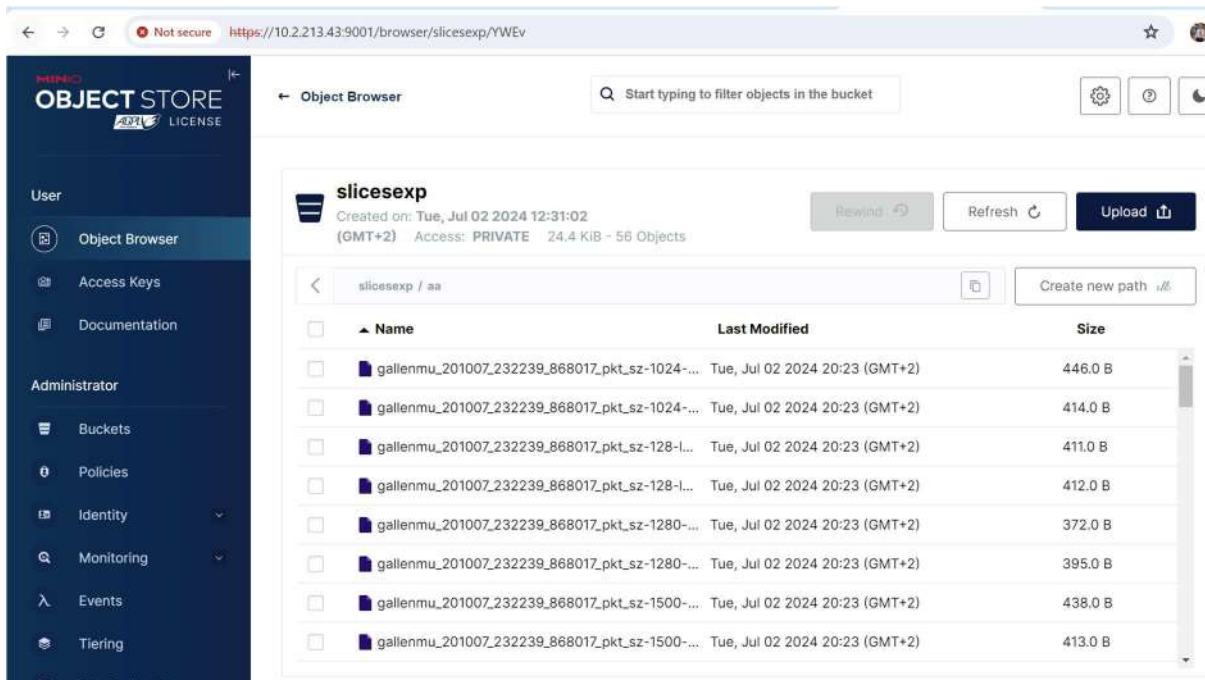
Then

```
$ dvc push
```



After pushing we can verify on the minio server that some new objects are stored in the corresponding bucket.





Also, we can verify on our github repository that we have only pushed code files and .dvc.







shashankshres - Added remote storage,-Added data		3876223 · last month	🕒 6 Commits
📁 .dvc	-Added remote storage,-Added data		last month
📁 _includes	added all data		last month
📁 experiment	added all data		last month
📁 figures	added all data		last month
📁 plot_scripts	added all data		last month
📁 results	added all data		last month
📁 template	added all data		last month
📁 web	added all data		last month
📄 .dvcignore	initialize dvc		last month
📄 .gitignore	-Added remote storage,-Added data		last month
📄 README.md	first commit		last month
📄 _config.yml	added all data		last month
📄 data.dvc	-Added remote storage,-Added data		last month
📄 index.html	added all data		last month
📄 publish.py	added all data		last month

For the next steps on this experiment, we will focus on the reproducibility aspects and also analyze in more details the overall data versioning and lineage tracking for integration into the DMI and also other pre-op blueprint services.





## 6. Conclusion

---

This report provides a summary on the ongoing developments, design and research ideas related to the interoperability and integration of SLICES with EOSC. Current developments on the metadata extraction, data sharing and archiving, through MRS is presented. The interoperability framework for integrating SLICES with EOSC is discussed. Also, the ongoing research and development focusing on the principles of data management are highlighted.

For the pre-op phase of the project, a mini-DMI was created supporting metadata registry. In future work, the developments and ideas will be integrated with other services discussed in the pre-op phase. These works will be done as a contribution for WP7. An extensive study and development on the modelling of the data for 5G blueprint will be conducted and presented as future deliverable. This is required for the better understanding of the metadata and data management requirements. Also due to changes in the standards of EOSC, the interoperability and integration methods may have to be revised. Adhering to the changes and research work, WP7 will contribute to the ongoing developments for an operational DMI in the near future.



