D2.5 EOSC Compute Platform Handbook

Lead partner:	EGI Foundation
Version:	1
Status:	Final
Dissemination Level:	PUBLIC
Keywords:	EOSC, Compute, Handbook, ecp
Document Link:	https://documents.egi.eu/document/3813

Deliverable Abstract

The EOSC Compute Platform Handbook is **written for EOSC users** to provide guidance on how to use the various services of the EOSC Compute Platform. The EOSC Compute Platform (ECP) is a system of federated compute and storage facilities together with higher level platforms to support various data and application hosting, data processing and analysis use cases.

EGI-ACE receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 101017567. Visit go.egi.eu/egi-ace for details.

EGI-ACE receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 101017567.

DELIVERY SLIP

Date	Name	Partner/Activity
From:	Giuseppe La Rocca	EGI Foundation / WP2
Moderated by:	Sjomara Specht	EGI Foundation/ WP1
Reviewed by:	Isabel Campos Anastas Mishev Ignacio Lamata Martínez	CSIC UKIM EGI Foundation
Approved by:	SDS	

DOCUMENT LOG

Issue	Date	Comment	Author
v.0.1	11.02.2022	Full draft for external review	G.Sipos, G. La Rocca, E. Fernández, L.Farkas, A. Manzi, A. Cristofori.
v.1	27.02.2022	Final	G. La Rocca

TERMINOLOGY

Terms used in this document are defined at https://confluence.egi.eu/display/EGIG.

Table of Contents

Executive summary

EGI-ACE is a 30-month project coordinated by the EGI Foundation with a mission to empower researchers from all disciplines to collaborate in data- and compute-intensive research through free-at-point-of-use services. Building on the distributed computing integration in the EOSC-hub project, EGI-ACE will deliver the EOSC Compute Platform (ECP) and will contribute to the EOSC Data Commons through a federation of cloud compute and storage facilities, PaaS services and data spaces with analytics tools and federated access services.

This handbook is instrumental in providing guidance and insights on how users from different scientific disciplines can use the services offered by the EOSC Compute Platform for supporting data- and compute-intensive research. After a brief overview about the EOSC Compute Platform, the handbook provides a deep view on the Platform services which are grouped into three main categories: Compute, Data and Advanced services. In Section 4, 5 and 6 the users will learn how to work data, using the computing solutions, and more, as offered by the EOSC Compute Platform.

1. Introduction

This handbook is a user manual on the EOSC Compute Platform and serves as a comprehensive introduction to the services provided by the Platform. The EOSC Compute Platform is a system of federated compute and storage facilities together with higher level platforms to support various data and application hosting, data processing and analysis use cases.

The EOSC Compute Platform is implemented as integrated compute and storage services, platforms, data spaces and tools that are contributed by leading compute centers, Universities, Research Institutions and Infrastructures, and private companies. For end-users, this Platform appears as an open, data-centric, distributed, hybrid and secure infrastructure, consisting of a growing number of cloud, High-Throughput Compute (HTC) and High-Performance Compute (HPC) providers. This Platform will be accessible with a mix of funding mechanisms through interfaces that leverage standard-compliant solutions for federated authentication, authorization and compute management at nationally provided, distributed compute and storage facilities.

The rest of this section is organised as follows:

- A brief introduction of the European Open Science Cloud (EOSC), the umbrella initiative of the Compute Platform, is provided in Section 1.1.
- A high-level overview of the services of the EOSC Compute Platform is described in Section 1.2.
- The implementation of the FAIR principles (Findable, Accessible, Interoperable, and Re-usable) in the EOSC Platform is discussed in Section 1.3.
- Data privacy considerations and its policies are outlined in Section 1.4.
- The next steps to be followed in this Handbook are presented in Section 1.5.

1.1. What is EOSC?

The European Open Science Cloud (EOSC) is an initiative of the European Commission (EC) and its Member States with the ambition to develop a ‘Web of FAIR Data and services’ for science in Europe. EOSC will be a multi-disciplinary environment where researchers can publish, find and re-use data, tools and services, enabling them to conduct their work more efficiently.

EOSC builds on existing infrastructure and services supported by the EC, Member States and research communities. It brings these together in a federated ‘system of systems’ approach, adding value by aggregating content and enabling joint use of services.

EOSC aims to become Europe’s trusted virtual environment for 1.7 million European researchers and 70 million professionals in science, technology, humanities and social sciences, in order to accelerate the creation of new knowledge, inspire education, stimulate innovation and promote accessibility and transparency.

The vision of EOSC was conceived in 2015 and officially launched in November 2018, with access to an initial set of services through the **EOSC Portal**[1]. In recent years, the EC provided financial support to implement EOSC by means of projects under the *EU Framework Program for Research and Innovation* (Horizon 2020 and Horizon Europe), by means of Working Groups (in 2019-2020)[2], and most recently in the form of Advisory Groups[3] (from 2021).

The **EOSC Association**[4] was established in Brussels in July 2020, to act as the focal point for the EOSC Partnership. It was founded as a Belgian non-profit Association (AISBL) and is open to interested stakeholder organisations that meet the published Rules of Participation.

In June 2021, the European Commission and the EOSC Association launched a Co-programmed European Partnership[5] until 2030, aiming to strengthen EOSC with European funding of almost €500 million, and an in-kind contribution from partners of an additional €500 million. The main high-level General Objectives (GOs) of the Co-programmed European Partnership are defined as follows:

- GO1: Ensure that Open Science practices and skills are rewarded and taught, becoming the “new normal”.
- GO2: Enable the definition of standards, and the development of tools and services, to allow researchers to find, access, reuse and combine results.
- GO3: Establish a suitable and federated infrastructure enabling open sharing of scientific results.

1.2. What is the EOSC Compute Platform?

The **EOSC Compute Platform** is a system of federated hybrid compute and storage facilities, research data hosting, processing and analytics tools, and a set of complementary services for distributed data and computing, to support processing and analytics for distributed data and computing use cases for Open Science. The Platform is built on the EGI Federation, and it is offered by the *EGI-ACE Project*[6]. The Platform federates the capacity of some of Europe’s largest research data centers and offers heterogeneous systems to meet different research needs, including state-of-the-art GPGPUs[7] and accelerators supporting Artificial Intelligence (AI) and Machine Learning (ML), which makes the Platform an ideal innovation space for AI/ML applications. The Platform is offered with different costs and funding models to ensure that the diverse scientific communities can meet their needs and benefit from the investment from the European Commission and the Member States.

[1] https://www.eosc-portal.eu/
[2] https://www.eoscsecretariat.eu/eosc-working-groups
[3] https://www.eosc.eu/advisory-groups
[4] https://eosc.eu/
[5] See MoU: https://www.eosc.eu/sites/default/files/EOSC_Memorandum_30_July_2021.pdf
[6] https://www.egi.eu/projects/egi-ace/
[7] General-Purpose computing on Graphics Processing Units

The technical architecture of the EOSC Compute Platform is multi-layered. (Fig. 1)

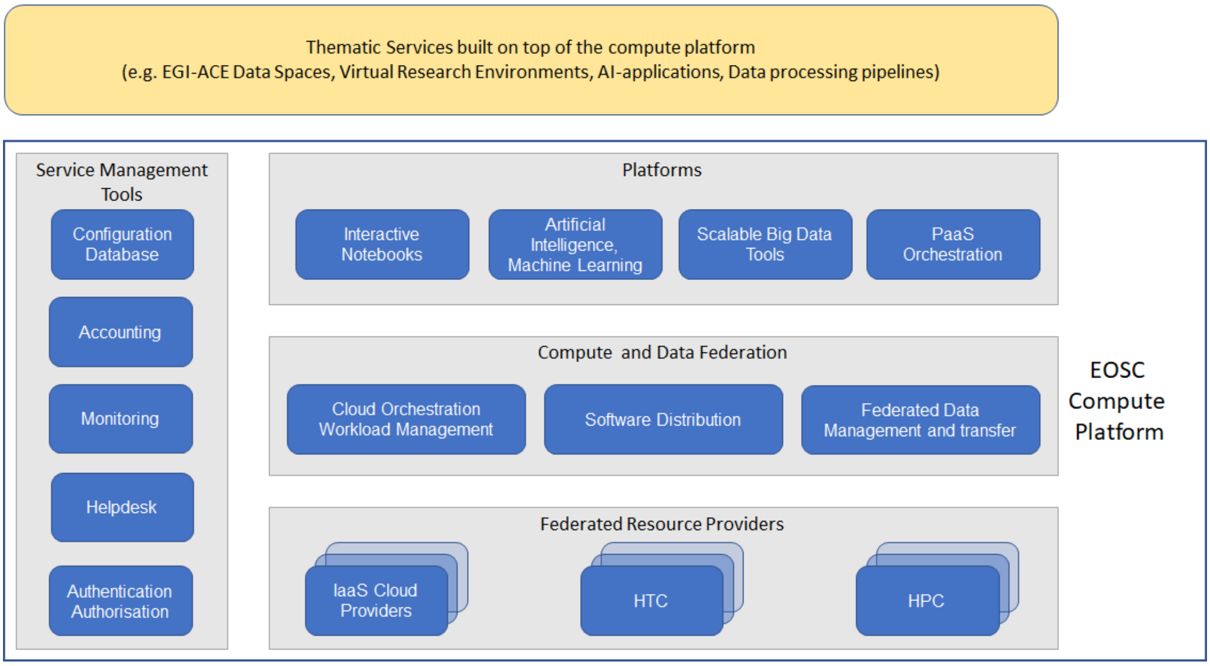

Thematic Services built on top of the compute platform
(e.g. EGI-ACE Data Spaces, Virtual Research Environments, AI-applications, Data processing pipelines)
Service Management Tools
Configuration Database
Accounting
Monitoring
Helpdesk
Authentication Authorisation
Platforms
Interactive Notebooks
Artificial Intelligence, Machine Learning
Scalable Big Data Tools
PaaS Orchestration
Compute and Data Federation
Cloud Orchestration Workload Management
Software Distribution
Federated Data Management and transfer
Federated Resource Providers
IaaS Cloud Providers
HTC
HPC
EOSC Compute Platform

Figure 1 - The EOSC Compute Platform architecture

- The **Federated Resource Providers** layer provides an *Infrastructure-as-a-Service*[8] (IaaS) hybrid infrastructure from academic and commercial providers to run and/or host research applications and data. The IaaS component of the EOSC Compute Platform are collaboratively provided by High Throughput Computing (HTC, described in Section 5.4), High Performance Computing (HPC, described in Section 5.5) and research cloud providers that are members of the EGI Federation from the following regions: Central Europe (Czech Republic, Germany, Poland, Slovakia), South Eastern Europe (Bulgaria, Romania, Turkey), Southern Europe (Italy, Portugal, and Spain), and Western Europe (France and The Netherlands). Through the Virtual Access (VA) funding mechanism, the capacity of the Federated Resource Tier is pre-paid by the EC and offered to researchers and projects of publicly funded environments.
- The **Compute and Data Federation** layer provides advanced solutions to facilitate common, recurring tasks in the hybrid IaaS federation, such as software distribution, virtual infrastructure orchestration and data management (*e.g.* transfer, replication, caching, *etc*).
- The **Platforms** layer extends the IaaS delivery model with high-level abstractions that simplify the execution of researchers' workloads. It offers generic *Platform-as-a-Service* (PaaS)[9] services that exploit the compute and storage resources of the lower level layers to create value-added services, such as development environments (see Section 6.2), Machine Learning and Deep Learning applications for data analysis

[8] https://en.wikipedia.org/wiki/Infrastructure_as_a_service
[9] https://en.wikipedia.org/wiki/Platform_as_a_service

(see Section 6.1), scalable Big Data tools, and workload management tools (see Section 5.6).

- The **Service Management Tools** enable user access to the EOSC Compute Platform (via authentication and authorisation mechanisms) and integrate the Compute Platform services with the rest of EOSC components. This layer includes Helpdesk, Monitoring, Accounting, Operations Management, Security and Incident response, and Software Quality Assurance.

On top of the Compute Platform are the Thematic Services, which are scientific domain-specific services and tools that provide simulation, machine learning and data analytics capabilities tailored to the needs of the specific research areas.

1.3. How the Compute Platform supports FAIR Principles

The European Commission aims to make science open and accessible to everyone, by enabling open access to research data, facilitating data sharing and reuse, and establishing mechanisms for interdisciplinary joint scientific collaborations. This section describes how the EOSC Compute Platform encourages the implementation of best practices according to the Findable, Accessible, Interoperable, and Re-usable (FAIR)[10] principles, to achieve the implementation of distributed and federated virtual environments to store, manage, and share research data.

1.3.1. Findable

The Compute and Data Federation services layer depicted in *Figure 1* exposes discoverable datasets and staging data inside and outside the EOSC Compute Platform Cloud. Services such as *Rucio* and *DataHub* provide abstraction layers to discover and access datasets stored in repositories with different technology in a scalable way. These solutions are used by large scientific communities, such as *SeaDataNet*[11], whose international oceanography community used DataHub to provide unified discovery and access to the large volumes of distributed marine and ocean data sets managed by marine data centers. Additional details about the Data Federation services can be found in Section 4.

The discovery of domain-specific software and pre-configured Virtual Machine images (e.g. Virtual Appliances) is made possible by the *Application Database*[12] (AppDB) service of the Platform. Researchers in any community can easily discover and reuse software and Virtual Appliances that have been registered in AppDB and are offered natively.

In the long-term, the guidelines to address the Findability of Open Science will be further extended by the PID Policy and Implementation Task Force (TF)[13] of the Implementation of EOSC Advisory Group. This TF is preparing recommendations to bridge the gaps raised in

[10] https://ec.europa.eu/research/openscience/index.cfm?pg=open-science-cloud
[11] https://www.seadatanet.org/
[12] https://appdb.egi.eu/
[13] https://www.eosc.eu/advisory-groups/pid-policy-implementation

the SRIA document, promote the use of mature and recognised PID infrastructures, standardize the use of PIDs graphs, and integrate PIDs into FAIR Data Management.

1.3.2. Accessible

Access to the EOSC Compute Platform resources, such as datasets and services, are protected by strong security mechanisms. This aspect of the FAIR principle is an important requirement both for providers operating services in the Platform, who want to control who can access their resources, and for researchers in scientific communities, who want to access and use the resources offered by the Platform in a uniform and easy manner. The Platform provides a high-level authentication and authorisation mechanism via the EGI *Check-in*[14] service. Check-in is part of EGI's Authentication and Authorisation Infrastructure (AAI) and operates as a proxy service that connects federated *Identity Providers* (IdPs) with the *Service Providers* (SP) registered in the Platform. Access to datasets and applications hosted on the Compute Platform can be defined through Check-in, ensuring user identity (authentication) and enabling resource access based on a highly configurable group membership (authorisation). To guarantee the Access aspect of the FAIR principle the Platform promotes the adoption of standard protocols to facilitate the data retrieval and the use of metadata to reference datasets even when the data are no longer available. As an example to demonstrate how the Platform contributed to support accessibility (A in FAIR) to data objects we highlight the collaboration started between ECRIN (the European Clinical Research Infrastructure Network)[15], OneData and INFN[16]. During this collaboration the EGI DataHub service was adopted to harvest metadata about the data objects, including object provenance, location and access details, from a variety of source systems and aggregate it in the ECRIN Clinical Research Metadata Repository (MDR)[17]. For further details, please refer to the official website[18].

