D4.1 - Infrastructure & Services Definition (a)

Work Package	WP 4, EO4EU Data Marketplace Ecosystem
Lead Author (Org)	Lakis Christodoulou, Marios Sophocleous, Philippos Philippou (EBOS)
Contributing Author(s) (Org)	Name (organization) NKUA, ECMWF, NVCR, SIS, CINECA, HES-SO, DANAOS, KEMEA, ENG, IES, MEEO, IVI
Due Date	30.11.2023
Date	30.11.2023
Version	V1.0

Dissemination Level

X	PU: Public
	PP: Restricted to other programme participants (including the Commission)
	RE: Restricted to a group specified by the consortium (including the Commission)
	CO: Confidential, only for members of the consortium (including the Commission)

Disclaimer

Versioning and contribution history

Version	Date	Author	Notes
0.0	15.08.2023	Lakis Christodoulou	TOC
0.1	15.09.2023	Lakis Christodoulou	V0.1
0.2	25.09.2023	Lakis Christodoulou	V0.2
0.3	10.10.2023	Lakis Christodoulou	V0.3
0.4	15.10.2023	Lakis Christodoulou	V0.4
0.5	30.11.2023	Lakis Christodoulou	V0.5
0.6	17.11.2023	Lakis Christodoulou (NKUA: Request for Changing the Structure)	V0.6
0.7	21.11.2023	Lakis Christodoulou Marios Sophocleous Philippos Philippou	V0.6.1
1.0	30.11.2023	Marios Sophocleous, Philippos Philippou, ECMWF	Finalisation of content and restructuring, adding new content in 3.2

Terminology

Terminology/Acronym	Description
API	Application Programming Interface
CFS	Customer Facing Services
CLI	Command Line Interface
CMCC	FONDAZIONE CENTRO EURO-MEDITERRANEOSUI CAMBIAMENTI CLIMATICI
CNC	CINECA CONSORZIO INTERUNIVERSITARIO
CSA	Coordination and Support Action
DANAOS	DANAOS SHIPPING COMPANY
DB	Dashboard
DoA	Description of Action
EBOS	EBOS TECHNOLOGIES LIMITED
EC	European Commission
ECMWF	EUROPEAN CENTRE FOR MEDIUM-RANGE WEATHER FORECASTS
ENG	ENGINEERING - INGEGNERIA INFORMATICA SPA
EO4EU	Horizon Europe project called: AI-augmented ecosystem for Earth Observation data accessibility with Extended reality User Interfaces for Service and data exploitation
FMI	ILMATIETEEN LAITOS
GA	Grant Agreement to the project
HPC	High Performance Computing
HTTP	Hypertext Transfer Protocol
IES	INTELLIGENCE FOR ENVIRONMENT AND SECURITY SRL IES SOLUTIONS SRL
IPM	Integration Planning Methodology
IVI	FRAUNHOFER GESELLSCHAFT ZUR FORDERUNG DER ANGEWANDTEN FORSCHUNG EV
KC	Key Cloak
KEMEA	KENTRO MELETON ASFALEIAS
LU	LATVIJAS UNIVERSITATE
MEEO	METEOROLOGICAL AND ENVIRONMENTAL EARTH OBSERVATION SRL
NKUA	National and Kapodistrian University of Athens
NVCR	NOVELCORE
SIS	SISTEMA GMBH
TRUST-IT	TRUST-IT SRL

Terminology/Acronym	Description
UDI	User Data Interface
UMM	User Management Model
VU	VILNIAUS UNIVERSITETAS

Table of Contents

List of Figures

List of Tables

Executive Summary

This document is entitled “D4.1 – Infrastructure & Services Definition” and represents the first deliverable of Work Package 4 of the EO4EU project.

This document focuses on the following outcomes:

- List the services and components of the EO4EU platform and their respective purpose.
- Provide an initial snapshot of the infrastructure requirements for the platform’s components & services.
- Provide an overview of the available infrastructure to be utilized for hosting the EO4EU platform.
- Explain the methodology on how the multi-cloud infrastructure can be achieved encompassing all the available infrastructure resources.
- State on which infrastructure each component or group (Tiers) of components/services will be hosted and the reasoning behind that choice.
- Provide a summarized version of infrastructure requirements taking into account the overall needs of all the components collectively.

The 3 main infrastructures available are the WEkEO, CINECA Cloud and CINECA High Performance Computer (HPC). WEkEO infrastructure will host the Platform Controller, CINECA Cloud will host all other components utilizing a Kubernetes Multi-Cluster and the Machine Learning/Inference Server service will be hosted by CINECA HPC, allowing for the best possible performance for ML model trainings. The specific computational, memory, storage and Graphics Processing Unit capabilities are summaries at the end of section 4.

1 Introduction

This document marks the initial release of two deliverable sets, intended to fulfill the requirements outlined in Task T4.1, specifically focusing on 'Integration Planning & Service Provision Specification.' Its primary purpose is to establish the groundwork for configuring and deploying multiple service components while defining the necessary infrastructure. The key objectives of this report are to precisely outline the specifications for both cloud (WEkEO & CINECA) and CINECA High Performance Computer (HPC) infrastructure. This delineation is crucial to support flexible and versatile data processing methods, aligning with the project's overarching vision. The aim is to improve accessibility to data sources and enhance the usability of environmental observation information. This improvement is achieved by furnishing technological tools, services, and concepts. The report achieves these objectives by incorporating insights from the structural and technical components identified in Work Packages (WPs) 2 & 3. Additionally, it conducts a thorough analysis and assessment of user requirements and system specifications.

1.1 Purpose of the Document

Deliverable 4.1 serves as a comprehensive technical analysis and description of the infrastructure and services required for the EO4EU Platform. It analyses the requirements, specifications, and technical components necessary for the deployment and integration of various software components under different environments, including cloud networks, local networks, and multi-domain setups and delivers the infrastructural elements to accommodate the deployment of a holistic ecosystem enhancing data access. This deliverable acknowledges the contributions of previous work packages, particularly WP2 & 3, which focused on analysing user requirements, system specifications and the development of the actual components and services.

Moreover, the primary objective of Deliverable 4.1 is to prioritize and finalize the system architecture, interfaces, and integration demands based on the components & services requirements. It aims to establish a list of performance metrics, including scalability and significance, to efficiently and effectively operate the EO4EU framework.

Finally, the deliverable is closely tied to WP5, which focuses on the deployment and integration of various use cases with the EO4EU Platform system. This linkage is critical for understanding and defining the specific infrastructure requirements and service provisions for each use case, ultimately leading to enhancements and optimizations of the overall system architecture and infrastructure.

1.2 Relation to other Activities and Deliverables

Deliverable 4.1 is instrumental in addressing the intricacies of the EO4EU Infrastructure and Services, establishing a direct nexus with 'WP2 – Requirements Elicitation and Conceptual Framework Specification-Task T2.3 Technical Specifications, Interoperability Requirements, and Scalability Analysis.' Within this work package and task, there is a meticulous definition and analysis of the System Architecture of the Software Components, encompassing both their functional and technical requirements.

As we delve into the analysis and extraction of end-user requirements and system features in WP2, a simultaneous evaluation of the technical requirements and system specifications of the EO4EU platform unfolds. This dual examination sheds light on the operational and technical necessities delineated in D4.1, aligning seamlessly with the ongoing developmental and integrative phases of the project lifecycle. Concurrently, 'WP3 – Data Orchestration & Machine Learning - T3.2 Systems and Services Orchestration' plays a pivotal role in complementing the needs and requirements of all Software Components. These components are slated for deployment on competitive cloud infrastructure. Tasks T3.1, T3.2, T3.3, T3.5, and their corresponding Deliverables D3.1-D3.5 in WP3

contribute directly to D4.1, providing a wealth of technical requirements and provisions for services. This information is crucial in designing, developing, and deploying on the appropriate hardware and software core infrastructure to meet high-end requirements.