## 7. References

---

- [1] "SLICES-DS Deliverable D4.2: SLICES infrastructure and services integration with EOSC and Open Science (initial proposal)".
- [2] "SLICES-DS Deliverable D4.3: Definition of the SLICES metadata profiles to support FAIR principles (initial proposal)".
- [3] "SLICES-DS Deliverable D4.5: SLICES infrastructure and services integration with EOSC, Open Science and FAIR: Recommendations and design patterns (final report)".
- [4] "European Interoperability Framework, European Union," 2017. [Online]. Available: [https://ec.europa.eu/isa2/sites/default/files/eif\\_brochure\\_final.pdf](https://ec.europa.eu/isa2/sites/default/files/eif_brochure_final.pdf).
- [5] "EOSC Interoperability Framework, EOSC Executive Board," February 2021.
- [6] "EOSC Architecture and interoperability Framework. EOSC Association.," Dec 2021. [Online]. Available: <https://eosc-portal.eu/sites/default/files/EOSC%20Future-WP3-EOSC%20Architecture%20and%20Interoperability%20Framework-2021-12-22%5B17%5D%5B6%5D-2.pdf>.
- [7] "EOSC Interop, EOSC Future WG," [Online]. Available: <https://eoscfuture.eu/eosc-future-working-groups/>
- [8] "Opinion paper on advanced digitalisation of research, An Opinion paper by the ESFRI-EOSC Task Force and Steering Board Expert Group, March 2024," [Online]. Available: <https://zenodo.org/records/10980285>.
- [9] Y. Demchenko, Sebastian Gallenmueller, Serge Fdida, Panayiotis Andreou, Damien Saucez and Thijs Rausch, "SLICES Data Management Infrastructure for Reproducible Experimental Research on Digital Technologies,," in *IEEE Global Communications Conference, 4–8 December 2023, Kuala Lumpur, Malaysia*.
- [10] Y. Demchenko, Cees de Laat and Wouter Los, "Future Scientific Data Infrastructure: Towards Platform Research Infrastructure as a Service (PRlaaS)," in *The International Conference on High Performance Computing and Simulation (HPCS 2020), 10-14 Dec 2020*.
- [11] Y. Demchenko, S. Gallenmüller, S. Fdida, P. Andreou, C. Crettaz and M. Kirkeng, "Experimental Research Reproducibility and Experiment Workflow Management," *2023 15th International Conference on COMMunication Systems & NETWORKS (COMSNETS)*, Bangalore, India, 2023, pp. 835-840, doi: 10.1109/COMSNETS56262.2023.10041378.
- [12] Fdida, S., Makris, N., Korakis, T., Bruno, R., Passarella, A., Andreou, P., ... & Knopp, R. (2022). SLICES, a scientific instrument for the networking community. *Computer Communications*, 193, 189-203.



[13]

Rakotoarivelo, T., Ott, M., Jourjon, G., & Seskar, I. (2010). OMF: a control and management framework for networking testbeds. *ACM SIGOPS Operating Systems Review*, 43(4), 54-59.



## Appendix A. FAIR Data principles – This text is taken from the SLICES DMP

---

### A.1. Making data Findable

Non confidential data of the project, such as deliverables will be published in the Zenodo certified digital repository. The repository provides structured metadata (e.g., JSON, Dublin Core) with transformation capabilities (e.g., Zenodo's JSON to DublinCore) supporting easy discovery (e.g., using keywords). Each dataset in the repository will have a unique and persistent identifier. The selected repository is designed for long-term data preservation and availability. Furthermore, a SLICES-PP community space will be set up in the Zenodo for project reports and software code. Software code will also be published through GitHub accompanied by appropriate documentation.

Naming consistency is important for efficiently locating a resource and understanding its use. However, it is up to the creator to provide proper names for research outputs and related data files. SLICES adopts certain Naming Conventions/Guidelines to improve the structure/consistency of files. The draft guidelines include the following recommendations:

- **File Naming:** A maximum length of 260 characters will be used for all file names, as long filenames may not be interoperable with some systems. Additionally, to further improve interoperability, the file names will not be exactly the same as keywords (e.g., while) as these may be interpreted as commands by some systems. Furthermore, the following characters will be avoided in the file names.
  - o < (less than)
  - o > (greater than)
  - o : (colon)
  - o " (double quote)
  - o / (forward slash)
  - o \ (backslash)
  - o | (vertical bar or pipe)
  - o ? (question mark)
  - o \* (asterisk)
- **Date Format:** The YYYYMMDD format will be used to allow for display of dates in a chronological order, even over the span of many years.
- **Leading Zeros:** Use leading zeros to make an ascending order of numbers correspond to alphabetical order.
- **Naming Scheme:** Use a consistent naming scheme throughout; do not use spaces or punctuation symbols as these may not be interoperable with some systems. Order / confirm which element should go first, so that files on the same theme are listed together and can be found easily. Project deliverables based on (and referring to) files,



as well as other documentation (see below) will provide more context information.