1.3.3. Interoperable

Present and future services of the EOSC Compute Platform are developed in accordance with EOSC security policies and the interoperability guidelines for distributed computing and authentication-authorisation. The adoption of these interoperability guidelines, which encourage the development of modular services and the adoption of standard interfaces, allows domain-specific Thematic services to combine mature software tools, datasets and services in order to address the needs of the different scientific communities.

Services of the Platform generally attempt to provide access through standard *REST* Application Programming Interfaces (API), which uses *HTTP* as the network protocol and provide *OpenAPI* compliant specifications[19] to facilitate interoperability and composability of services[20].

[14] https://www.egi.eu/services/check-in/
[15] https://ecrin.org/
[16] https://www.ba.infn.it/index.php/en/
[17] https://ecrin.org/clinical-research-metadata-repository
[18] http://ecrin-mdr.online/index.php/Project_Overview
[19] https://swagger.io/specification/
[20] In computing, service composability is a design principle that encourages the design of services that can be reused in multiple solutions that are themselves made up of composed services.

The *Workload Management Service*, based on the *DIRAC*[21] technology, represents one of the high-level solutions to ensure interoperability with the distributed heterogeneous resources. As an example, this service is currently used by the *WeNMR*[22] structural biology community to run millions of jobs that predict, model, and dissect biomolecular interactions at the atomic level.

1.3.4. Re-Usable

Although reproducibility lies at the heart of science it still remains one of the greatest challenges. From a technical perspective, the reproducibility of research outcome requires a detailed description of the computational environment used to conduct a study, including the data, software and dependencies used, together with documentation describing how all the different pieces fit together and contribute to producing the scientific result.

To enhance reproducibility and facilitate reproducible research, the Platform offers an interactive, Web-based piece of software called *Jupyter Notebooks*[23] (described in Section 6.2.1) complemented with *Binder*[24] (described in Section 6.2.2). Jupyter Notebooks allows users to create data and code-driven narratives made of interactively executable code, equations, descriptive text, interactive dashboards and other rich media. In combination with the *container*-based[25] execution environments of the EOSC Cloud Compute Platform, Jupyter Notebooks improves the reproducibility of analysis. The growing popularity of computational notebooks presents a unique opportunity to support reproducible research. However, the reproducibility of the notebooks also depends on the availability of the required data it uses. Herein, it is normally recommended that users share the required input dataset alongside the Jupyter notebooks, which can readily be done through repository services such as *GitHub*[26] and *Zenodo*[27]. This guarantees that datasets are persistently and uniquely identified via *Digital Object Identifiers* (DOIs)[28]. Zenodo is an open access repository for research publications, scientific data and other 'research objects' and, together with Jupyter and Binder, is considered a key pillar of Open Science.

EGI-ACE provides access to Jupyter Notebooks and Binder to offer an interactive and reproducible computing platform for building powerful development environments, as discussed in Section 6.2. These environments, combined with the DataHub service, have been tested in a pilot data transfer activity in the context of the *PaNOSC*[29] project.

1.4. Data Privacy

The EOSC Compute Platform services comply with a set of policies that ensure that the services operate in good faith, without deliberately exposing the user to security risks,

[21] http://diracgrid.org/
[22] https://www.wenmr.eu/
[23] https://jupyter.org/
[24] https://jupyter.org/binder
[25] Containers are a mechanism to pack and run applications together with their runtime dependencies.
[26] https://github.org/
[27] https://zenodo.org/
[28] https://www.doi.org/
[29] https://www.panosc.eu/

without claiming intellectual property of user-owned data, and protecting sensitive data generated during interactions between users and the services. The services do not access or use customer content for any purpose other than as required by law and to maintain the services and provide them to the service customers and their end users.

More specifically, to be part of the EOSC Compute Platform, the service provider must:

- Comply with the EGI's *Policy on the Processing of Personal Data*[30].
- Provide its own Privacy Policy, which can be based on the Privacy Policy template provided by the *AARC Policy Development Kit* (PDK)[31].
- Comply with all principles established in the *GÉANT Data Protection Code of Conduct*[32].
- Meet all requirements of any relevant EGI policies or procedures[33] and must also be compliant with the relevant national legislation.

When accessing a service through EGI Check-in, the user is shown both the *Terms of Use* and a *Data Privacy Policy,* and consent to transfer data to the provider is required. These documents cover all aspects required by the GDPR[34] (*e.g.* who is the *Data Controller* and the *Data Processor*, what personal data is processed when accessing and using the service, how long personal data will be retained, *etc.*). Additionally, Check-in also maintains a browsable list of regulatory documents[35] for further reference.

If users need to process personal data on the services as part of their research activities, the EOSC Compute Platform can support the creation of specific *Data Processing Agreements*[36] to comply with GDPR requirements.

1.5. The Handbook and You

If someone else has already organised access for you on the EOSC Compute Platform, then you can probably skip ahead to Section 4, Section 5, and Section 6. You will learn about working with data, using the computing solutions, and more, as offered by the EOSC Compute Platform.

If you need to request access and set up an account on the EOSC Compute Platform, please continue to Section 3, which explains the process step by step. Additional information about the main stakeholders of the EOSC Compute Platform and how the services of this Platform are offered to them is provided in Section 2.

[30] https://documents.egi.eu/document/2732

[31] https://aarc-project.eu/policies/policy-development-kit/

[32] https://wiki.refeds.org/display/CODE/Code+of+Conduct+for+Service+Providers

[33] https://www.egi.eu/about/policy/policies_procedures.html

[34] General Data Protection Regulation. See https://fra.europa.eu/sites/default/files/fra_uploads/fra-coe-edps-2018-handbook-data-protection_en.pdf

[35] Available at https://aai.egi.eu/proxy/module.php/rciaminfo/services.php

[36] See https://gdpr.eu/what-is-data-processing-agreement/ for more information

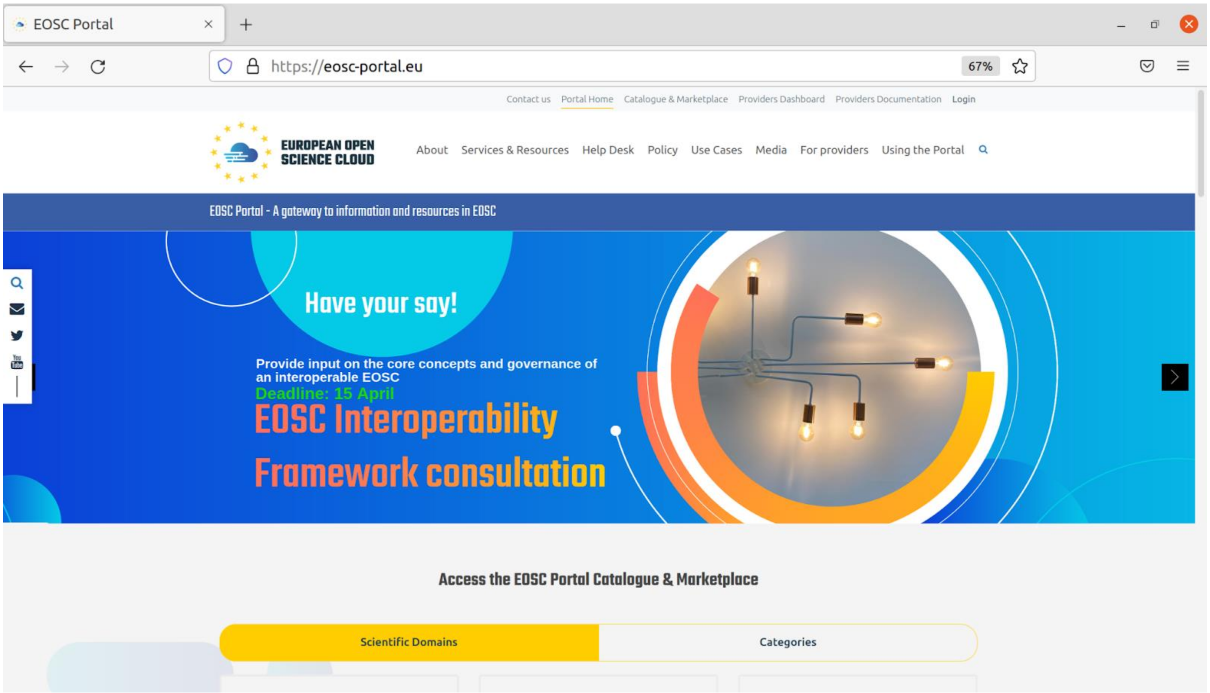
2. Accessing the EOSC Compute Platform

EGI-ACE's primary objective is to deliver services for Open Science. The project makes the services via two channels, both fitting to different audiences:

1. Individual users interested in one or few services for a relatively short period of time are served via the EOSC Portal and the Marketplace. More detailed information about the EOSC Portal and the Marketplace are provided in Section 2.1 and Section 2.2.
2. Structured scientific communities, research projects and Research Infrastructures require combined, and customized integration of Compute Platform services. They are served via the EGI-ACE open call, which is explained in Section 2.2.2.

2.1. The EOSC Portal

The EOSC Portal[37] (See *Figure 2.1*) is a catalogue that brings services and resources (e.g. databases, data catalogues) designed for Open Science visible in a single place. Through the EOSC Portal researchers will be able to discover open and seamless services, data, and other resources from a wide range of national, regional and institutional public research infrastructures across Europe. Through the Marketplace section of the Portal (See *Figure 2.2*) one can access the services either in fully open access mode, or after service-specific authentication and authorisation steps.



EOSC Portal - A gateway to information and resources in EOSC

Have your say!

Provide input on the core concepts and governance of an interoperable EOSC

Deadline: 15 April

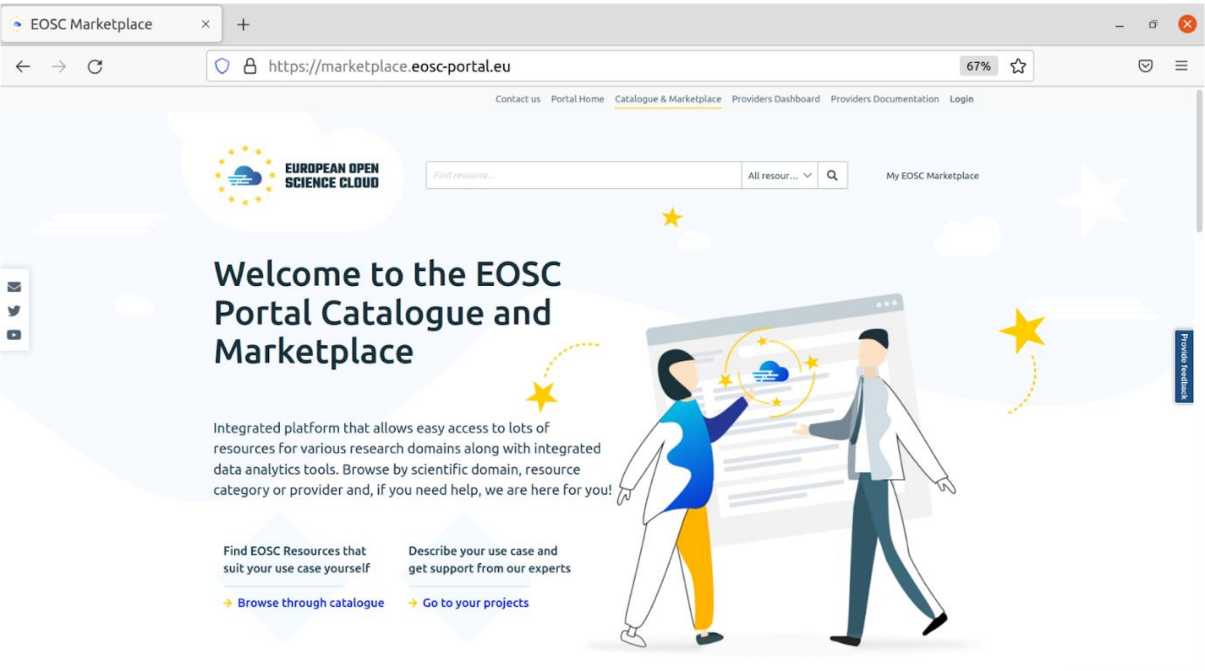
EOSC Interoperability Framework consultation

Access the EOSC Portal Catalogue & Marketplace

Figure 2.1 - The EOSC Portal website

[37] Available at https://eosc-portal.eu/

In the context of the EOSC Future project, interoperability guidelines will be developed to further extend the Portal in order to help services to be compliant with specific standards or interface. These guidelines will contribute to facilitating the interoperability of datasets and tools from different providers and enable researchers to perform their work more quickly and disseminate their research results more widely. The Portal offers information for potential service providers on how to onboard their services to the EOSC Portal Catalogue and Marketplace (in the “For providers” section). Future guidance for providers of data, providers of training is expected in the coming years.



Welcome to the EOSC Portal Catalogue and Marketplace

Integrated platform that allows easy access to lots of resources for various research domains along with integrated data analytics tools. Browse by scientific domain, resource category or provider and, if you need help, we are here for you!

Find EOSC Resources that suit your use case yourself

Describe your use case and get support from our experts

Figure 2.2 - The EOSC Marketplace website

The EOSC Portal Marketplace is the user interface for researchers looking for quality services for Open Science. From a technical perspective, from the EOSC Portal Marketplace, the researchers can browse the catalogue and explore multi-disciplinary resources from leading organizations. The Marketplace offers capabilities to manage the entire research data lifecycle from creation to processing, analysis, preservation, access and reuse. Through the Marketplace users can also get assistance from international technical teams who provide advice on the most suitable solutions to address their digital needs.

2.2. The EOSC Cloud Compute Delivery model

In this section, we will dive into the engagement process designed in the context of the EGI-ACE process to expand the demand-side of EOSC across sectors and scientific disciplines. To engage with simple researchers and multi-national scientific communities, two channels have been defined, as shown in *Figure 2.3*.