Simultaneously, there unfolds a continuous deployment of EO4EU Software Components, evaluating software operations, functionalities, applications, and outcome services. This involves leveraging cutting-edge technologies such as the KUBERNETES container orchestration system for automating software deployment and management, coupled with the utilization of the KAFKA distributed event streaming platform. Rigorous performance testing and evaluation of software responses and data communications, spurred by end-user requests, lead to a comprehensive investigation and assessment. The aim is to identify the requisite infrastructure and software services for the seamless deployment of the EO4EU platform system. An additional pivotal consideration involves a meticulous analysis and interpretation of parameters, such as access and streaming of multiple EO Data Sources, multi-processing of data, computing power assessment, analysis of HPC servers' architecture and infrastructure, scrutiny of cluster/cloud infrastructure, and evaluation of communication framework requirements, alongside a detailed examination of data storage and organizational needs.

Deliverable 4.1 is intricately entwined with 'WP5 – EO Data Uptake Demonstration of the EO4EU frameworks.' This particular work package is dedicated to deploying and integrating various use cases within the EO4EU Platform system. As each use case undergoes deployment and testing, a nuanced comprehension and definition of the infrastructure requirements and service provisions for each become imperative. These insights contribute significantly to the generation of new technical perspectives, ultimately enhancing and optimizing the overall system architecture and infrastructure of the EO4EU Platform system.

1.3 Deliverable Overview and Report Structure

The following provides a summary of the Deliverable in Chapters and gives the corresponding Report Structure, such as:

Chapter 1: Introduction

- Purpose of the Document: This section elucidates the primary objectives and intentions behind the creation of this document, providing a foundational understanding of its scope.
- Relation to other Activities and Deliverables: Here, we establish the document's interconnectedness with various activities and deliverables within the broader context of the EO4EU project, offering insights into its significance.
- Deliverable Overview and Report Structure: This subsection provides an overview of the entire deliverable, offering a roadmap for readers to navigate through the subsequent sections and understand the hierarchical structure of the report.

Chapter 2: Services Definitions & Requirements

This section serves as the core of the document, delving into the detailed definitions and requirements of the services within the EO4EU platform. It encompasses various tiers, including the Data Tier, Infrastructure as a Service (IaaS) Tier, Platform as a Service (PaaS) Tier, Machine Learning Tier, and Front-End Tier.

Chapter 3: Infrastructure Specifications

This section provides a comprehensive overview of the infrastructure specifications associated with the EO4EU project. It delves into the specifics of the WEKEO Infrastructure, CINECA Cloud Infrastructure, and CINECA High-Performance Computer (HPC) Infrastructure. Additionally, it outlines the allocation of components and services on the infrastructure.

- 3.4 Allocation of Components & Services on Infrastructure: This subsection focuses specifically on the integration planning and tier distribution towards the design and development of the required infrastructure and services. It provides a strategic perspective on how different tiers are distributed across available infrastructure.

Chapter 4: Summary of Infrastructure & Services Specifications

This section consolidates and summarizes the key specifications related to both infrastructure and services, providing a concise reference point for readers.

Chapter 5: Conclusion

The document concludes by summarizing the key findings, insights, and implications derived from the exploration of services definitions, requirements, and infrastructure specifications.

2 Services Definitions & Requirements

This chapter provides a high-level description of the EO4EU platform components and services based on the architecture established in D2.4. It continues to state the infrastructural needs of each component and service in order to properly plan the deployment of each one of the components on the available infrastructure ensuring the maximum possible performance.

2.1 EO4EU Platform Architecture

EO4EU architecture consists of five different tiers, as seen in Figure 1. The 5 different Tiers are the Data Tier, the Infrastructure as a Service (IaaS) Tier, the Platform as a Service (PaaS) Tier, the Machine Learning Tier and the Front-End Tier.

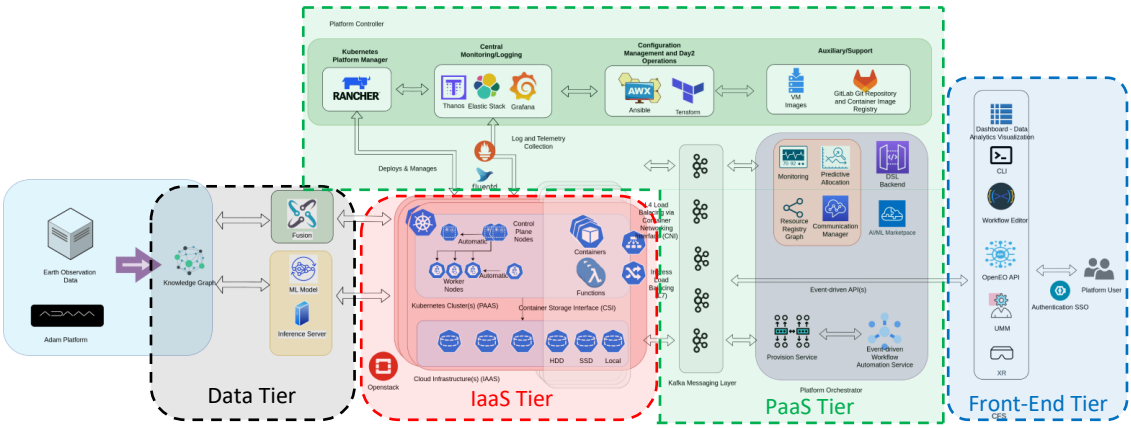

Figure 1. EO4EU Platform System Architecture

2.2 Data Tier

The platform takes input from a collection of data sources containing diverse data requiring pre-processing. To accomplish this task, EO4EU employs a Knowledge Graph, a Fusion engine, and an ML component at this tier to enhance and standardize the data.

2.2.1 Knowledge Graph

The Knowledge Graph (KG) serves the purpose of enhancing users' search capabilities, enabling access to Earth Observation (EO) data through semantic search. It supports free-text queries, allowing users to discover and utilize EO data even without expertise in the field. The KG organizes and processes datasets from various sources, such as Copernicus Services and third-party platforms like ADAM. By gathering textual descriptions (metadata) through embeddings, it provides a structured and unified approach to accessing diverse EO datasets. Its user-friendly design aims to make EO data accessible to non-experts, allowing them to work with previously unknown and undiscoverable resources. The KG's semantic processing of metadata enhances the accuracy and relevance of search results, making it easier for users to find the information they need. Through the transformation of user queries or automated requests into internal representations (vectors), the KG identifies similarities with dataset vectors. This efficient matching process results in a targeted list of relevant results, streamlining the data discovery process for users.

The minimum infrastructural needs to effectively run the KG are shown in Table 1 below.

Table 1. Infrastructure Needs for KG.

Specification	Minimum Values
Computational Needs	4 CPUs
Memory Needs	64 GB RAM
Storage Needs	100 GB
GPU Needs	Low

2.2.2 Data Fusion Service

The Data Fusion Engine serves to enhance context awareness by amalgamating data readings and achieving situation awareness. It operates through two main functions: firstly, creating fusion models and pipelines for spatiotemporal data aggregation, and secondly, dynamically executing multiple workflows in parallel in response to user requests. Fusion pipelines become accessible to users through publication on the AI/Marketplace and are utilized within the Workflow Execution Environment (WFE). When a new workflow is initiated, the Fusion Proxy is engaged to create a specific workflow within the Kubeflow environment. This involves establishing a consumer, triggering execution, initializing the necessary environment, running the pipeline, and subsequently publishing results in the data repository. The following component is informed via Kafka. Computation tasks are carefully managed to be performed with a focus on high productivity, utilizing High-Performance Computing (HPC) or GPU environments as necessary. Each pipeline executes within the Dockerized Kubeflow environment and communicates results to subsequent components through the EO4EU message bus.

The minimum infrastructural needs to effectively run the Data Fusion Engine are shown in Table 2 below.