File organisation is important for efficiently locating a resource, even in cases where there is no predefined structure available. SLICES utilizes certain guidelines to improve the consistency of the structure of the data. The initial guidelines include the following recommendations:

- **Hierarchical File Structure:** each file will be placed in an appropriate folder according to the work package and category/task it belongs to (e.g., WP7 – Data Management /Metadata/Experiment).
- **Dissemination:** will store all material that can be utilized for communication and dissemination activities, such as templates (e.g., presentation, deliverable) and articles. Material concerning dissemination will be organized in appropriate folders and placed under “WP8 - Communication, dissemination and exploitation” in MyBox.

## A.2. Making Data Accessible

The SLICES-PP grant agreement (Section Open Science) states that publications stemming from the project will be published in Open Research Europe that has been launched by EC in March 2021 for H2020 and HE beneficiaries. Open Research Europe is a recommended venue for publishing results of the European projects/grants, and it ensures automatic compliance with HE policy.

The data produced will be made openly available (by default) and will be disseminated in various ways. In particular:

- **Survey/Interview data**, including documentation such as questionnaires and codebooks, that do not contain personal data or have been anonymized, may be made available via certified repositories and/or via the SLICES-PP website. For non-anonymized data, appropriate consents will be drafted to assure the rights of the data subjects. Survey/interview data concerning interviewees that do not provide consent to share, object to processing of their data, or withdraw their consent at any point, will not be shared. Data that will be published will be licenced by a CC BY-SA licence. Zenodo uses JSON Schema as internal representation of metadata and offers export to other popular formats such as Dublin Core or MARCXML, transforming the metadata to appropriate vocabularies.
- **Software** tools will be published in the GitHub repository and will be accompanied by an appropriate license, e.g. GPL-3.0, MIT. Publishing software will be accomplished using the standard upload connection from GitHub to Zenodo, where a persistent identifier is assigned. Database files will also be accompanied by an ODC-By license.
- **Scientific publications** that may arise from the project results will be published in open access venues and shared through Zenodo.

## A.3. Making Data Interoperable



Data stored in the repositories mentioned in the previous Section utilize file formats that are inherently open and allow for straightforward reuse. The repositories support export to established standards, such as Dublin Core and DataCite, ensuring wide interoperability.

The project will also design appropriate metadata profiles as part of the data governance/management framework of the future SLICES-RI to ensure the full support of FAIR principles utilizing machine-readable metadata attributes to allow for easy discovery of data managed and used within the RI by both humans and computers. Additionally, within the framework, specially catered metadata will be used to improve machine and human-understandability as well as machine-actionability, allowing services to access information and understand complex and domain-specific metadata structures to take appropriate actions. The metadata profiles will be described in the final version of the DMP.

Furthermore, the project will develop transformation mechanisms allowing for the internal metadata formats to be transformed into various metadata profiles, for both import of external data and export of internal data, facilitating re-combinations with different datasets from different origins, and enabling seamless data exchange and re-use between researchers, institutions, organisations, countries. The metadata will be accompanied by their definitions and appropriate vocabularies/controlled lists for interoperability but also other actions, such as input validation.

Finally, interoperability is one of the goals of WP7 (“Data Management and ethics requirements”), which will actively participate in domain-specific and cross-disciplinary initiatives involved in semantic interoperability. Additionally, T7.3 will provide the specifications of the mechanisms required to integrate with the European Open Science Cloud.

#### **A.4. Making Data Reusable**

Data stored in the certified repositories will include appropriate and accurate metadata with relevant attributes to ensure reusability. For example, each Zenodo record, contains a minimum of DataCite's mandatory terms, with optionally additional DataCite recommended terms and Zenodo's enrichments. Furthermore, Zenodo requires a license as part of the metadata of each digital object, ensuring that users accessing the digital object are subject to the license specified in the metadata by the uploader. The data uploaded within the project will be traceable to the project's community space and the users that uploaded the data. Where applies, the project consortium may provide additional domain-specific information to specific data to make it more broadly accessible.

The project will also design a set of Data Quality Management (DQM) tools to ensure accuracy, consistency and interpretability of the data within the future SLICES-RI, addressing also factors such as completeness, timeliness and believability which cannot be tackled directly through the tools, but they can be “inferred” partly from other measures. SLICES envisions that DQM tools will fall into five different categories: (i) Data Cleaning; (ii) Data Integration; (iii) Data Reduction; (iv) Data Transformation; and (v) Data Interpretation.



To the extent that data form the basis of project deliverables, internal quality review procedures will apply (e.g., project deliverables will be assigned two reviewers). These are described in the SLICES-PP D9.1: Project Quality plan and detailed work plan.

Licensing issues will be addressed as per instructions of Article 16 of the Grant Agreement. Data and code will be made available through certified digital repositories that support the relevant types of licences, and that will be able to preserve the data for the long term (in principle “indefinitely”). We will set up a SLICES-PP community space in the Zenodo archive for the project results; at the end of the project, this community space could be handed over to an EOSC organisation for future extension and maintenance, ownership arrangements permitting.