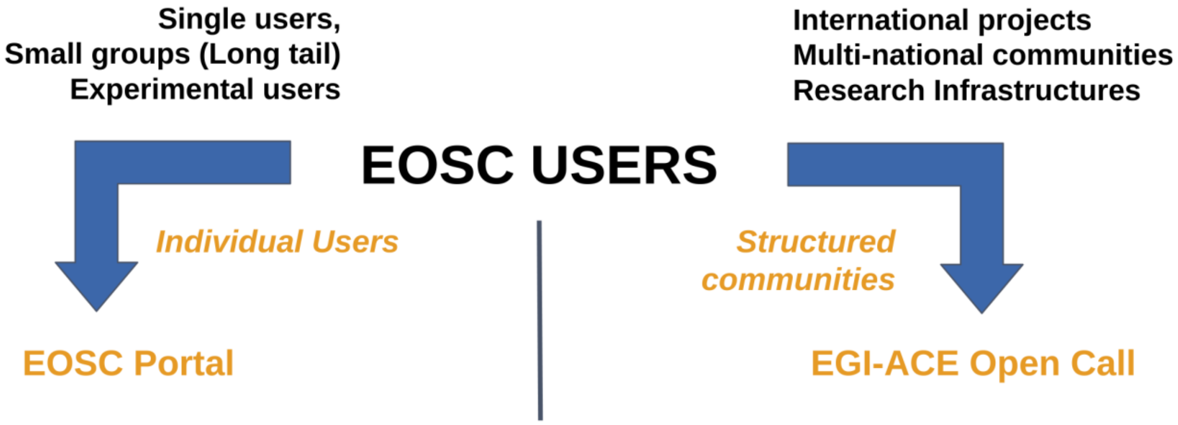


Figure 2.3 - The EOSC Compute Platform delivery model

2.2.1. Engagement with individual users (long tail)

For single or small groups of users involved in piloting activities, the EOSC Compute Platform has already pre-allocated, ready-to-use resources. Requests to access the pre-allocated resources and high-level solutions offered by EOSC can be submitted directly via the Marketplace. From the Marketplace this stakeholder group can easily discover the EOSC resources and services that suit the specific use cases. For example, in this way we served researchers from the Mario Negri Institute for Pharmacological Research in Milan[38] who focused on the identification of new markets aimed at predicting and preventing the development of epilepsy in individuals exposed to risk factors. During the COVID-19 pandemic, researchers from the University of Bologna addressed the relationship between coronaviruses, human health, pig health and animal breeding in their most recent research. In practice, they investigated how pigs could represent a risk source of coronavirus infections for humans as this could cause damaging effects by disrupting livestock production chains. Researchers from the University of Helsinki in Finland used the Chipster platform[39] for helping virologists to make sense of millions of virus genomes. In all these examples the high-level solutions offered by EOSC contributed to support their research activities and stimulate innovation.

2.2.2. Engagement with structured communities

For projects with international footprint, Research Infrastructures (RIs) and multi-national communities the EGI-ACE project has opened a call for use cases. Overall, this stakeholder group is mainly interested in the EOSC service offerings for long term, customized use (>3 years) and, for this reason, this group provides an excellent opportunity to expand the demand-side of EOSC across sectors and scientific disciplines.

Through the open call[40] EGI-ACE offers access to the EOSC infrastructure and platform services, dedicated user support and training. The access to services of the EOSC Compute

[38] https://www.marionegri.it/eng/home
[39] https://chipster.csc.fi/
[40] https://www.egi.eu/projects/egi-ace/call-for-use-cases/

Platform are sponsored by the European Commission via the Virtual Access (VA) mechanism and various national and institutional funding. More details about the cost of use are provided in Section 3.2.

For this stakeholder group, EOSC offers dedicated technical support, access to pledged resources and services with tailored configurations for helping the group to address a specific need.

2.2.3. Characteristics of the EOSC engagement process

Table 1.1 - Main characteristics of the two engagement processes

Process Type	Characteristics
Engagement with individual users	● Optimised towards ready-to-use resources ● Short-term engagement (weeks/months) ● Encourage self-service configurations ● Site-level support via the EGI Federated members ● Available users docs
Engagement with structured communities	● Optimised towards custom allocation of resources ● Long-term partnership (years) with agreed MoU or Collaboration Agreements ● Dedicated support via the EGI Federated members involved in the agreed MoU or Collaboration Agreements ● Custom set-up of services configuration

For individual users the starting point is to submit a service order via the EOSC Marketplace at: https://marketplace.eosc-portal.eu.

For the structured communities the project offers the possibility to submit use case applications at: https://www.egi.eu/projects/egi-ace/call-for-use-cases/. The evaluation process used to decide which use cases will be officially supported by the project is described in the section below.

2.3. Evaluating Requests and Granting Access

The review of the EGI-ACE use cases submitted by structured communities is carried out by multiple reviewers in a blind peer-review fashion. During the review process, the following four different aspects of the use case are assessed:

- **Maturity:** This assessment determines the current state of maturity of the application/platform proposed by the scientific community, understanding what it does and the results it produces.

- **Feasibility:** This assessment determines whether the objectives can be realistically achieved within the envisaged timeframe if there is special need for technical support and training and services.
- **EOSC readiness:** This assessment determines whether there are clear plans to share the ultimate results (datasets, services, etc.) in EOSC.
- **Impact:** This assessment determines the impact of the use case across Europe, or even beyond Europe.

If the use case is accepted during the blind peer-review process, the project grants access to infrastructure and platform services, resources, and offers dedicated support and training when needed. More specifically, the support is offered taking into consideration different aspects such as:

- The beneficiaries of the use case,
- Countries where the community has a footprint,
- How much the scientific topic matches the EGI members' support priorities, and
- National and institutional investments.

At the end of the peer-review process, if the use case is accepted, the following level of support are offered by the project:

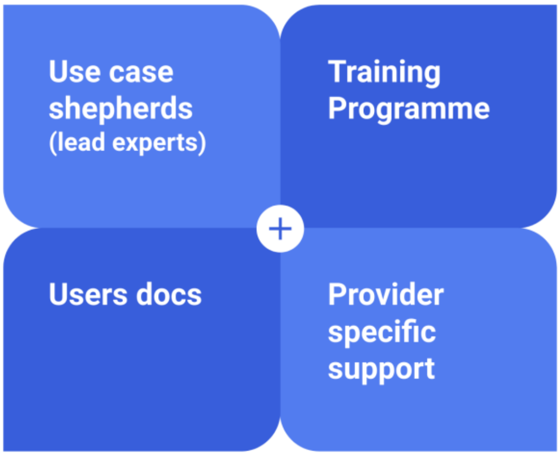


Figure 2.4 - The level of support offered by EGI-ACE to scientific communities

- **Assign technical lead experts**: Shepherds are technical experts who are assigned to the use cases that are served by the project and have the overall responsibility for mapping the use cases to providers, managing the implementation of the use cases, and reporting about achievements, lessons and other outcomes. Shepherds create "Competence Centers", use case specific support groups that consist of service providers, technical experts and other interested parties providing assistance for a use case.
- **Training programme**: To improve technical skills of researchers and scientific communities on the EOSC Compute Platform services, the project delivers a detailed training programme. Up to know this training programme is only composed by online

webinars available on the official website[41]. If traveling or physical presence will be supported, additional face-to-face events will be organised during the second part of the project.

- **Users docs**: Documentation about the EOSC Compute Platform services to target the needs of Users and Providers are available on the website[42].
- **Provider specific support**: If necessary, dedicated support from the services/resources providers are offered to support the specific needs of the use case.

[41] https://www.egi.eu/webinars/

[42] https://docs.egi.eu/

3. Accounts in the EOSC Compute Platform

Before getting access to the services and resources of the EOSC Compute Platform is to register an account. Different options are offered to access the services and the resources of the Platform. More specifically, the access to the Platform is primarily based on the OpenID Connect (OIDC)[43] standard, which replaces the legacy authentication and authorization based on X.509 digital certificates[44]. X.509 digital certificates are still used to grant access to HTC computing and storage resources even if their use will be deprecated soon. Local accounts are still adopted to access HPC clusters of the IaaS hybrid layer.

Pros and cons in using local accounts, X.509 digital certificates, and token-based authentication mechanisms to access the services of the EOSC Compute Platform are reported in *Table 3.1*.

Table 3.1 - Use of different accounts in the EOSC Compute Platform: pros and cons

Type	Pros	Cons
Local account	● Access with public key Authentication. ● File transfer and Web applications are available.	● Scalability
X.509 Certificate	● **Trust** - Digital certificates allow individuals, organisations, and even devices to establish trust in the digital world. ● **Scalability** - it can secure billions of messages exchanged daily across the internet.	● **Usability** - X.509 certificates are difficult to handle for users. ● **Interoperability** - Hard to integrate identity federations.
OIDC token	● Accelerate the process by allowing users to sign into websites with a single click. ● Reduce frustration associated with maintaining multiple usernames and passwords. ● Gain greater control over online identity. Users can control how much personal information can be shared with websites. ● Reduced password security risks.	● The sign-in process sometimes can be very confusing – It requires navigating to another site and giving the consensus to authorize the release of the personal information.

[43] https://openid.net/connect/

[44] https://sectigo.com/resource-library/what-is-x509-certificate

3.1. Setting up Your Account

The following section details the steps through which researchers can quickly create a personal account to start profiting from the EOSC Compute Platform to support their daily work.

The EGI Check-in[45] service provides a secure and user-friendly federated authentication and authorisation framework for helping researchers to access the different layers of the EOSC Compute Platform (*see Figure 1.2*). EGI Check-in enables Single Sign-On (SSO) to services through eduGAIN[46] and other identity providers. Users without institutional accounts can access services through social media or other external accounts, including Google, Facebook, LinkedIn or ORCID (just to name a few). From a technical perspective, to grant access to the Platform Check-in uses two different tools: via COManage[47] or PERUN[48].

1. To **register an account**, please go to https://aai.egi.eu/signup and submit the web form. The full registration flow to register a personal account is described in the official documentation[49].

As soon as the account has been setup, users of the Computing Platform can leverage the EGI Check-in service to obtain OAuth2[50] access tokens (and, optionally, refresh tokens) to securely access the available services described in Section 4, Section 5, and Section 6.

Second step is to subscribe to one of the domain-specific groups (alias for Virtual Organisations) that are configured to support the needs of a specific community.

2. To **enrol into one of the domain-specific groups** created to address the needs of a scientific community, please check out the available groups (alias Virtual Organizations) in the EGI Operations Portal[51] (under the VO Management menu). Each VO registered in the EGI Operations Portal has a different scope. Most of them serve specific communities (e.g.: alice, enmr.eu, etc.), others can be used for piloting and training activities in EOSC (e.g.: vo.access.egi.eu, training.egi.eu). For more details, please check the dedicated VO id card published in the EGI Operations Portal. From a technical perspective, to register a VO, a user has to apply for VO membership. By default this VO membership is valid for 1 year. The URL to subscribe to the VO is published in the VO id card (Enrolment URL). VO membership is usually received and evaluated by the VO Managers. Users are notified by email about the approval/rejection of the VO membership request.

[45] https://www.egi.eu/services/check-in/
[46] https://edugain.org/
[47] https://incommon.org/software/comanage/
[48] https://perun-aai.org/
[49] https://docs.egi.eu/users/aai/check-in/signup/
[50] https://oauth.net/2/
[51] https://operations-portal.egi.eu/

3. (optional) HTC resources and some of the Data services of the EOSC Compute Platform still use X.509 certificates. To **get a digital certificate** from one of the trusted Certification Authorities (CAs), please check the available documentation[52].

3.2. Cost of Use

The resources in the EOSC Compute Platform are offered by service providers that either run their own data centers or rely on community, private and/or public cloud services. More specifically, these service providers offer:

- Single Sign-On (SSO) via EGI Check-in allows users to login with their institutional (community) credentials.
- Global image catalogue at AppDB[53] with pre-configured virtual machine images.
- Resource discovery features to easily understand which providers are supporting your community, and what are their capabilities.
- Global accounting that aggregates and allows visualization of usage information.
- Monitoring of availability and reliability to ensure SLAs are met.

The full capacity of the EOSC Compute Platform is offered through the following access policies:

- **Policy-based** access is granted to users based on specific policies defined by the service providers. Access needs to be requested and will be checked for such services.
- **National funding** mobilized by the national e-Infrastructures participating in the EOSC Compute Platform.
- **EGI participants' funding** supporting the cost of EGI federation services.
- **Pay-per-use** means access to the services is provided with a fee.

[52] https://docs.egi.eu/users/aai/check-in/vos/voms/

[53] https://appdb.egi.eu/

4. Working with Data

One of the main daily activities of researchers is to collect or create data that needs to be processed, analysed, or shared. The EGI-ACE offering in this area is quite vast and it includes low level storage services, data management frameworks and data transfer services.

The Federate Data Subsystem in EGI-ACE comprises a group of services (e.g. DataHub and Rucio) that provide data management capabilities to enhance the raw storage delivered by the Federated Resource Providers. Specialized services (e.g. openRDM) offer advanced organisation of data during ongoing research projects as an integrated environment with data management and digital lab notebook. The Federated Data Services offer APIs and CLIs that are integrated by both the Data Spaces and by the Platforms available in EGI-ACE and can be interfaced to the different computing services (Federated Cloud, HTC and HPC).

4.1. Range of Data Services

As introduced in the previous section, EGI-ACE offers mainly three categories of services to deal with data:

- Low level storage services coming from the Federate resource layer .
- Data Management services coming from the Federated Data Subsystem.
- Special data management services either used for scheduling transfer between storages or to organise lab experiments data.

Low level storage services are provided under a unique service in the EGI Catalogue: Online Storage[54], which includes the categories summarized in *Table 4.1.*

Table 4.1 - Online storage types in EGI-ACE

Type	Description	Available From
Online Storage - Block Storage	Block-level storage that can be attached to VMs as volumes, a simple solution for durable data that does not need to be shared beside a single VM.	Cloud Compute, Container Compute
Online Storage - Grid Storage	File storage for high throughput and/or performance scenarios.	High Throughput Compute
Online Storage - Object Storage	Persistent, hierarchical blob storage for cloud native applications, archiving, and when data is shared between	Cloud Compute, Container Compute

[54] https://docs.egi.eu/users/data/storage/

Type	Description	Available From
	different VMs or multiple steps of processing workflows.	

The higher-level data management services available to researchers are listed in *Table 4.2*.

Table 4.2 - Data management services of the Computing Platform

Type	Description
EGI DataHub	High-performance data management solution that offers unified data access across multiple types of underlying storage, allowing users to share, collaborate and perform computations on the stored data easily. It serves as well as Open Data repository with OAI-PMH interface and PID/OID minting and metadata indexing service.
EGI Rucio	Service tailored to medium/big scientific collaborations with the functionality to organise, manage, and access their data at scale. The data can be distributed across heterogeneous data centers at widely distributed locations.

Other services are specialized data management services, either for scheduling transfer between storages or to organise lab experiments data. These are summarized in *Table 4.3*.

Table 4.3 - Specialized data management services

Type	Description
Very Large Data Transfers to the Compute Platform	Low level service to move data from one Grid or Object storage to another. It’s used internally by Rucio itself to schedule transfers based on the data policies defined by the users.
Additional Data Management Cases	Combined FAIR data management platform, Electronic Laboratory Notebook (ELN) and Inventory Management System allowing a complete overview of workflows and information, from initial data generation to data analysis and publication.

4.2. Storing and Managing Research Data

As explained in the previous section, depending on the type of EGI Compute Platform and use case you are addressing, you might need to choose a different data service to store, access and manage data.

- **EGI Cloud user**: Online storage and in particular Block and Object storages, with optional high-level management implemented by the EGI DataHub.
- **EGI HTC user**: Online Storage, in particular the Grid Storage, with optional data management functionalities implemented by the EGI Rucio.
- **EGI HPC user**: High performance parallel file systems (site specific) and/or Object Storage eventually managed via the EGI Rucio or EGI DataHub.

The access to data storage services can be achieved through the use of credentials as explained in Section 3. Depending on the different Data services available in EGI, there are different authentication mechanisms available as not all data management services are yet enabled to use EGI Check-in and OIDC Token credentials. For those services legacy X509 certificates are needed instead.

The following services need an X.509 certificate to store or access data:

- Online Storage - Grid Storage
- EGI Rucio
- EGI Data Transfer

The following services instead are already integrated with EGI Check-in (or need other types of credentials but X.509 certificates):

- EGI DataHub
- Online Storage - Block Storage
- Online Storage - Object Storage
- openRDM

In the following subsections of this chapter we will provide more details on the different types of storage/data management service and how to interact with them.