Table 2. Infrastructure needs for Data Fusion Engine.

Specification	Minimum Values
Computational Needs	4 CPUs
Memory Needs	16 GB RAM
Storage Needs	100 GB
GPU Needs	Medium

2.2.3 Machine Learning/Inference Server Service

The inference server component comprises two main sub-components. The initial sub-component is the core inference server, employing KServe as its foundation. KServe, a standard Model Inference Platform on Kubernetes, is specifically designed to facilitate highly scalable use cases. It establishes a performant and standardized inference protocol across multiple machine learning frameworks, supporting modern serverless inference workloads with Autoscaling, including the capability to Scale to Zero, with or without GPUs. Models are uploaded to storage, currently S3, in a format compatible with the supported inference server backend, which is presently NVIDIA Triton. Triton supports various model formats, including ONNX, the format currently used—an open-source project defining an interoperable format for machine learning models. The KServe component exposes a RESTful API for inference, utilizing JSON as the query language.

The second sub-component interfaces with the EO4EU platform. This sub-component receives messages from the platform, reads data in various formats, and transforms them into the format required by the internal Inference server. It also converts the results into a format expected by the platform. This component incorporates a converter from Kafka to HTTP and a webserver code that interacts with the inference server.

The minimum infrastructural needs to effectively run the ML/Inference Server Service are shown in Table 3 below.

Table 3. Infrastructure needs for ML/Inference Service.

Specification	Minimum Values
Computational Needs	8 CPUs
Memory Needs	64 GB RAM
Storage Needs	200 GB
GPU Needs	Very high

2.3 Infrastructure as a Service (IaaS) Tier

IaaS provides is responsible to provide the necessary abstraction of the infrastructure, allowing the Platform as a Service (PaaS) components to efficiently consume compute, storage and networking resources dynamically and programmatically via the Openstack REST APIs.

2.3.1 Openstack

OpenStack is a cloud computing platform with open-source architecture, offering Infrastructure as a Service (IaaS) capabilities. It provides essential cloud infrastructure services, including computing, storage (both block and object), and networking. Serving as the foundation for higher-level services like Kubernetes and the Platform as a Service (PaaS) tier, OpenStack is a flexible and versatile option for constructing and overseeing cloud environments.

The minimum infrastructural needs to effectively run the Openstack are shown in Table 4 below. However, as the number of applications increases on Kubernetes, the needs will increase accordingly.

Table 4. Infrastructure needs for Openstack.

Specification	Minimum Values
Computational Needs	2 CPUs
Memory Needs	8 GB RAM
Storage Needs	40 GB
GPU Needs	None

2.4 Platform as a Service (PaaS) Tier

PaaS is responsible to provide a higher-level cloud computing service compared to IaaS. PaaS builds upon the foundational infrastructure offered by IaaS and provides additional services and tools to simplify application development, deployment, and management.

2.4.1 Platform Controller

The platform controller encompasses all the required components and services in order to properly control and run the overall platform ranging from the Kubernetes multi-cluster, to monitoring the available resources to simplifying the deployment of applications on the platform.

The minimum infrastructural needs to effectively run the Platform Controller are shown in Table 5 below.

Table 5. Infrastructure needs for Platform Controller.

Specification	Minimum Values
Computational Needs	4 CPUs
Memory Needs	32 GB RAM

Storage Needs	60 GB
GPU Needs	None

2.4.2 Platform Orchestrator

The Platform Orchestrator, at the core of the system, consists of five sub-components. These include the Monitoring Manager, which handles events for both platform and workflow evolution; the Communication Manager, responsible for managing messaging systems (such as Kafka, Nats) and ephemeral storage; Predictive Allocation, ensuring resource resilience in live workflows; Registry Handler, facilitating interaction with the resource and service registry; and Provision Manager, overseeing the creation and lifecycle of user-defined workflows.

The minimum infrastructural needs to effectively run the Platform Orchestrator are shown in Table 6 below.

Table 6. Infrastructure needs for Platform Orchestrator.

Specification	Minimum Values
Computational Needs	4 CPUs
Memory Needs	32 GB RAM
Storage Needs	80 GB
GPU Needs	None

2.4.3 AI/ML Marketplace

The AI/ML Marketplace is a collaborative open-source repository housing AI/ML processing algorithms, metadata, and data model structures contributed by project partners. It functions as an assistive library within the Dashboard, showcasing algorithms and models for reuse. This library, not a standalone software, includes configuration files, code, and documentation, promoting automated model reuse and data processing acceleration throughout the project. As a European Open Science Cloud (EOSC) portal, the AI/ML Marketplace provides direct access to users and partners for sharing and reusing models, fostering innovation. Zenodo is used for open-source publishing of results and documents. Platforms like Acumos and OpenML enhance interoperability and reusability. The AI/ML Marketplace communicates with the Function Execution Engine based on user demands, services, and system communication needs.

The minimum infrastructural needs to effectively run the AI/ML Marketplace are shown in Table 7 below.

Table 7. Infrastructure needs for AI/ML Marketplace.

Specification	Minimum Values
Computational Needs	1 CPUs
Memory Needs	4 GB RAM
Storage Needs	40 GB
GPU Needs	None

2.5 Machine Learning Tier

The ML tier provides all machine learning models in a toolbox for the post processing of the retrieved or fused data. The ML Tier is essentially decentralised in several other components or services of the platform hence, the infrastructure needs for this Tier are incorporated in the relevant components.

2.6 Front-End Tier

The Front-End Tier is responsible to provide multi-dimensional User Interface-UI (Web, XR, CLI, API) that enables the user to interact and control the platform.

2.6.1 Online Portal/Data Analytics Visualisation

The purpose of the Data Analytics Visualization tool is to provide accessible and advanced visualizations that support decision-making and policy development. It enhances understanding through advanced analytics, facilitates interaction with AI/ML modules for intelligent data generation, and employs a novel situated analytics approach with augmented reality for immersive exploration of Earth Observation (EO) data. Tailored for both scientific and EU civilian users seeking real-time analytics, the tool assists computational scientists in comprehending research data. Its hybrid visualization mechanism merges exploratory and explanatory approaches, improving data exploration and insight generation. Additionally, the tool employs intelligent ML algorithms and visualization methods to trigger smart events related to weather, climate, and environmental changes, fostering an intelligent awareness among citizens. Overall, it serves as a versatile and user-friendly platform for exploring diverse datasets, particularly focusing on environmental observations and EO data.

The minimum infrastructural needs to effectively run the Online Portal/Data Analytics Visualisation are shown in Table 8 below.

Table 8. Infrastructure needs for the Online Portal/Data Analytics Visualisation.

Specification	Minimum Values
Computational Needs	2 CPUs
Memory Needs	16 GB RAM
Storage Needs	80 GB
GPU Needs	None

2.6.2 DSL Engine/Workflow Editor

The Domain-Specific Language (DSL) Engine serves a dual purpose: it acts as a validation and control mechanism for System Workflows and supports the development of a specific language tailored for the Workflow Editor (WFE), known as Graph Description Language (GDL). This language encapsulates details about system nodes, their attributes, metadata, and relationships. Operating as a standalone component, the DSL Engine tightly integrates with the WFE through the WFE Auxiliary (AUX) Service. When a workflow is ready for deployment, the DSL Engine conducts compilation and validation processes. Valid workflows are then compiled into YAML format and sent back to the AUX Service for deployment to systems. In the case of invalid workflows, error reports are generated and communicated to the AUX Service for appropriate handling. In essence, the DSL Engine ensures the accuracy and adherence of System Workflows to predefined rules while supporting the seamless integration and deployment of workflows within the system.

The minimum infrastructural needs to effectively run the DSL Engine/Workflow Editor are shown in Table 9 below.

Table 9. Infrastructure needs for the DSL Engine/Workflow Editor.

Specification	Minimum Values
Computational Needs	4 CPUs
Memory Needs	32 GB RAM
Storage Needs	80 GB
GPU Needs	None

2.6.3 Authentication Single Sign-On/ User Management Module

The Single Sign-On (SSO) functionality streamlines user access by enabling them to sign in once for multiple applications in the EO4EU Software Platform. It simplifies identification and authentication, allowing users to access various services with a single set of credentials. Integrated into the platform, SSO enhances user convenience and security, leveraging a versatile cloud framework with AI/ML functionalities. The UMM is an identity and access management tool that provides user federation, strong authentication, user management, fine-grained authorization, single-sign-on and connection to external identity providers.

The minimum infrastructural needs to effectively run the DSL Engine/Workflow Editor are shown in Table 10 below.

Table 10. Infrastructure needs for the Authentication SSO/UMM.

Specification	Minimum Values
Computational Needs	1 CPUs
Memory Needs	32 GB RAM
Storage Needs	40 GB
GPU Needs	None

2.6.4 Extended/Virtual Reality

The XR (Extended Reality) system enhances the visualization and exploration of Earth Observation (EO) data through a web-based interface. It includes an Augmented Reality (AR) component for real-world EO data analysis and a Virtual Reality (VR) component for 3D visualization on VR-supported hardware. Optimized for EO data, both interfaces offer tools for user interaction, fostering a better understanding of environmental observations. The XR system utilizes the Web XR device API, facilitating seamless communication with other EO4EU applications for obtaining processed EO data. Overall, it aims to provide a concise and immersive experience for users exploring EO data in augmented and virtual realities.

The minimum infrastructural needs to effectively run the XR/VR component are shown in Table 11 below.

Table 11. Infrastructure needs for the XR/VR component.

Specification	Minimum Values
Computational Needs	4 CPUs
Memory Needs	32 GB RAM
Storage Needs	100 GB
GPU Needs	Medium

2.6.5 OpenEO API

The OpenEO API is designed as a control interface management tool, facilitating communication between users and various EO4EU software components. Its purpose is to establish intelligent data interfaces and enable smart communication among different software modules. The API defines user requirements, requests services, and aligns with system component specifications to generate the requested user data. It serves as a conduit for smart communication between EO4EU applications and software functions, functioning as a remote communication server that receives user requests and sends EO data in the requested format. Additionally, the API communicates with the Kubernetes Platform Software Management and other Control Software modules, connecting to retrieve real-time and continuous data from the requested cluster or available servers and software resources for processing EO sources' data.

The minimum infrastructural needs to effectively run the OpenEO API are shown in Table 12 below.

Table 12. Infrastructure needs for the OpenEO API.

Specification	Minimum Values
Computational Needs	2 CPUs
Memory Needs	32 GB RAM
Storage Needs	80 GB
GPU Needs	None

3 Infrastructure Specifications

There are 3 main infrastructure units that will be utilised to host the overall EO4EU Platform. Each one plays a distinctive role in ensuring a seamless and high-performing ecosystem: the WEKEO Infrastructure, CINECA Cloud Infrastructure, and HPC Infrastructure. By strategically integrating these infrastructural components, the EO4EU Platform System can position itself as an open EO data processing and visualization system. The synergy between these infrastructures allows for the seamless provision of services, meeting the diverse requirements inherent in the handling and analysis of EO data on a significant scale. This harmonious integration ensures optimal efficiency and efficacy, making the EO4EU Platform System a powerful and accessible resource for the scientific community and stakeholders involved in Earth Observation.

3.1 WEKEO Infrastructure

WekEO, part of the Copernicus Data and Information Access Services (DIAS), operates on a robust cloud infrastructure that, consists of high-performance servers with multi-core CPUs, extensive storage solutions for data handling, and high-speed networking for efficient data dissemination. Load balancers are also a key component of the WekEO, tasked with ensuring optimal distribution of network traffic to maintain service stability. This scalable infrastructure is designed to support the vast and growing demands of Earth observation data processing and storage, providing a seamless service to end-users without exposing the complexities of the underlying hardware.

Hardware Advantages:

- Robust Cloud Infrastructure: WEkEO, integrated into the Copernicus Data and Information Access Services (DIAS), operates on a formidable cloud infrastructure.
- High-Performance Servers: The infrastructure is equipped with high-performance servers featuring multi-core CPUs. This design ensures the computational power necessary for handling complex tasks related to Earth observation data processing.

3.1.1 Infrastructure as a Service

WekEO, as a part of its cloud offerings utilizes OpenStack that is set of software tools for building and managing cloud computing platforms. OpenStack provides essential infrastructure as a service (IaaS) functionality, such as processing, storage, and networking resources through a dashboard that is accessible by both users and administrators.
Within WekEO, OpenStack enables the orchestration of virtual machines, object and block storage systems, and facilitates the virtual networking capabilities required for the efficient distribution and analysis of Earth observation data. Its services include but are not limited to, Nova for compute, Swift for object storage, Cinder for block storage, and Neutron for networking, ensuring a flexible and scalable environment tailored for data-intensive tasks inherent in satellite data processing and geospatial analysis.
WEkEO, leverages OpenStack utilization as a fundamental component of its cloud offerings. OpenStack, a comprehensive set of software tools, is employed for the construction and management of cloud computing platforms. This choice reflects a commitment to industry-standard, open-source solutions, fostering flexibility and interoperability.

3.1.2 Resource Allocations

In the envisioned multi-cloud system, ECMWF contributes with WekEO cloud infrastructure. Following a thorough budgetary discussion with the parties, a preliminary distribution of resources was determined as outlined below. The provision of additional resources will be evaluated in due time if needed, following the alpha-release of the platform.

- **Instances:** 100 Virtual Machines

- **vCPUs:** 384 Cores
- **Memory:** 768 GB
- **Volume:** 300 Volume
- **Volume Storage:** 64 TB

The following are the summarized resource allocations to be provided for the EO4EU Platform System.

- Multi-Cloud System: WEkEO is envisioned as part of a broader multi-cloud system, and in this context, the European Centre for Medium-Range Weather Forecasts (ECMWF) contributes to the WEkEO cloud infrastructure.
- ECMWF Contribution: The contribution from ECMWF to the multi-cloud system signifies a collaborative approach, where diverse entities pool resources. This collaborative effort enhances the overall capabilities of the WEkEO cloud infrastructure, allowing for optimized resource allocations and improved efficiency.

In summary, the WEkEO cloud infrastructure is designed with a focus on robust hardware, leveraging high-performance servers and the flexibility of OpenStack. Its integration into a multi-cloud system with contributions from ECMWF reflects a strategic and collaborative approach, underscoring the commitment to creating a powerful, versatile, and cooperative environment for Earth observation data and information access services.

3.2 CINECA Cloud Infrastructure

The ADA Cloud infrastructure (co-funded by the European ICEI project) is a Tier-1 system available for scientific research that integrates and completes the HPC ecosystem. It provides both high performance and high flexibility computing in a tightly integrated infrastructure (Figure 1). The flexibility of the cloud helps to better adapt to the diversity of user workloads, while still providing high-end computing power. The other world-class HPC systems (GALILEO100, Leonardo) can be integrated into the workflow as the need for computing tasks increases or scales beyond the ADA cloud provisions. For example, data can be stored on dedicated areas (known as DRES) that can be accessed by all of the HPC systems.