4.2.1. Online Storage - Grid Storage

Grid storage[55] enables **storage of files in a fault-tolerant and scalable environment** and sharing it with distributed teams. Your data can be accessed through multiple standard protocols and can be replicated across different providers to increase fault-tolerance. Grid storage gives you complete control over what data you share, and with whom you share the data.

The main features of grid storage:

- Access highly scalable storage from anywhere
- Control the data you share
- Organise your data using a flexible and hierarchical structure

[55] https://docs.egi.eu/users/data/storage/grid-storage

Grid storage file access is based on the gridFTP[56] and WebDav/HTTP[57] protocols, together with XRootD[58] and legacy SRM[59] (under deprecation at some of the endpoints).

Several grid storage implementations are available in the EGI infrastructure, the most common being:

- EOS[60]
- dCache[61]
- DPM[62]
- StoRM[63]

The Grid storage endpoints on the infrastructure are discoverable thanks to their publication on the EGI Information Systems (BDII) as described in the dedicated documentation page[64].

The interaction with the storage is eased by a CLI for the gfal2 library[65], which provides an abstraction layer on top of several storage protocols, offering a convenient API that can be used over different protocols. You can check our online documentation about the available commands[66] and APIs. The Grid storage is already integrated with the EGI Workload Management service described in section 6.4, allowing automatic stage-in and stage-out of data when running your workloads.

4.2.2. EGI Rucio

EGI Rucio[67] is a data management service which abstracts the data distribution in multi-location storage on the EGI Infrastructure. Hence, it's quite convenient if you are planning to store data in Grid storages and organise it into a common namespace to abstract over the physical location of the files.

Furthermore Rucio allows you to manage your data with expressive statements that allow the definition of Policy on data and let Rucio figure out the details of how to reach the desired result. For example, *Three copies of your files on different continents with a backup on tape*. You can also automatically remove copies of data after a set period or once its access popularity drops.

You can check the online documentation on how to get started with Rucio[68] which will also introduce the related terminology and explain how to set up the Rucio client.

[56] https://en.wikipedia.org/wiki/GridFTP
[57] https://en.wikipedia.org/wiki/WebDAV
[58] https://xrootd.slac.stanford.edu/
[59] https://www.gridpp.ac.uk/wiki/SRM
[60] https://eos-web.web.cern.ch/eos-web/
[61] https://www.dcache.org/
[62] https://twiki.cern.ch/twiki/bin/view/DPM/
[63] https://italiangrid.github.io/storm/
[64] https://docs.egi.eu/users/data/storage/grid-storage#endpoint-discovery
[65] https://dmc-docs.web.cern.ch/dmc-docs/gfal2/gfal2.html
[66] https://docs.egi.eu/users/data/storage/grid-storage#access-from-the-command-line
[67] https://docs.egi.eu/users/data/management/rucio/
[68] https://docs.egi.eu/users/data/management/rucio/getting-started/

As explained in the documentation, in the management of your research data there are two possible users' roles in Rucio to consider:

- Admins: they look after the day-to-day requirements of their community of research which means they will need to set up and maintain accounts, identities, quotas, storage endpoints (RSEs).
- End users: they upload, register and access data, create rules in order to replicate data among the different storage endpoints, etc.

The Rucio client implements both commands for admins[69] and end users[70] and specific documentation is available for the two sets of available commands.

4.2.3. Online Storage - Block Storage

Block storage[71] provides **block-level storage volumes** to use within virtual machines (VMs) available on the EGI Cloud. Block storage volumes are raw, unformatted block devices, which can be mounted as devices in VMs.

Block storage volumes that are attached to a VM are exposed as storage volumes that **persist independently from the life of the VM** and need to be explicitly destroyed when data is not needed anymore. Users can create a file system on top of these volumes or use them in any way they would use a block device (such as a hard drive).

The EGI Cloud is based on the Openstack technology, therefore the management of the block storage can be done either using the graphical dashboard or the Openstack client. EGI has also implemented a client (FedCloud client[72]) which can be used as well to create and manage block storage. All those possible access methods are described on our documentation[73]. The above methods are also needed when dealing with provisioning volumes to your Virtual machines as explained in our documentation[74].

4.2.4. Online Storage - Object Storage

Object storage[75] is a standalone service that **stores data as individual objects, organised into containers**. It is a highly scalable, reliable, fast and inexpensive data storage. It has a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web.

The main features of object storage are:

- Storage containers and objects have unique URLs, which can be used to access, manage, and share them.

[69] https://docs.egi.eu/users/data/management/rucio/admin/
[70] https://docs.egi.eu/users/data/management/rucio/commands
[71] https://docs.egi.eu/users/data/storage/block-storage
[72] https://docs.egi.eu/users/getting-started/openstack
[73] https://docs.egi.eu/users/data/storage/block-storage#manage-from-the-command-line
[74] https://docs.egi.eu/users/data/storage/block-storage#access-from-your-vms
[75] https://docs.egi.eu/users/data/storage/object-storage

- Data can be accessed from anywhere, using standard HTTP requests to a REST API (e.g. VMs running in the EGI Cloud or in another cloud provider's cloud, from any browser/laptop, etc.).
- Access can be public or can be restricted using access control lists.
- There is virtually no limit to the amount of data that can be stored. Only the space used is accounted for.

The documentation includes more details on the specific concepts[76] of Object storage, like storage containers, objects, etc.

Object storage in the EGI infrastructure is available via two protocols, Swift and S3[77]. There are different clients and access methods to use depending on the protocols, and there is specific documentation for Swift[78] and S3[79] as well.

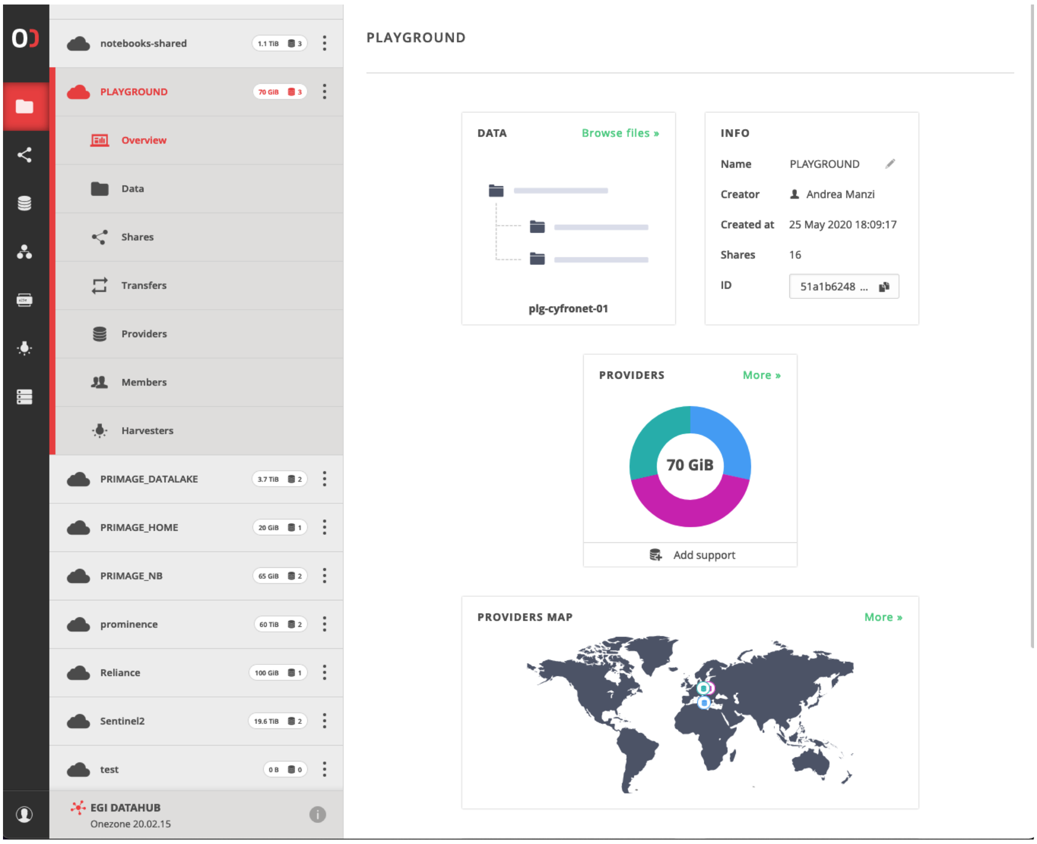
4.2.5. EGI DataHub

The EGI DataHub[80] is based on Onedata technology and it aims at abstracting the access to distributed storage available on the infrastructure (both POSIX and Object Storages), implementing a unique namespace which can be easily accessed via GUI or APIs.

As described before, the service is integrated with EGI Check-in, and it provides a GUI both for end-users and administrators to access and manage data and configure the storage endpoints. Our documentation[81] includes concepts and terminology related to Onedata and the EGI DataHub.

An important aspect of the EGI DataHub is its flexible metadata mechanism which allows the storage of metadata in the form of simple key value pairs, as well as entire metadata documents (currently in JSON and RDF formats), which can be attached to data resources and used during indexing and querying. Building on top of this metadata mechanism, EGI DataHub enables users to publish their data as open access content.

[76] https://docs.egi.eu/users/data/storage/object-storage#concepts
[77] https://www.architecting.it/blog/object-storage-standardising-on-the-s3-api/
[78] https://docs.egi.eu/users/data/storage/object-storage#access-from-the-command-line
[79] https://docs.egi.eu/users/data/storage/object-storage#access-via-the-s3-protocol
[80] https://docs.egi.eu/users/data/management/datahub/
[81] https://docs.egi.eu/users/data/management/datahub#components-and-concepts



INFO

Name PLAYGROUND

Creator Andrea Manzi

Created at 25 May 2020 18:09:17

Shares 16

ID 51a1b6248 ...

Figure 4.1 - The EGI DataHub graphical interface

The EGI DataHub supports several protocols and standards for open data such as:

- OAI-PMH[82] for publishing data, handle system integration for registering
- DOI[83] identifiers, enabling full open data management life cycle management from ingestion through curation to open access.

The EGI DataHub enables seamless sharing of data between users, with strict access control. Users can share access to individual files as well as spaces by sending automatically generated access tokens. Access control is fine grained and managed by a set of privileges assigned in the form of access control lists on the level of users, groups and spaces. Users can create groups for collaborating on a specific space or set of spaces.

Users can access the EGI DataHub by directly interacting with the GUI or installing the **oneclient** component and the related Python bindings[84], which enable POSIX access to storage federated via the EGI DataHub. For what concerns the integration with API, data access can be achieved either via REST API or via CDMI[85]. For **oneclient** and API access a specific token needs to be generated either via GUI or via Onezone API as described in

[82] https://www.openarchives.org/pmh/
[83] https://www.doi.org/
[84] https://docs.egi.eu/users/data/management/datahub/clients/
[85] https://docs.egi.eu/users/data/management/datahub/api/

our documentation[86]. The documentation also includes a specific section[87] on how to register, handle services and mint a Permanent Identifier (PID) for a space or a subdirectory.

4.3. Very Large Data Transfers to the Compute Platform

Support for large data transfers in and out of the EGI Compute Platform can be achieved through the use of EGI Data Transfer[88] which allows the transfer between:

- Two grid storages (including staging from tape storages)
- One grid storage and a one object storage
- Two object storages.

The service is based on the FTS[89] service and a web interface, WebFTS, is also provided that allows a simpler user experience.

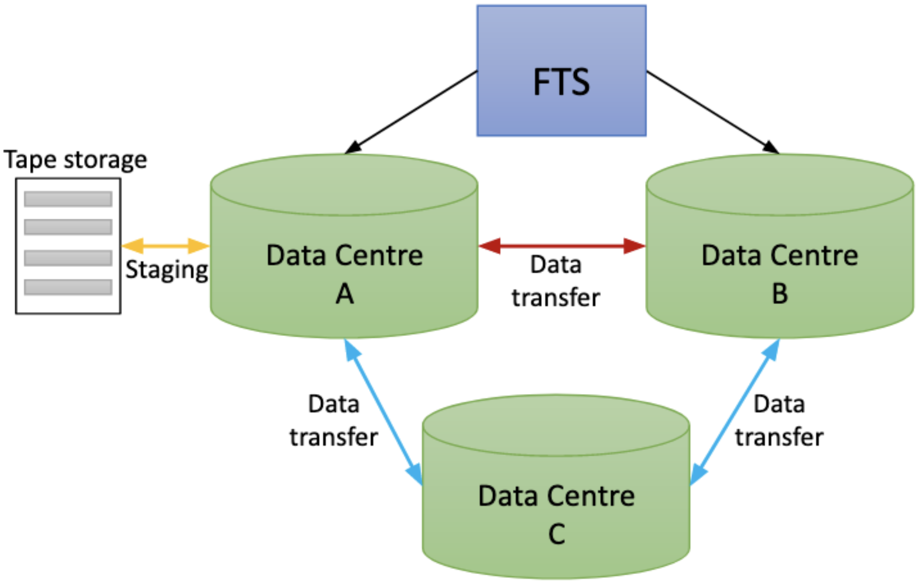


Figure 4.2 - Data flow example with EGI Data Transfer

Step by step tutorials for the users can be found on the EGI documentation in which is provided a tutorial for “Data transfer with grid storage”[90] and “Data transfer with object storage”[91].

Furthermore the transfer in and out of the EGI Compute Platform is provided free of charge so that the desired data can be transferred from any provider to the EGI infrastructure.

[86] https://docs.egi.eu/users/data/management/datahub/api#getting-an-api-access-token
[87] https://docs.egi.eu/users/data/management/datahub/api#working-with-pid--handle
[88] https://docs.egi.eu/users/data/management/data-transfer/
[89] https://fts.web.cern.ch/fts/
[90] https://docs.egi.eu/users/tutorials/data-transfer-grid-storage/
[91] https://docs.egi.eu/users/tutorials/data-transfer-object-storage/

4.4. Additional Data Management Cases

While the services described in the previous sections are available in the EGI infrastructure, there are specialized services in the EGI-ACE portfolio that are offered for in-house installations by research communities, with support for customization and configuration from EGI.

The openRDM[92] service offers advanced organisation of data during ongoing research projects as an integrated environment offering data management and digital lab notebook. The openRDM service is based around the active research data management (ARDM) platform openBIS[93].