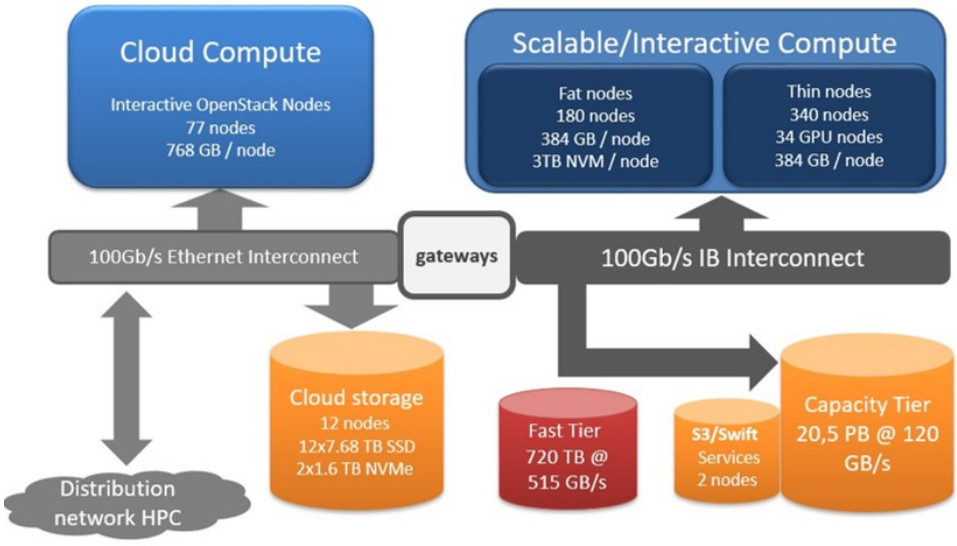

Figure 2. Schematic representation of HPC cloud infrastructure.

3.2.1 Services and System Architecture

ADA Cloud infrastructure is based on OpenStack Wallaby and offers a range of services including the following:

- Core services: Keystone, Nova, Cinder, Neutron
- Volume encryption (Barbican): Self provisioning or managed through dashboard or CLI
- File system sharing (Manila): Self provisioning or managed through dashboard or CLI
- Container orchestration (Magnum): Self provisioning or managed through dashboard
- Others: Load balancing (Octavia), Orchestrator (Heat) and DB service (Trove).

The System Architecture includes the following features:

- 71 interactive OpenStack nodes
- Each node has 2 x CPU Intel CascadeLake 8260 each with 24 cores of 2.4 GHz, 768 GB RAM and 2 TB SSD storage. With HT each nodes has 96 vcpus.
- High-bandwidh low-latency 100 Gb/s ethernet network interconnection
- Cloud storage is based on CEPH version 15.2.13 and 12 servers equipped with 1PB of total raw dis (ful SSD/NVMe)

This cloud infrastructure is tightly coupled to the 20PB raw LUSTRE and 6PB GSS storage seen by all other infrastructures. This setup allows the use of all available CINECA HPC systems (Tier-0 Marconi, Tier-1 Galileo100) and enables HPC workloads to be addressed in conjunction with the cloud resources.

3.2.2 Cloud Service Providers

Cloud service providers offer different models for infrastructure provision (D2.1, Table8). Public clouds, for example, are open to everyone, while private clouds are used by a single organisation. A community cloud is a computing environment shared by organisations with similar goals, requirements, policies and compliance. Community clouds are designed for specific communities that share a common set of concerns, such as regulatory compliance, security standards or industry-specific requirements. Organisations within the community cloud benefit from shared infrastructure and resources, while maintaining a degree of isolation and privacy. From a user perspective, the ADA cloud can be positioned as both a public cloud and a community cloud, with a consortium of European data centres providing capabilities for specific scientific communities (such as the flagship Human Brain project).

3.2.3 Services Deployment Model IaaS

The ADA Cloud offers users an Infrastructure as a Service (IaaS) model. IaaS is a cloud computing model that delivers virtualised computing resources over the Internet. Users can rent these resources from a cloud service provider rather than owning and managing physical servers, storage and networking components. IaaS allows users to scale up and down their infrastructure as needed without having to worry about hardware maintenance, providing flexibility and scalability. Alongside the benefits of flexibility, this places additional responsibility on the user, particularly for auto-provisioning projects. Nevertheless, specific support (both technical and application) is provided to help users in establishing their application workflows. The different layers that make up the IaaS model are shown in the diagram below.

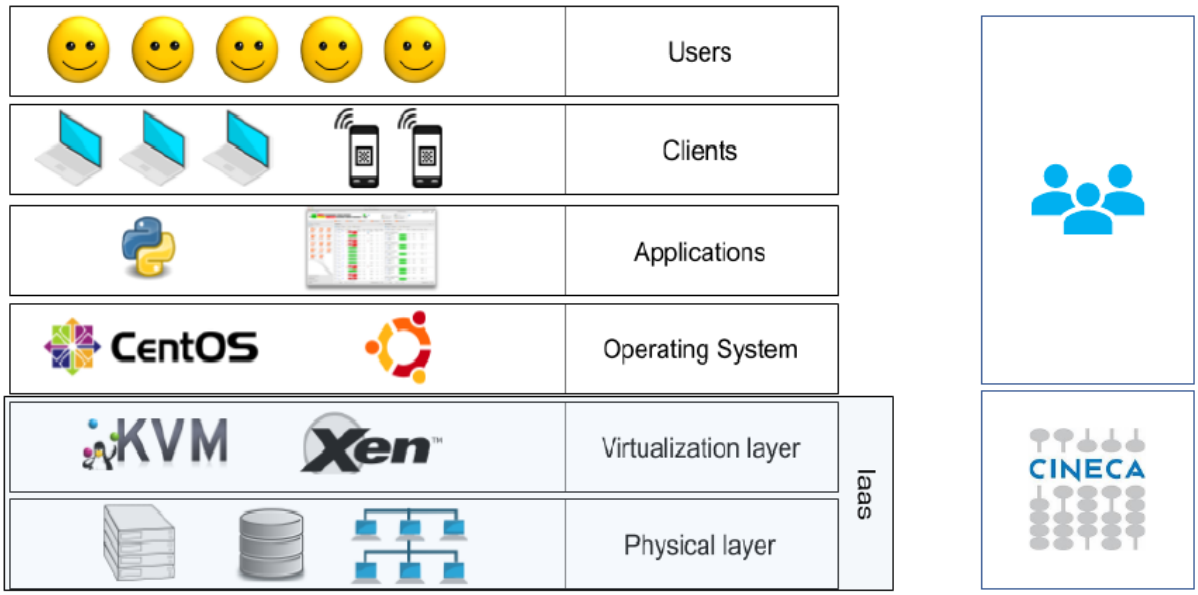

Figure 3.Layers of service deployment model.

3.2.4 Resource Allocations

In the conceived multi-cluster orchestration, the resources at CINECA will be mainly deployed for the computationally intensive workloads. To this end, an estimate of the resources required to support the EO4EU services was made after a budget analysis with the relevant stakeholders. Following this, initial resources allocation was made as listed below. The need for additional resources, if any, will be assessed after the first operational testing of the integrated platform.

- Computing: 200 vCPU and 1.5 TB of RAM
- Block Storage: 5120 GB
- 40 Public IPs

3.3 CINECA High Performance Computer (HPC) Infrastructure

The EO4EU system will leverage on the High-Performance Computing (HPC) infrastructure of CINECA, in particular the LEONARDO cluster. LEONARDO, the pre-exascale Tier-0 EuroHPC supercomputer, is ranked in the Top500 list of the most powerful supercomputers at position 4. LEONARDO has two partitions: Booster Module and Data-centric Module. The booster module partition is based on BullSequana XH2135 supercomputer nodes, each with four NVIDIA Tensor Core GPUs and a single Intel CPU. It also uses NVIDIA Mellanox HDR 200Gb/s InfiniBand connectivity, with smart in-network computing acceleration engines that enable extremely low latency and high data throughput to provide the highest AI and HPC application performance and scalability. The Data-centric partition is based on BullSequana X2140 three-node CPU Blade and is equipped with two Intel Sapphire Rapids CPUs, each with 56 cores.