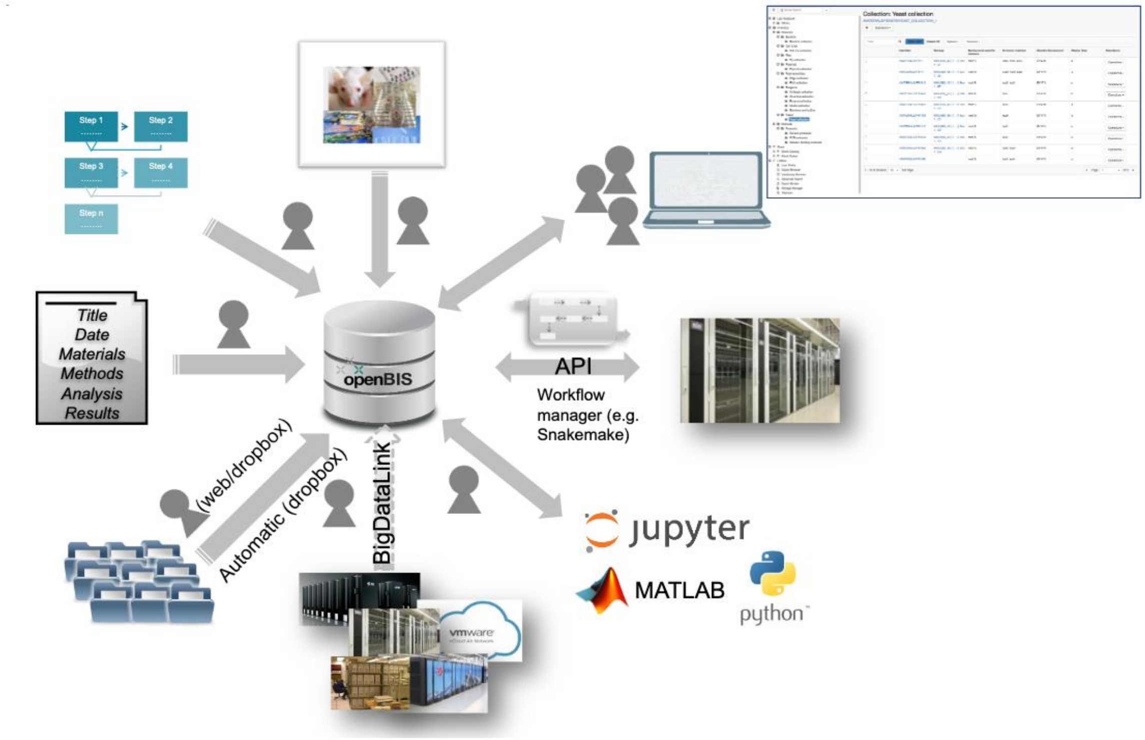


Figure 4.3 - openRDM functionalities

A demo instance of openRDM is available from some of the sites in the EGI Federation, so that users can test its functionality. Communities are then supported to install the service on their premises.

openBIS combines a data management platform with a digital lab notebook and a sample and protocol management system. It enables scientists to meet the ever-increasing requirements from funding agencies, journals, and academic institutions to publish data according to the FAIR data principles – according to which data should be Findable, Accessible, Interoperable and Reusable. More details on openBIS are available on the online documentation[94].

[92] https://marketplace.eosc-portal.eu/services/openrdm-eu
[93] https://openbis.ch/
[94] https://openbis.ch/index.php/docs/user-documentation/

5. Working with Compute

Most scientific work researchers do include computation. Could be the need to run a small analysis. Or a data reduction job that has to be repeated a million times on every measurement coming in from an ionosphere observation facility. Or a handful of cutting-edge simulations of climate models that each demand a huge number of compute cores. The EOSC Compute Platform provides a variety of compute services for each of these use cases, and you can choose the best compute tool for each application, be it CPU or GPU.

A paradigm that works as well for compute as it does for storage is elasticity. Cloud elasticity means there is always room for one more processing core or storage volume: researcher's simulations are not constrained to the fixed size of a cluster in their department's basement but can scale to however many cores are needed for the question being analysed.

The offer of compute services of EGI-ACE covers different types of applications and use cases: from customisable and elastic Virtual Machine based cloud to a fully managed distributed platform to run jobs. These are tightly integrated with the storage solutions described in the previous chapter and can be complemented with the advanced services described in the next chapter. From a technical point of view the Cloud and Container Compute offering of the EOSC Compute Platform relies on the EGI Federated Cloud Infrastructure[95], an IaaS-type cloud, made of academic private clouds and virtualised resources and built around open standards. Its development is driven by requirements of the scientific community.

The EGI-ACE compute offer is summarized in *Table 5.1.*

Table 5.1 - Compute services in EGI-ACE

Type	**Description**	**Use Case**
Cloud Compute	Provides access to VM-based computing with associated storage. It delivers a customisable platform where users have complete control over the software and the supporting compute capacity.	The flexibility of the computing platform enables a variety of workloads: user gateways or portals, interactive computing platforms and almost any kind of data- and/or compute-intensive workloads.
Container Compute	Supports the execution of container applications on top of Cloud Compute.	It provides deployment of Kubernetes, a widely used container orchestrator, that allows users to run and manage container-based applications.
High Throughput Compute	Provides access to large, shared Grid Computing systems for running computational jobs at	Analysis of large datasets in an "embarrassingly parallel" fashion, i.e. by splitting the data into small pieces,

[95] https://www.egi.eu/federation/egi-federated-cloud/

Type	Description	Use Case
	scale.	and executing thousands, or even more independent computing tasks simultaneously, each processing one piece of data.
High Performance Compute	Supports highly optimized applications that need massively parallel computing with low latency and high bandwidth interconnection network.	Complex computational problems using tightly coupled parallel processing: simulations, analysis of large datasets or AI/ML workloads. Typically these applications rely on MPI[96] for supporting inter process communication.

For all these services, resources can be scaled (within a single provider or using multiple providers in a coordinated fashion) to meet your research needs. Some of the EGI-ACE providers include in these offerings GPUs for accelerating your computations.

5.1. Software Distribution

Before running your applications, you need to bring your software to the EOSC Computing Platform. EGI-ACE provides two software distribution services to make it possible: the CernVM File System and the EGI Applications Database.

The CernVM-File System (CernVM-FS or CVMFS)[97] provides a scalable, reliable, performant and low- maintenance software distribution service. It is available as a POSIX read-only file system in user space and mounted in `/cvmfs` at all the HTC providers of EGI and can be configured as needed on Cloud Compute providers. User communities can upload their software in the repository management service[98], which provides a single place for users to publish their software, and a global network replica servers make this software instantly available on compute resources around the infrastructure.

The EGI Applications Database (AppDB)[99] delivers a public software catalogue, aiming at providing end users with easy to find scientific software and ease of use for cloud VM deployment. Any EGI user can upload their software in the form of Virtual Machine images to AppDB. Communities select from those VM images which ones to support, and providers will download and make them available for users automatically. AppDB provides a user-friendly interface for registering the VM images, managing the community endorsed VM images and to discover the availability of those images at the distributed computing providers. The workflow in EGI AppDB is depicted in *Figure 5.1*.

[96] Message Passing Interface (MPI) https://www.mpi-forum.org/
[97] https://cernvm.cern.ch/fs/
[98] https://docs.egi.eu/users/compute/content-distribution/
[99] https://appdb.egi.eu/

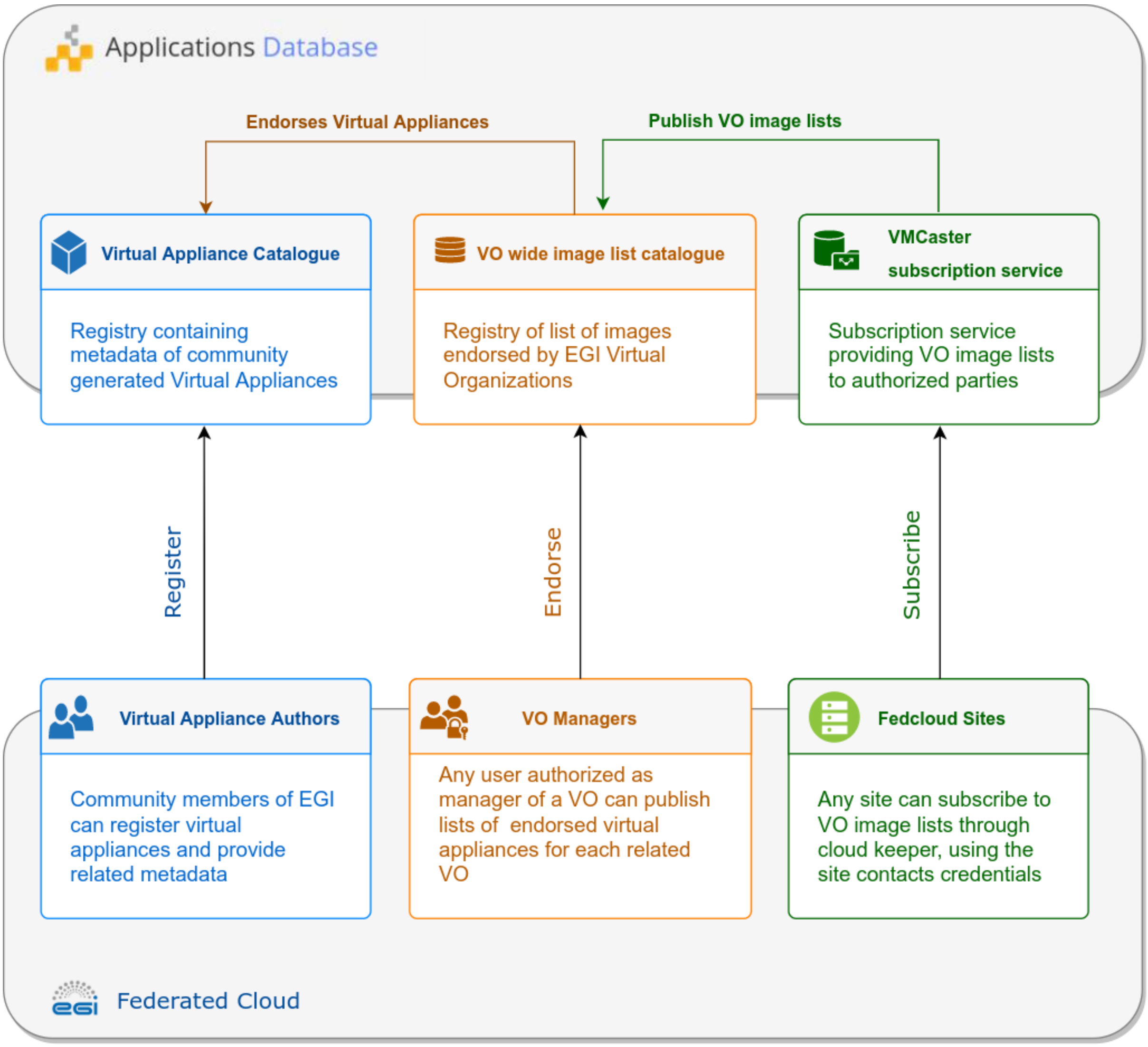

Applications Database
Endorses Virtual Appliances
Publish VO image lists
Virtual Appliance Catalogue
Registry containing metadata of community generated Virtual Appliances
VO wide image list catalogue
Registry of list of images endorsed by EGI Virtual Organizations
VMCaster subscription service
Subscription service providing VO image lists to authorized parties
Register
Endorse
Subscribe
Virtual Appliance Authors
Community members of EGI can register virtual appliances and provide related metadata
VO Managers
Any user authorized as manager of a VO can publish lists of endorsed virtual appliances for each related VO
Fedcloud Sites
Any site can subscribe to VO image lists through cloud keeper, using the site contacts credentials
Federated Cloud

Figure 5.1 - AppDB Software distribution components

Besides those services, EGI-ACE also provides a curated list of software packages for infrastructure operators to deploy the software products that power the EOSC Compute Platform: the Unified Middleware Distribution (UMD) and Cloud Middleware Distribution (CMD). The software included in these repositories[100] goes through a quality assurance process to ensure that new versions of the software meet the requirements for production in the global infrastructure.

5.2. Cloud Compute

The foundational compute service in EGI-ACE is Cloud Compute[101], which allows researchers to deploy and scale Virtual Machines (VMs) on-demand. It offers computational

[100] See https://wiki.egi.eu/wiki/Middleware for more information.
[101] https://marketplace.eosc-portal.eu/services/egi-cloud-compute

resources in a secure and isolated environment, controlled via APIs, without the overhead of managing physical servers.

Although the compute resources in EGI-ACE are provided by independent resource centers member of the EGI Federated Cloud, once you gain access to the EOSC Cloud Compute platform (see Section 2), you can access all resources, irrespective of which provider is hosting them, using the same identity.

The Cloud Compute service in EGI-ACE supports executing multiple kinds of compute- and data-intensive workloads (both batch and interactive), hosting long-running services (e.g. web servers, databases, or applications servers), and creating disposable testing and development environments on VMs. You can select the hardware configurations (CPU, memory, GPU, disk) for your VMs. You can also run any Operating System or application, by leveraging the VM images and available applications from the software distribution services described in the Section 5.1.

When scientific projects need different computing power at different moments in time, the number of VMs can be automatically scaled, both up and down, to meet demand, by employing an elastic computing cluster (see Section 6.3.2).

Working with VMs is easy, you can use a graphical user interface (GUI) of an orchestrator, e.g. Infrastructure Manager (see Section 6.3.1), to create new virtual servers, attach some block storage to them, and enclose them within custom-designed Virtual Networks (VNs)[102] that include built-in firewalls to add layers of protection. A few clicks and minutes later you will be able to access the machine via Secure Shell (SSH) and begin running software or importing data.

While all the above can be conveniently done from a GUI, nearly everything can also be done via a command line interface (CLI)[103] or an Application Programming Interface (API). This means that repetitive tasks or complex architectures can be turned into shell scripts.

Creating VMs happens so often in EGI-ACE that tools were developed to capture the provisioning of these VMs and allow you to recreate them in a flash, in a deterministic and repeatable way, using an Infrastructure-as-Code (IaC) approach. Automating this activity will help you to not forget important configuration (e.g. the size and hardware resources needed) or steps (e.g. making sure your datasets are attached to your VMs). Automation also makes it easier to share scientific pipelines with collaborators.

When you need to refer to your VMs using human-readable, memorable names instead of IP numbers, the Dynamic DNS[104] service of EGI-ACE allows you to register your domain names, which can then be dynamically updated by your VMs to point to themselves, ensuring you have meaningful, always up-to-date domain names for all the services and endpoints of your application[105]. This also allows you to simplify the configuration of

[102] With private or public subnets, as required (including public IP numbers)

[103] https://docs.egi.eu/users/getting-started/cli/

[104] https://marketplace.eosc-portal.eu/services/dynamic-dns-service

[105] See also the documentation at https://docs.egi.eu/users/compute/cloud-compute/dynamic-dns/

clients/servers, to freely move VMs between sites without disrupting clients, and to request certificates that remain valid even when the VMs are moved to another site.

For more information about Cloud Compute, please check out its documentation[106] or the tutorial on how to create your first VM[107].

5.3. Container Compute

The Container Compute[108] service in EGI-ACE lets you deploy, run, and manage Docker[109] containers. Containers easily package an application's code, configurations, and dependencies into simple building blocks. They allow applications to be deployed quickly, reliably and consistently, regardless of the environment where they will run.

There are multiple ways to run your containerized workloads in EGI-ACE:

- **Manually on a single VM**, by creating a VM where you can execute your container. This is facilitated by VM images in the Software Distribution with pre-installed Docker software.
- **Manually on multiple VMs**, by creating multiple VMs and deploying a container orchestrator (e.g. Kubernetes). This is facilitated by VM images with pre-installed Kubernetes software.
- **Automatically on elastic compute clusters**, by employing one of the orchestrators (see Section 6.3) supported in EGI-ACE. This approach will not only automatically provision the VMs needed to run your containerized workload but can also automatically resize the cluster that runs your workload to match demand. It can even run your workload on clusters that span more than a single IaaS Cloud provider.