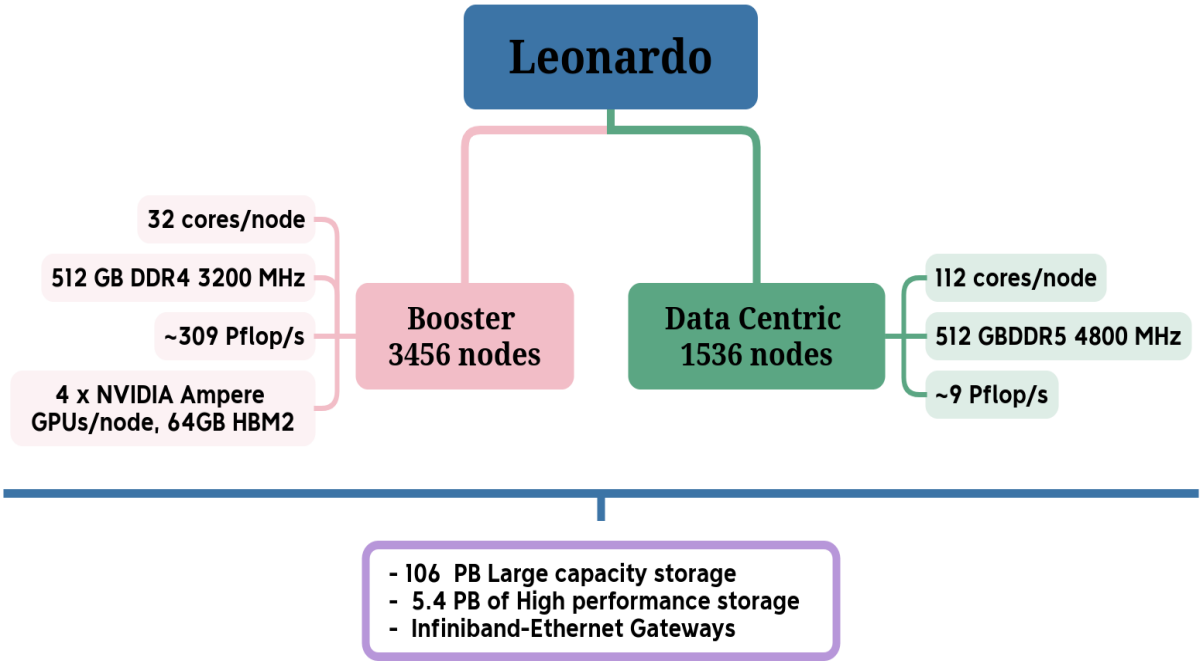

Figure 4. Leonardo HPC infrastructure

The HPC budget for the project is 100,000 std-h, of which 50,000 std-h have already been used for the ML tasks (Section 2.1.3.1), and a disc storage quota of 1TB for data processing.

3.3.1 Cloud and HPC Inter-connection

Processing, analysing and visualizing Earth Observation (EO) data requires a significant amount of compute power. The strict time requirements of a workflow system, like the one needed by the EO4EU project, relies mainly on Cloud resources, which allow for scalable, elastic and on-demand resources across different environments. However, in some cases it could be beneficial to leverage on batch computing systems. In particular, training of ML models on High Performance Computing (HPC) systems with cutting-edge GPU accelerators can significantly speed up the process. However, the interconnection between cloud and HPC systems is challenging as the technical solutions that establish the connections should be compliant with the security policies of the infrastructure provider. Due its specific computational power, HPC systems are often under strict security restrictions, both physical and software, protecting facilities from unwanted access and intrusions. In this section, we present some possible solutions to share computational workloads between the cloud and HPC systems.

In the specific case of CINECA facilities, the access happens through SSH certificates, renewed periodically by leveraging on Two Factor Authentication. The launch of a HPC job requires to build a SLURM script to be passed through SSH to the login nodes and then inserted in the Job Queuing system. All the data can be stored in specific filesystem and then copied, after the computation ends, from the HPC storage to other remote storage.

More specifically, CINECA can offer three different ways to access its own resources:

- Short lived SSH certificates which must be renewed (useful mainly for human users).
- Securing the virtual machine to be used to access the HPC resources: audit from external security society with penetration testing work. VM needs to have fixed IP address. Security firm releases a Vulnerability Assessment and Penetration Testing (VAPT) document certifying the security of the machine. Collect a log of the user's actions on the machine. Sign an assumption of responsibility document. Long lived certificate.

- Partner machine is proved to be secured by the company and a Security Manager of the Company takes responsibility on security. Long lived certificate.

Integration Steps:

- Create a chain user in CINECA UserDB.
- Get certificate to access the HPC facilities.
- Add fixed public IP to the machine accessing the HPC system.
- Launch test script on HPC systems.
- Security audit the system which accesses the HPC system.

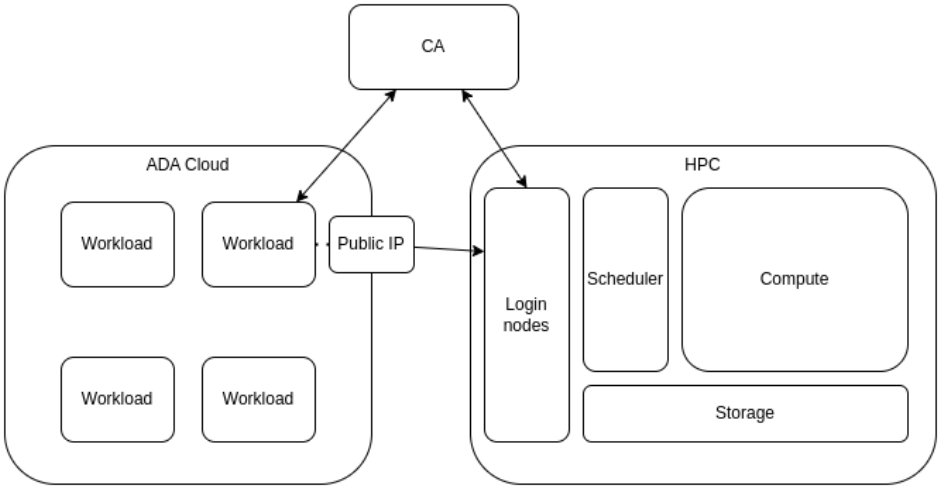

Figure 5. HPC Network System Architecture Diagram

In the infrastructure under discussion, there is currently an established method for inter-cloud access. Public access is provided over the internet using HTTPS, and for multi cloud communications, a secure tunnel has been implemented. As of now, this tunnelling is solely for the purpose of user management. The deployment of this tunnel reflects a conservative resource strategy; it is active only for user management to maintain security without overcomplicating the system. Although only one tunnel is currently in operation, the architecture allows for scalability. Should the need arise, additional service-specific tunnels can be deployed. The current tunnelling method employs sshuttle[9] due to its simplicity, efficiency and transparent operation, requiring no significant changes to the server-side infrastructure.

In terms of broader security measures and futureproofing, alternative solutions are being considered. These options are undergoing evaluation, but the priority remains on centralizing processing-heavy operations within a single cluster to maintain optimal efficiency and performance. The infrastructure utilizes a dual-cluster model, with one cluster dedicated to processing-intensive tasks and another reserved for external service monitoring. This design ensures efficient use of resources and minimizes the risk of a single point of failure, thereby enhancing overall system resilience.

The ongoing review and assessment of the infrastructure ensure that it continues to meet security, efficiency, and scalability needs, adapting as necessary to the evolving requirements of the system it supports.

3.3.2 Multi-cloud Architecture

EO4EU's cloud architecture spans two distinct cloud infrastructures: the CINECA ADA cloud and the WEkEO cloud, with different strengths and purposes, providing robust and scalable services to meet EO4EU's diverse operational demands.

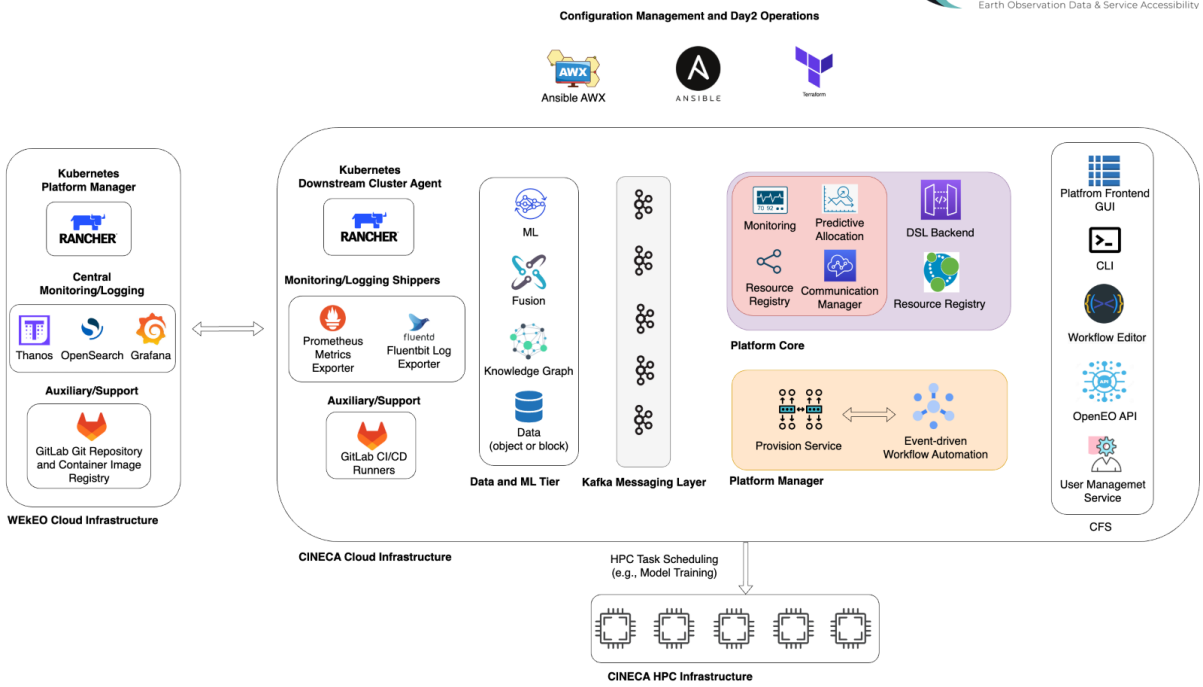

Figure 6. EO4EU Multi-Cloud Architecture

Figure 6 reports a diagram of the EO4EU Multi-Cloud Architecture. The main cloud infrastructure leveraged by EO4EU is the CINECA Ada cloud, where the project has allocated most of the computing resources. Most computationally heavy workloads, such as the backends, the fusion engine, the machine learning and other CPU intensive tasks are thus hosted on the CINECA Ada cloud. Moreover, the CINECA Ada Cloud is located closely to HPC resources. This adjacency significantly reduces latency and enhances data throughput rates, factors that are critical for HPC applications that demand rapid processing of large datasets and intensive computational tasks.

On the other hand, the WEkEO Cloud can host workloads closer to the data sources. Indeed, the WEkEO infrastructure provides to its users a service named Harmonized Data Access (HDA), i.e., a REST API which allows unified access to Copernicus services as well as other EO platforms. The data retrieved through HDA is meant to be streamed, compressed, relayed to the components hosted in the CINECA Ada Cloud, providing a de-facto integration of the two clouds. To avoid single points of failure, the EO4EU IAM infrastructure, the monitoring infrastructure and the IaC tools are currently hosted on the WEkEO cloud. Moreover, WEkEO resources can be leveraged in the case of maintenance of the CINECA Ada Cloud.

The Multi-cloud infrastructure features an established integrated method for inter-cloud access. Public access is provided over the Internet using HTTPS, while for multi-cloud communications, a secure tunnel has been implemented. As of now, this tunnelling is solely for the purpose of user management. The deployment of this tunnel reflects a conservative resource strategy; it is active only for user management to maintain security without overcomplicating the system. Although only one tunnel is currently in operation, the architecture allows for scalability. Should the need arise, additional service-specific tunnels can be deployed. The current tunnelling method employs sshuttle1 due to its simplicity, efficiency and transparent operation, requiring no significant changes to the server-side infrastructure.

The continuous review and assessment of the infrastructure ensures that it meets security, efficiency, and scalability needs, adapting as necessary to the evolving requirements of the system it supports.

3.4 Allocation of Components & Services on Infrastructure

3.4.1 Integration Planning towards the Design and Development of the required Infrastructure & Services

Integration planning and services specifications play a crucial role in designing, developing, prototyping, and testing a software dynamic platform system composed of various software components like API, dashboard, web XR/VR application, visual workflow editor, knowledge graph, and user management model, but it also defines the technical requirements for the corresponding infrastructure and services needed. This platform system aims to facilitate and process big data using AI/Machine Learning capabilities on a cluster or cloud architecture. Here are the key requirements and a recommended methodology approach for successful integration:
The following Table 13, indicates the basic Infrastructure and Services requirements of Integration Planning followed in the EO4EU Platform Software System:

Table 13. Integration Planning for the Infrastructure and Services' Requirements.

Requirements of Integration Planning	Description of Integration Planning
Interoperability	Ensures that all software components can communicate and exchange data seamlessly. Standard protocols and data type formats, such as JSON data format in the EO4EU Project, are followed to facilitate smooth data interactions and communications.
Scalability	The design of the EO4EU platform aims to handle increasing amounts of data and users. The system is capable to scale horizontally and vertically to accommodate growth.
Security	The EO4EU platform system implements robust security measures to protect sensitive data, user credentials, and the platform's functionalities. This includes encryption, authentication, and authorization mechanisms, such as the AAA framework for intelligently controlling the access on the EO4EU software components, applications, and data processed.
Data Processing Efficiency	Optimize data processing workflows and algorithms to handle big data efficiently. As multiple EO source data are considered to be big data, these multi-streaming data channels required distributed computing techniques and compression/de-compression ML algorithms to speed up processing, but to also selectively process the desired data.
Fault Tolerance	The EO4EU platform is resilient to failures and will be built based on adequate hardware and software resources to sustain the overloading of data, but to also be capable of multi-processing based on a large number of users working concurrently. Implementing redundancy and fault-tolerant strategies to ensure continuous operation is a must for the EO4EU platform system.
High Performance	Striving for high performance by optimizing code, using caching mechanisms, and employing efficient algorithms.
Documentation	Constructing a well-defined document of the integration architecture, API specifications, data formats, and data flows to aid development and future maintenance.

3.4.2 Tier Distribution to the available infrastructure

The IaaS Tier running the Openstack component encompassing all the required Kubernetes Clusters is deployed on the CINECA Cloud infrastructure. This infrastructure was chosen because of the availability of high computing power compared to the WEkEO infrastructure allowing the proper operation of several applications running within the Kubernetes Cluster. Since most of the components that will run within the Kubernetes Cluster do not require a very high demand in terms of GPU usage, there is no need to use the CINECA HPC for this Tier. Furthermore, since ML Tier is distributed in several sections on the platform, the IaaS Tier should ensure that there will be available resources for the operation of those, already trained AI/ML models without any performance trade-offs.

The complete Front-End Tier is deployed on the CINECA Cloud infrastructure due to the fact that there is a plethora of applications included in this Tier and some of them require a significant amount of computing power as well as data storage resources allowing proper storage of the data either on S3 Bucket or the Elastic Search instances for the visualisation of the processed or unprocessed EO data.

In terms of the PaaS Tier, a modular approach was implemented. The Tier was split into 2 main components, the Platform Controller and the Platform Orchestrator. Due to the complicated nature of the Platform Controller, it has been deployed on the WEkEO infrastructure, utilising its unique flexibility simplifying deployment of components and the overall control of the platform including the Key Cloak instance for the Authentication SSO component. On the other hand, the Platform Orchestrator component of the PaaS Tier require more demanding resources hence, it has been deployed on the CINECA Cloud Infrastructure.