For more information about Container Compute, please check out the documentation[110].

5.4. High Throughput Compute

High Throughput Computing (HTC) is a computing paradigm that focuses on the efficient execution of a large number of loosely coupled tasks (e.g. data analysis jobs). HTC systems execute independent tasks that can be individually scheduled on the distributed computing resources of EGI's federated infrastructure.

The EGI HTC[111] service provides easy, uniform access to shared high-quality computing and storage resources with integrated monitoring and accounting tools that provide information about the availability and resource consumption. Every provider of the infrastructure offers a front-end gateway (ARC-CE[112] or HTCondor-CE[113]) that manages the

[106] https://docs.egi.eu/users/compute/cloud-compute/
[107] https://docs.egi.eu/users/tutorials/create-your-first-virtual-machine/
[108] https://marketplace.eosc-portal.eu/services/egi-cloud-container-compute-beta
[109] https://www.docker.com/get-started
[110] https://docs.egi.eu/users/compute/cloud-container-compute/
[111] https://docs.egi.eu/providers/high-throughput-compute/
[112] https://www.nordugrid.org/arc/ce/
[113] https://opensciencegrid.org/docs/compute-element/htcondor-ce-overview/

authorization and the submission of jobs to a local batch system (e.g. Slurm, PBS, or HTCondor). These systems rely on X.509 certificates and VOMS[114] for authentication and authorization of users. See Section 3.1for details.

Compared to Cloud Compute or Container Compute where you have complete control over the resources and the ultimate responsibility for managing them, the HTC service offers a completely managed service where you only need to worry about the submission of jobs to the set of available distributed providers. HTC is tightly integrated with other components of the infrastructure: it provides access to files available through grid storage (see Section 4.2.1), it makes software available as 5.1, and it can be used with the workload manager (see Section 5.6) to simplify the management of jobs in the infrastructure.

A new flavour of the HTC service is under integration into EGI-ACE, based on the SURF Spider[115] platform. Spider is a versatile high-throughput data-processing platform, aimed at processing large, structured data sets, that runs on top of an internal elastic Cloud. Besides the job submission features, Spider supports Jupyter notebooks and many other user-centric features.

5.5. High Performance Compute

High Performance Computing[116] (HPC) provides highly optimized computing systems that deliver large amounts of parallel computing power to run applications. Similarly to HTC, it offers a managed service where you can find a fully operational environment where to submit your jobs. Traditionally, HPC systems are managed in an isolated way and offer limited federation options. In EGI-ACE a set of pilots are established to explore the integration of several HPC systems into the federation and facilitate access to them as part of the EOSC Compute Platform.

In most HPC systems, user access is performed via the Secure Shell Protocol (SSH) to a set of login nodes where users can interact with the system and submit jobs for their execution. The SSH credentials are locally managed usernames and password or SSH keys. For EGI-ACE, we are testing new ways of accessing the providers that leverage federated authentication, so users do not need to manage a new set of credentials for each of the individual systems. The following mechanisms are being tested for the EGI-ACE pilots:

- SSH access with OIDC tokens with ssh-oidc[117] (see Section 3 for details). This enables access to SSH using federated identity technology as supported by EGI Check-in. This is the preferred authentication mechanism for EGI-ACE as it minimizes the requirements to the users: you just need to get an access token from Check-in.
- Access via a HTC-like middleware. In this case, users do not get direct access via SSH, but a set of middleware components will handle the user access and the

[114] https://italiangrid.github.io/voms/index.html
[115] http://doc.spider.surfsara.nl/en/latest/Pages/about.html
[116] Available later in 2022
[117] https://github.com/EOSC-synergy/ssh-oidc

interaction with the HPC system. You would be able to submit jobs using the same tools as for the HTC service (see also Section 5.4for details). While this is the simplest option to implement for the infrastructure, it's not always possible to implement due to strict policy restrictions in HPC systems.

- HPC as a Service. In those providers who deliver HPC hardware via IaaS interfaces, you can easily deploy small-scale HPC systems to run your workloads. While IaaS providers rely on hypervisors to create Virtual Machines, specialized hardware like GPU accelerators and low latency networks like InfiniBand can be configured as PCI Passthrough devices without major performance overheads. You can use orchestration tools like the Elastic Cloud Computing Cluster (see Section 6.3.2) to create elastic virtual clusters on these providers. This approach allows for a very flexible and customizable environment while providing access to HPC features.

For providing portability of applications between systems (Cloud, HTC, HPC), EGI-ACE recommends the use of containers with udocker.

udocker[118] is a user-level tool that executes containers in user space, without requiring root privileges like in managed services such as HTC, HPC, or other externally managed batch/interactive systems. It is a wrapper around several tools and technologies that pulls container images and executes them with minimal functionality that mimics the Docker command line tool, thus facilitating its use for users who are already familiar with Docker.

Access of HPC systems to data available from the various data services of EGI-ACE described in the previous chapter is currently under piloting.

5.6. Workload Management

EGI's Workload Manager[119] is a service that provides efficient management and distribution of workloads on the distributed computing infrastructure of the EOSC Compute Platform. The service, based on the DIRAC Interware project[120], delivers a complete solution for communities that need to access heterogeneous computing and storage resources distributed geographically, integrated in different HTC and cloud infrastructures or standalone computing clusters and supercomputers. EGI Workload Manager handles workload distribution in a transparent way by choosing appropriate computing and storage resources in real-time for the users. It manages data storage, data movement, data accessing and data processing, while supporting different storage technologies (HTC, HPC and Iaas Cloud).

Figure 5.6 highlights the main components of the Workload Manager. Its architecture allows easily adding new types of computing resources and applying usage policies, as required by user communities. A REST API provides a language neutral interface to the service. A Web portal provides simple and intuitive access to most of the service functionalities including management of computing tasks and distributed data. This web GUI also has a

[118] https://github.com/indigo-dc/udocker
[119] https://docs.egi.eu/users/compute/orchestration/workload-manager/
[120] http://diracgrid.org

modular architecture designed specifically to allow easy extension for the needs of particular applications.

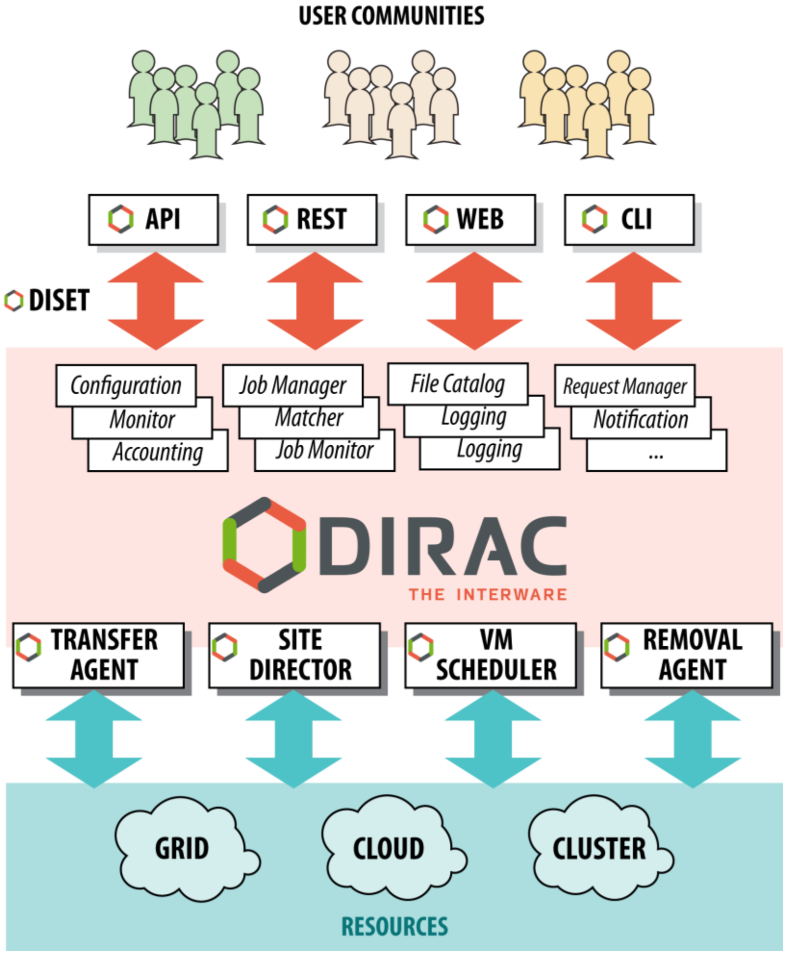
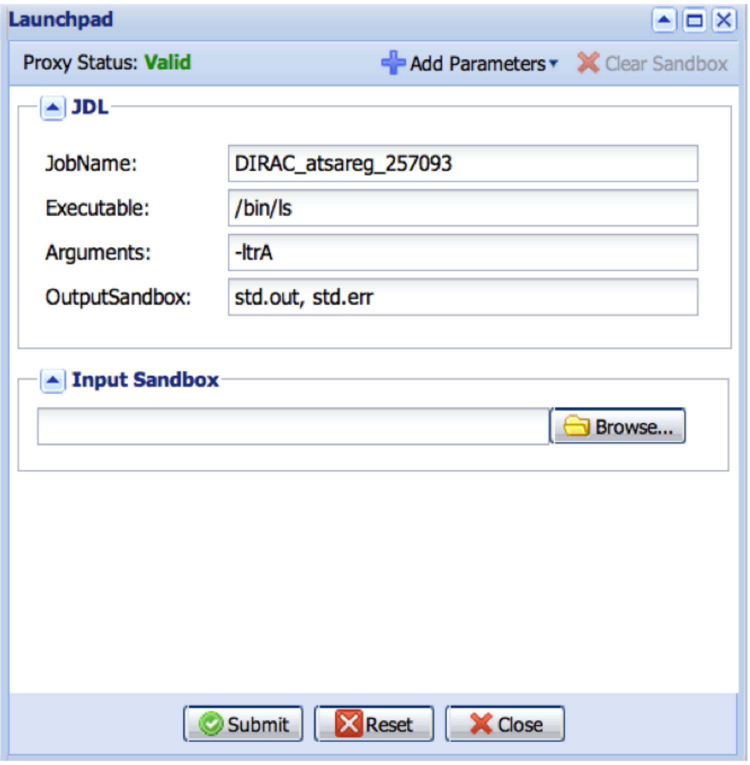


Figure 5.6 - DIRAC Components

User's workload is expressed in the form of jobs, each job representing a unit of work that can be managed by DIRAC. Jobs are described through the Job Definition Language (JDL)[121] and can be easily created with the Job Launchpad included in the web portal (see *Figure 5.7*).

The basic information that every job defines is the executable to use its arguments. Usually its input and output are also specified. When user jobs are submitted to the Workload Manager Service, the service performs reservation of computing resources by means of so-called pilot jobs which are submitted to various computing centres with appropriate access protocols.

[121] https://dirac.readthedocs.io/en/latest/UserGuide/GettingStarted/UserJobs/JDLReference/index.html



Proxy Status: Valid

JobName: DIRAC_atsareg_257093

Executable: /bin/ls

Arguments: -ltrA

OutputSandbox: std.out, std.err

Figure 5.7 - DIRAC Job Launchpad

Once deployed on worker nodes, pilot jobs verify the execution environment and then request user payloads from the central DIRAC Task Queue. Altogether, the pilot jobs, and the central Task Queue, form a dynamic virtual batch system that overcomes the heterogeneity of the underlying computing infrastructures.

6. Other Advanced Services

Researchers and scientists are increasingly challenged with managing and analysing ever growing volumes of data. This requires ever growing compute and storage resources, as well as new approaches to orchestrate compute services or curate, triage, and work with data.

The EOSC Computing Platform, through EGI-ACE, provides advanced services that can meet these modern needs. In this chapter we look at some of these higher-level services, with a particular focus on Machine Learning (ML) and Artificial Intelligence (AI).

6.1. Machine Learning

The Computing Platform provides a collection of specialized services, tools, and libraries for applying machine learning for feature detection and predictive analytics to datasets, the so called DEEP[122] platform.

Machine learning with deep neural network approaches is called Deep Learning. The availability of powerful GPUs on the Computing Platform makes building, training and evaluating large deep neural networks possible.

The DEEP platform[123] offers a complete framework for users, practitioners, and developers of AI, allowing transparent training, sharing, and serving of Machine Learning and Deep Learning (ML/DL) applications, both locally and on hybrid cloud systems.

The provided set of tools and services cover the entire ML/DL development cycle, from model creation, data processing, training, validation and testing, to the sharing and publication of the models. You are able to focus on domain-specific challenges, while the DEEP platform can manage AAI, resource management, marketplace, CI/CD software quality assurance, etc. with assistance of experts from competence centers.

The DEEP platform consists of the following services:

6.1.1. DEEP Open Marketplace

The DEEP Open Marketplace[124] is a curated repository of applications ready to be used or extended, fostering knowledge exchange and re-usability of applications. It aims to serve as the central knowledge hub for ML/DL applications, breaking knowledge barriers across distributed teams.

Pre-configured Docker containers, repository templates, and other related components are included in the DEEP Marketplace. You can not only access, but also contribute to these via the DEEP Marketplace portal, which allows you to easily browse through the available

[122] https://deep-hybrid-datacloud.eu/

[123] Developed in project DEEP-Hybrid-DataCloud and enhanced by services developed in the EOSC-Synergy project

[124] https://marketplace.deep-hybrid-datacloud.eu/

modules, download and interact with them through the DEEP Platform's API (see Section 6.1.4for details).

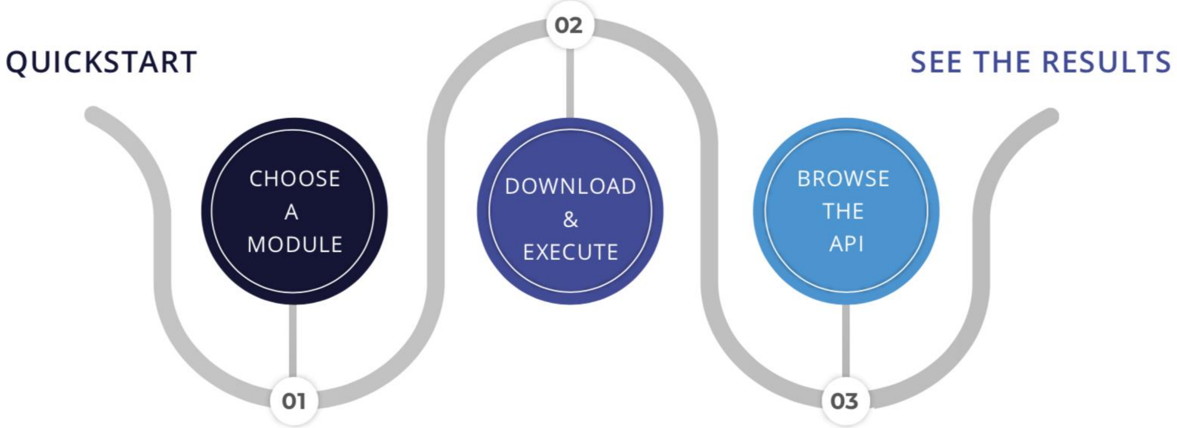


Figure 6.1 - User workflow for interacting with the DEEP Open Marketplace

6.1.2. DEEP Training Facility

The DEEP Training Facility[125] allows you, as a data scientist, to develop and train your models, with access to the latest generation of EU computing e-Infrastructures.

Access to the training facility is facilitated by the DEEP Dashboard, which allows you to interact with the modules hosted in the DEEP Marketplace (see Section 6.1.1 for details) or deploy external workloads, train your modules, and perform inference, while hiding the complexity of making deployments using the DEEP framework to the underlying Cloud, HTC and HPC resources.

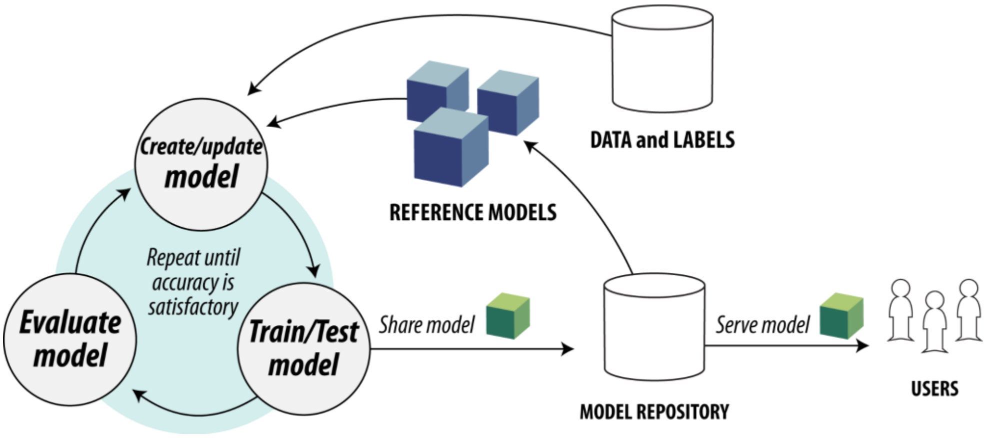


Figure 6.2 - DEEP platform workflow

The Training Facility requires modules and user applications to be containerized and integrated with the DEEP Platform API (see Section 6.1.4for details) in order to expose their functionality. The training system will exploit the existing resources that are integrated with

[125] https://marketplace.eosc-portal.eu/services/deepaas-training-facility

it (OpenStack, Kubernetes, Apache Mesos or HPC systems), through the INDIGO PaaS Orchestrator (see also Section 6.3.3 for details).

6.1.3. DEEP Development Environment

For you to be able to develop your ML/DL models and code, leveraging the DEEP learning frameworks, components and tools, you can take advantage of the development environment based on Jupyter Notebooks or JupyterLab (see also Section 6.2 for details).

6.1.4. DEEPaaS API

For the DEEP platform to be a true PaaS offering, its API is key. The DEEPaaS[126] API makes the DEEP modules accessible even to non-experts, providing a consistent and easy way to access their functionality. It enables you, as a data scientist, to expose your applications through an HTTP endpoint and deliver a common interface for ML, DL and AI applications.

The DEEPaaS API is available for both inference and training. The DEEP platform provides asynchronous training and control to launch, monitor, stop and delete the model training.

6.1.5. DEEP Storage

The official storage service provided by the DEEP platform is Nextcloud[127]. It supports data stored in external storages, via standardized protocols like WebDAV, Amazon S3, or OpenStack Swift (e.g. using the *rclone* tool[128]).

Besides the official storage service, other storage and data services (see Section 4.1 for details) are supported to host data for processing and analytics, such as Ceph[129] or Min.io[130]. Both offer an AWS S3-compatible object store API.

Data repositories such as DSpace and Dataverse, are being validated and used in the framework of EOSC-Synergy project. The repositories can be deployed as a service for user communities via a Repository as a Service (RepaaS). These services are enablers of FAIRness of datasets and can be used by the user communities.

6.1.6. DEEP Authentication and Authorisation

Access to the DEEP Platform (see also Section 2 for details) is provided through the INDIGO Identity and Access Management (IAM)[131], which covers all DEEP facilities and services (DEEP IAM). Further extending the AAI framework of the DEEP Platform to enable authentication using the EGI Check-in service is also planned.

[126] https://github.com/indigo-dc/DEEPaaS
[127] https://nc.deep-hybrid-datacloud.eu/
[128] The complete list of protocols supported by *rclone* is available at https://rclone.org/
[129] https://ceph.io/en/
[130] https://min.io/
[131] https://iam.deep-hybrid-datacloud.eu/login

6.2. Development Environments

One of the goals of the EOSC Computing Platform is to empower researchers to easily perform domain-specific research tasks, without having to deal with the complexities of the underlying infrastructure. An essential tool to achieve this is a flexible development environment, where scientists can create and share documents that contain live code, equations, narrative text and rich media output.

6.2.1. Notebooks

EGI Notebooks[132] is a Web-based, interactive development environment based on the JupyterHub[133] technology running on a cloud provider of EGI-ACE. Its integration with GitHub and Zenodo enables researchers to easily engage with the concept of Open Science.

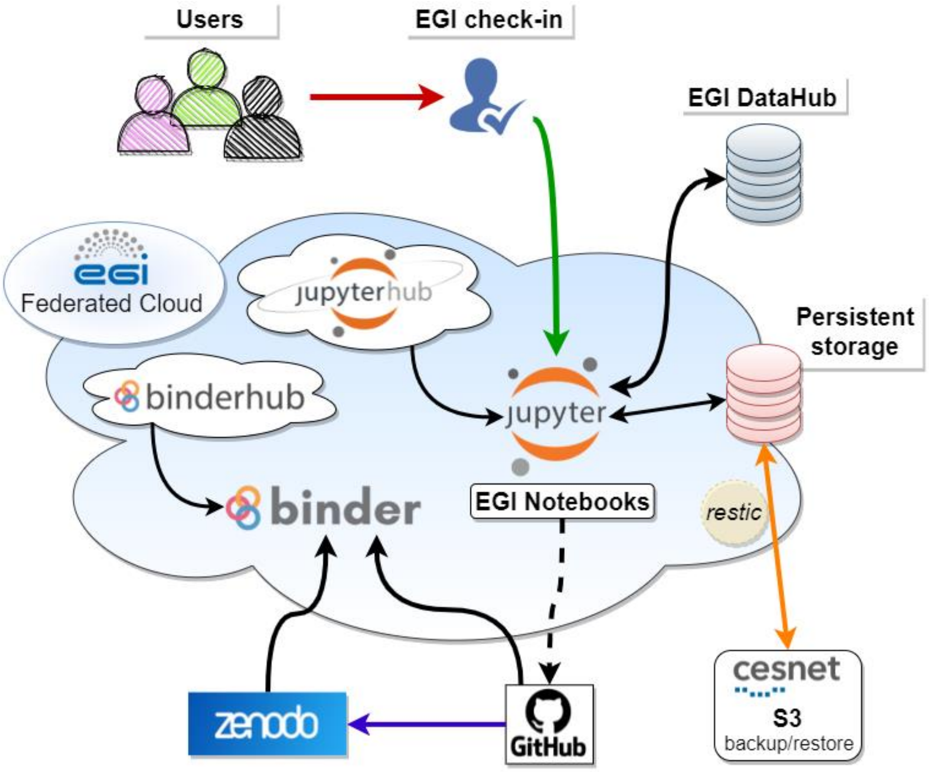


Figure 6.3 - Notebooks and Binder workflows

Notebooks are accessible to individually registered users as well as jointly enrolled virtual organisations, see Section 2 for details. Computing resources are provisioned from participating providers integrated with the EGI Cloud. Basic storage capacity is likewise provisioned from sites in the EGI Cloud, but additional storage can be made available case-by-case from a variety of storage capacity providers (e.g. EGI DataHub, AWS S3-compatible Online Storage - Object Storage, or specialized, area-specific data stores).

[132] https://marketplace.eosc-portal.eu/services/egi-notebooks
[133] https://jupyter.org

Essential notebook images are readily available to all researchers and scientists, but additional images can be made available upon request, with multiple choices for the supported programming language (e.g Python, Julia, R, Octave or MATLAB[134]). EGI Notebooks are backed-up/restored to/from S3 storage provided by CESNET. One-off use of alien images is facilitated through the Binder service (see Section 6.2.2 for more details).

6.2.2. Binder

Binder[135] allows the re-creation of a custom computing environment for reproducible execution of notebooks (and potentially many other types of applications) that can be easily shared with other users with just a link. Users that create their own notebooks in the EGI Notebooks to analyse data available in EGI's infrastructure can easily create a shareable entry in Zenodo from a GitHub repository that anyone can then reproduce in the Binder service.

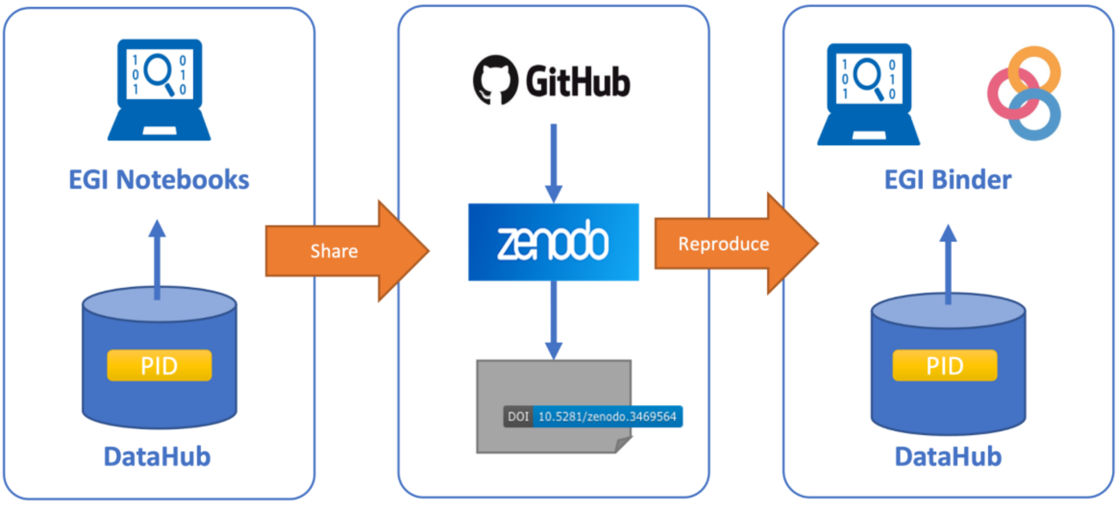


Figure 6.4 - Binder workflow

Binder starts from a code repository that contains the code or notebook you'd like to run and a set of configuration files that specify what's the exact computational environment your code needs to run. Binder builds docker containers on-the-fly following the configuration files, which support specifying conda environments; installing Python, R and Julia environments; installing additional OS packages; and even complete custom Dockerfiles to bring any application to the system. The code repository can be hosted on popular git hosting platforms like GitHub[136] and GitLab[137] and can also be referenced with a DOI from Zenodo[138], FigShare[139] or Dataverse[140].

[134] https://www.mathworks.com/products/matlab.html
[135] https://docs.egi.eu/users/dev-env/binder/
[136] https://github.com
[137] https://gitlab.com
[138] https://zenodo.org/
[139] https://figshare.com/
[140] https://dataverse.org/

EGI's binder offers a similar setup to the publicly accessible mybinder.org[141] service but integrated with the EGI infrastructure and these features:

- Users have a personal access token that can be used to access other EGI services
- Selected spaces of EGI DataHub are directly available under the /datahub folder simplifying the access to shared data
- Environments are guaranteed 2GB of RAM and can reach 4GB as maximum and there are no hard limits on the session time per user, although sessions will be shut down automatically after 1 hour of inactivity. The public mybinder.org service has a 2GB memory limit and six hours of session time with automatic shut down after 10 minutes of inactivity.

User communities can have their customized Binder service instance with extra features. EGI offers consultancy and support, as well as can operate the setup.

6.3. Cloud Orchestrators

One of the challenges for researchers in recent years is to manage an ever-increasing amount of compute and storage services, which then form ever more complex end-user applications or platforms.

To address this need, several cloud-based orchestrators are available that can support the creation of virtual infrastructures on top of IaaS Cloud resources. These tools have different levels of abstractions and features, detailed below. The *Table 6.1* below helps you understand which to choose (or gets used automatically) in specific scenarios.

Table 6.1 - Orchestrators in EGI-ACE

Name	Workload Type	Use Case
Infrastructure Manager	VMs and containers	Used to run workloads on a single IaaS Cloud provider.
Elastic Cloud Computing Cluster	VMs and containers	Used when you need to run workloads on clusters that can be elastically scaled and potentially span more than one IaaS Cloud provider.
PaaS Orchestrator	VMs, containers (both long running and batch), and HTC jobs	Used when you have both IaaS Cloud and HTC workloads. The Machine Learning uses it for ML/DL workloads.
Dynamic On-Demand Analysis Software	Containers	Used when your workload is composed of Docker containers (Helm charts).

[141] https://mybinder.org

These orchestrator services are (planned to be) integrated with the Service Management Tools, allowing federated identity access, monitoring and accounting.

6.3.1. Infrastructure Manager

Infrastructure Manager (IM)[142] is a tool that enables the effective deployment of all the infrastructure necessary for end-user applications or services, either composed by VMs or by Docker containers.

IM orchestrates the deployment of custom virtual infrastructures on multiple back-ends. It automates the deployment, configuration, software installation, monitoring and update of virtual infrastructures. It supports a wide variety of back-ends, either public (such as Amazon Web Services, Google Cloud or Microsoft Azure), on-premises (such as OpenStack) or container-based (such as Kubernetes), thus making user applications cloud agnostic (see *Figure 6.5*).

In order to facilitate deterministic repeatability of virtual infrastructure deployments, IM adopts an Infrastructure-as-Code (IaC) approach. IaC allows defining application architecture using high-level “recipes” and resorts to automated procedures for both the provisioning of virtual computing resources, and the automated configuration of said resources. In addition it features DevOps capabilities based on Ansible[143], enabling the contextualization of the infrastructure at run-time by installing and configuring all the required software that may not be available in the VM images, ensuring researchers will obtain a fully functional virtual infrastructure. The virtual infrastructures can be defined using its native language called Resource and Application Description Language (RADL)[144] or the TOSCA OASIS standard[145].

[142] https://marketplace.eosc-portal.eu/services/infrastructure-manager-im, see also EGI webinar https://indico.egi.eu/event/5495/

[143] https://www.ansible.com

[144] https://imdocs.readthedocs.io/en/latest/radl.html

[145] https://docs.oasis-open.org/tosca/TOSCA-Simple-Profile-YAML/v1.0/TOSCA-Simple-Profile-YAML-v1.0.html

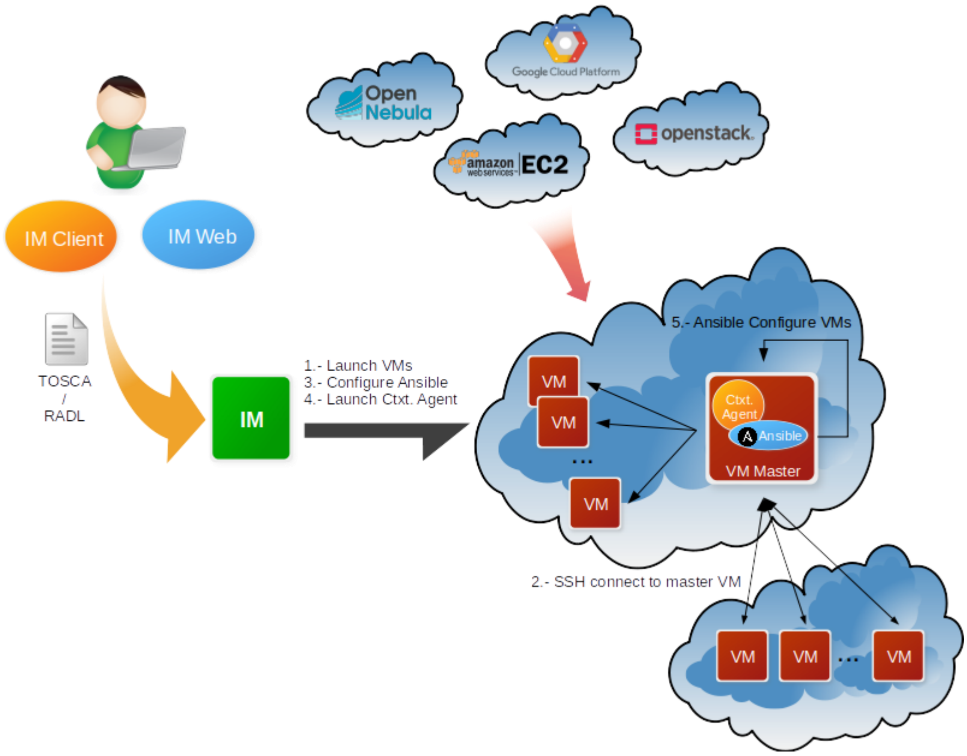


Figure 6.5 - Infrastructure Manager workflow

IM considers these aspects when creates and manages virtual infrastructures:

- The software and hardware requirements specification for the user applications, using a simple language defined to be easy to understand by non-advanced users who just want to deploy a basic virtual infrastructure, but with enough expressivity for advanced users to set all the configuration parameters needed to get the infrastructure fully configured.
- The selection of the most suitable Virtual Machine Images (VMI) based on the user's requirements.
- The provision of VMs on the configured cloud platform and/or Docker container workloads using the configured container orchestrator.
- Support of hybrid infrastructures, where nodes are spread among different cloud providers, enabling cloud bursting scenarios.
- The contextualization of the infrastructure at run-time by installing and configuring all the required software that may not be available in the images (either VMIs or Docker images).
- The elasticity management, both horizontal (adding/removing nodes) and vertical (growing/shrinking the capacity of nodes).

IM provides both XML-RPC and REST APIs to enable high-level components to access its functionality. These APIs allow clients to create, destroy, and inspect virtual infrastructures. IM also provides a command line tool (im-client) and two web interfaces:

- **im-dashboard** can deploy predefined TOSCA templates with a few clicks, and
- **im-web** has advanced functionality for advanced users.

For more information about IM, please check out the documentation[146] and the code.[147]

6.3.2. Elastic Cloud Computing Cluster

Elastic Cloud Computing Cluster (EC3)[148] is a tool to create elastic virtual clusters on top of IaaS providers. Being based on Infrastructure Manager (or IM, detailed in Section 6.3.1), EC3 supports the same wide choices of back-ends.

EGI-ACE offers recipes to deploy TORQUE[149] (optionally with MAUI), SLURM[150], SGE[151], Apache Mesos[152], OSCAR[153], ENES[154], Nomad[155] and Kubernetes[156] clusters that can be self-managed.

EC3 creates elastic cluster-like infrastructures that can automatically scale up or down, depending on demand and the configured policies (e.g. minimum and maximum number of nodes). This creates the illusion of a real cluster without requiring an investment beyond the actual usage, delivering a cost-effective elastic Cluster-as-a-Service on top of an IaaS cloud.

The deployment of the virtual elastic cluster consists of two phases. The first one involves starting a VM to act as the cluster front-end, while the second one involves the automatic management of the cluster size, depending on the workload and the specified policies.

For the first step, the EC3 launcher deploys the front-end using IM. Once the front-end and the elasticity manager (CLUES) have been deployed, the virtual cluster becomes totally autonomous and users can submit jobs, either from the cluster front-end or from an external node that provides job submission capabilities. Researchers will have the perception of a cluster with the number of nodes specified as minimum and/or maximum size.

CLUES will not only monitor the working nodes (to detect the ones that are idle) but will also intercept job submissions and will dynamically manage the cluster size transparently to the user, scaling up and down on demand. It uses IM to create/destroy the VMs that will act as worker nodes for the cluster.

EC3 supports three deployment models:

- **Homogeneous clusters** are composed of working nodes that have the same hardware and software characteristics. This is the default deployment model of EC3, where only one type of node is used, shown in *Figure 6.6*.

[146] https://www.grycap.upv.es/im/ and https://docs.egi.eu/users/compute/orchestration/im/
[147] https://github.com/grycap/im
[148] https://marketplace.eosc-portal.eu/services/elastic-cloud-compute-cluster-ec3
[149] http://docs.adaptivecomputing.com/torque/4-2-10/torqueAdminGuide-4.2.10.pdf
[150] https://slurm.schedmd.com/
[151] http://star.mit.edu/cluster/docs/0.93.3/guides/sge.html
[152] https://mesos.apache.org/
[153] https://docs.oscar.grycap.net/deploy-ec3/
[154] https://marketplace.eosc-portal.eu/services/enes-climate-analytics-service
[155] https://www.nomadproject.io/
[156] https://kubernetes.io/

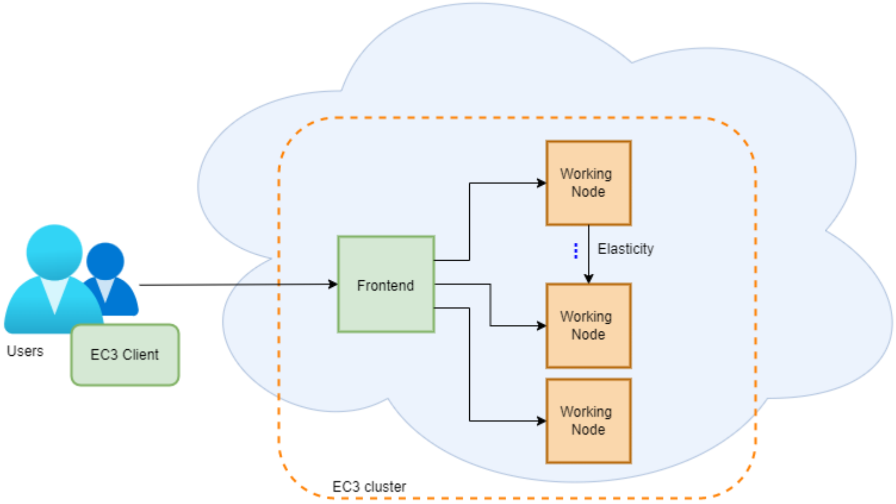


Figure 6.6 - Homogeneous cluster example

- **Heterogeneous clusters** allow the working nodes comprising the cluster to be of different hardware and/or software characteristics. This is of special interest when you need nodes with different configuration or hardware specifications, but all working together in the same cluster. It also allows you to configure several queues and specify to which queue the working node belongs to. This cluster type is depicted in *Figure 6.7*.

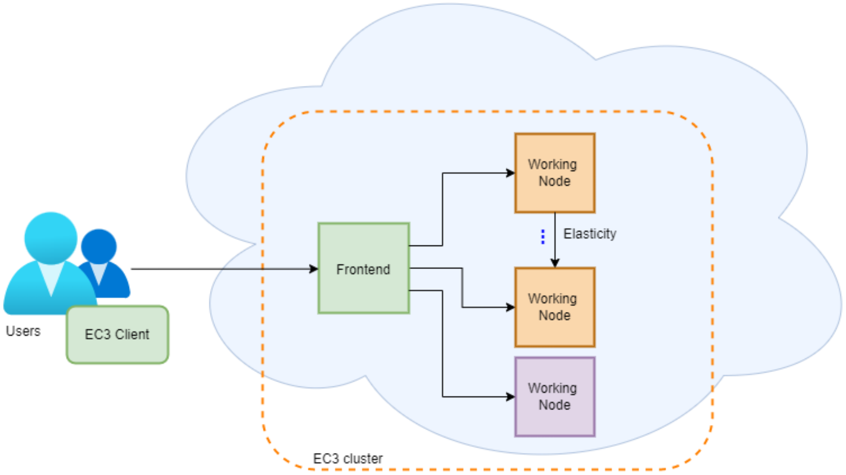


Figure 6.7 - Heterogeneous cluster example

- **Hybrid clusters** (sometimes also referred to as **Cloud bursting**) consist of launching nodes in two or more different cloud providers. This can overcome user quotas or saturated resources. When a limit is reached and no more nodes can be deployed inside the first cloud provider, EC3 will launch new nodes in the next defined cloud provider. The nodes deployed in different cloud providers can be different too, so that heterogeneous clusters with cloud bursting capabilities can be deployed and automatically managed with EC3. The nodes would be automatically interconnected using VPN or SSH tunnelling techniques. This cluster type is depicted in *Figure 6.8*.

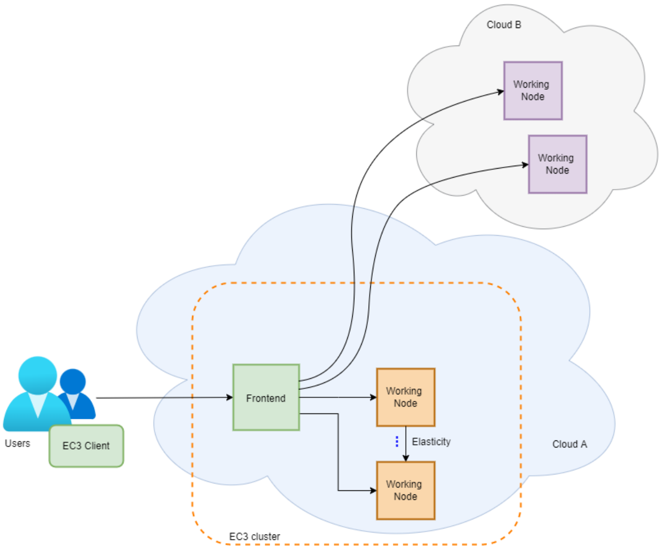


Figure 6.8 - Hybrid cluster example

For more information about EC3, please check out the documentation[157] and the code[158].

6.3.3. PaaS Orchestrator

The PaaS Orchestrator[159] is the core component of the INDIGO PaaS[160], an abstraction and federation layer on top of heterogeneous distributed computing environments, with the following key features:

- Supports provisioning compute and storage resources in IaaS Cloud providers, as well as deploying containerized workloads to container orchestrators and HTC sites
- Supports deployments needing specialized hardware (e.g. GPUs and Infiniband)
- Automatic retries in case of deployment failure or timeout
- Integration with Hashicorp Vault[161] to manage public cloud credentials
- Multi-tenancy support
- Supports multiple identity providers
- Has a REST API, command line, and web interface for managing workloads

The Orchestrator receives deployment requests expressed as TOSCA templates and orchestrates the deployments on the best available cloud sites.

In order to select the best site, the orchestrator implements a complex workflow: it gathers information about the SLAs signed by the providers with the user, the availability of the compute and storage resources, and the location of the data requested by the user (if any).

Once the best site has been identified, the orchestrator starts the deployment workflow through one of its provider plugins:

[157] https://servproject.i3m.upv.es/ec3/ and https://docs.egi.eu/users/compute/orchestration/ec3/
[158] https://github.com/grycap/ec3
[159] https://marketplace.eosc-portal.eu/services/paas-orchestrator
[160] https://link.springer.com/article/10.1007/s10723-018-9453-3, see also EGI webinar https://indico.egi.eu/event/5720/
[161] https://www.hashicorp.com/products/vault

- **The Cloud/IaaS adapter** uses Infrastructure Manager (see Section 6.3.1for details) to provision virtual compute and storage resources on public (Amazon Web Services, Microsoft Azure, Google Cloud Platform) or private (OpenStack) infrastructures.
- **The Apache Mesos connectors** manage the interactions with the relevant cluster framework: *Marathon* for long-running services and *Chronos* for batch-like jobs.
- **The Kubernetes adapter** manages deployments on Kubernetes clusters.
- **The HPC adapter** submits jobs to HPC sites through a QosCosGrid (QCG) Gateway[162].

For more information about PaaS Orchestrator, please check out the documentation[163] and the code[164].

6.3.4. Dynamic On-Demand Analysis Software

Dynamic On-Demand Analysis Software (DODAS)[165] acts as a cloud enabler for scientists seeking to easily exploit distributed and heterogeneous clouds to process, manipulate or generate data. It enables the execution of user analysis code, both in batch mode and interactively (via the Jupyter interface). It is highly customizable and offers several building blocks that can be combined together depending on use case. The currently available blocks allow combining Jupyter, HTCondor, Apache Spark, and more.

DODAS is based on Docker containers and relies on Kubernetes to compose the building blocks via a web-based user interface, thanks to Kubeapps.

To use DODAS to run your code, all you need is a Helm[166] chart to compose your application from available components, which will allow you to take advantage of:

- The INDIGO Identity and Access Management (IAM) that also supports the federated access model (you can login with your institutional credentials)
- Caching and remote I/O optimization via XrootD[167].
- S3-compatible storage access via Min.io[168].
- Software libraries and user configurations via CVMFS.

For more information about DODAS, check out the related publications and presentations[169].

[162] http://www.qoscosgrid.org/trac/qcg
[163] https://github.com/indigo-dc/orchestrator#guides
[164] https://github.com/indigo-dc/orchestrator
[165] https://marketplace.eosc-portal.eu/services/dynamic-on-demand-analysis-service-dodas-portal and https://web.infn.it/dodas/index.php/en/, see also EGI webinar https://indico.egi.eu/event/5695/
[166] https://helm.sh/docs/topics/charts/
[167] https://xrootd.slac.stanford.edu/
[168] https://min.io/
[169] https://web.infn.it/dodas/index.php/en/publications-and-presentations

7. Consultants

In the following section we report the EGI-ACE service providers consulting partners, and the generic supporters, selected to provide first-line of support and technical expertise on how to integrate with the services of the EOSC Compute Platform.

7.1. Generic Support

The first line of support for the EOSC Compute Platform is support@egi.eu.

7.2. Services Supporters of the EOSC Compute Platform

Service	Institute	Contact
AppDB	IASA	William Karageorgos
DataHub	Cyfronet	Lukas Dutka, Bartosz Kryza, Marek Magryś
Workload Manager	CNRS	Ghita Rahal, Gino Marchetti, Andrei Tsaregorodtsev
DEEPaaS	CSIC	Álvaro López García
Accounting	CESGA	Carlos Fernandez
AAI, Monitoring	GRNET	Nicolas Liampotis
DODAS, PaaS Orchestrator	INFN	Giacinto Donvito, Daniele Spiga
Monitoring	SRCE	Emir Imamagic
RUCIO, Data Transfer, APEL, CVMFS	STFC	Ian Collier
EC3/IM	UPV	Miguel Caballer
Notebooks/Binder	CESNET	Zdeněk Šustr