Finally, the Data Tier, both the KG and the Data Fusion Engine have been deployed on the CINECA Cloud infrastructure since they do require a significant amount of computing and data storage resources, as well as allowing for GPU support since it is needed for these components. The ML/Inference server service of the Data Tier is the most demanding component of the platform. This component is responsible for the training of the ML models, a process that requires advanced GPU usage. This component is deployed on the CINECA HPC that is especially featured for such applications.

3.4.2.1 ML/Inference Server Service Benchmarks

In order to understand the actual benchmarking needs for this service/component, several tests have been performed to demonstrate the essence of using significant GPU enhancement.

3.4.2.1.1 Inference Benchmark

To evaluate the performance of the inference server, we used NVIDIA Model Analyzer tool which allows to evaluate the performance of the inference server with artificial load. We first evaluated the performance on a full Leonardo node using GPUs. The GPUs used are 4 x Nvidia Ampere GPU 64GB HBM2. We optimized a single compression ML model and obtained the following optimized configuration.

Table 14. Inference Benchmarks.

Model Config Name	Max Batch Size	Dynamic Batching	Instance Count	p99 Latency (ms)	Throughput (infer/sec)	Max GPU Memory Usage (MB)	Average GPU Utilization (%)
Bigearth-net-compression-compress-torch_config_6	64	Enabled	4:GPU	705.693	767.391	29138	97.9
Bigearth-net-compression-compress-torch_config_14	64	Enabled	8:GPU	2274.503	767.324	53123	98.5
Bigearth-net-compression-compress-torch_config_21	32	Enabled	12:GPU	917.124	767.049	49265	99.6
Bigearth-net-compression-compress-torch_config_default	2	Enabled	4:GPU	240.243	538.458	3764	90.9

The following graph shows the throughput as a function of latency for each configuration.

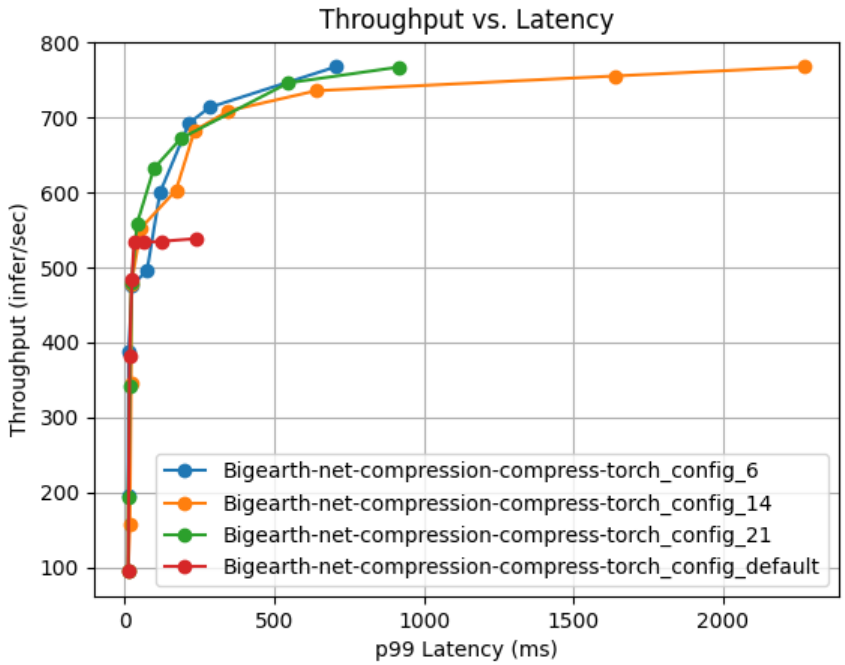

Figure 7. Throughput for each configuration.

As we can see, the maximum performance is between 700 and 800 inferences per second. However, higher throughput comes at the cost of higher latency. The same model inferred without GPUs has a throughput of less than 8 inferences/second, which clearly shows the speed-up that comes from having GPUs available. In practice, any model that has a real load should be served by GPUs, and CPUs should only be used for infrequently used models with insignificant loads.

3.4.2.1.2 Training Benchmark

We used the CINECA infrastructure to train our model. We used GPUs for training because we know that speed-up is even more important in training than in inference because of the additional computational cost. To train our models, we used the Singularity software, which allows a container to be run on the HPC. The hardware found on the CINECA clusters is one of the best hardware available for machine learning making it ideal for training.

The use of network storage to store data has been a major difficulty in training on HPC. One of the main EO data types is GeoTIFF, which is a format consisting of a single tif file for each instrument measurement. For example, for Sentinel-2 we used 10 bands, which consists of 10 tif files for a single instance. This results in a large number of small file access operations, which are relatively slow in a remote storage environment. Without solving this, single epoch pass time was 1h on local SSD and 12h on remote storage. The solution we implemented on CINECA cluster was to use the WebDataset format, which allows data to be read directly from multiple tar files. This solves the problem of many small files. The structure of the data is not significantly changed. With these changes, the CINECA cluster has been an invaluable resource, as evidenced by our current resource usage of 8678 hours with 343 jobs being launched. More computations will be performed in the coming months as the use cases slowly mature and require model training. We expect to use at least an order of magnitude more of resources till the end of the projects. The hours have been used to train and evaluate models. The results of the training can be found in the machine learning deliverable D3.3.

4 Summary of Infrastructure & Services Specifications

Table 15. Summary of minimum infrastructure & services specifications.

Service/Component	Tier	Requirements	Infrastructure	Total Infrastructure Requirements
ML/Inference Server	Data Tier	Computational Power: 8 CPUs Memory: 64 GB RAM Storage: 200 GB GPU Demand: Very high (12 GPUs, 64GB RAM)	CINECA HPC	Computational Power: 8 CPUs Memory: 64 GB RAM Storage: 200 GB GPU Demand: Very high (12 GPUs, 64GB RAM)
Platform Controller	PaaS Tier	Computational Power: 4 CPUs Memory: 32 GB RAM Storage: 60 GB GPU Demand: No need	WEkEO	Computational Power: 4 CPUs Memory: 32 GB RAM Storage: 60 GB GPU Demand: No need
Knowledge Graph	Data Tier	Computational Power: 4 CPUs Memory: 64 GB RAM Storage: 100 GB GPU Demand: Low	CINECA Cloud	Computational Power: 28 CPUs Memory: 268 GB RAM Storage: 740 GB GPU Demand: Medium need for some components
Data Fusion	Data Tier	Computational Power: 4 CPUs Memory: 16 GB RAM Storage: 100 GB GPU Demand: Medium		
OpenStack	IaaS Tier	Computational Power: 2 CPUs Memory: 8 GB RAM Storage: 40 GB GPU Demand: No need		
Platform Orchestrator	PaaS Tier	Computational Power: 4 CPUs Memory: 32 GB RAM Storage: 80 GB GPU Demand: No need		

AI/ML Marketplace	PaaS Tier	Computational Power: 1 CPUs Memory: 4 GB RAM Storage: 40 GB GPU Demand: No need		
Online Portal/Data Analytics Visualization	Front-End Tier	Computational Power: 2 CPUs Memory: 16 GB RAM Storage: 80 GB GPU Demand: No need		
DSL Engine/Workflow Editor		Computational Power: 4 CPUs Memory: 32 GB RAM Storage: 80 GB GPU Demand: No need		
Authentication SSO/UMM		Computational Power: 1 CPUs Memory: 32 GB RAM Storage: 40 GB GPU Demand: No need		
XR/VR Service		Computational Power: 4 CPUs Memory: 32 GB RAM Storage: 100 GB GPU Demand: Medium		
OpenEO API		Computational Power: 2 CPUs Memory: 32 GB RAM Storage: 80 GB GPU Demand: No need		

5 Conclusion

This document, denoted as "D4.1 – Infrastructure & Services Definition," marks a significant milestone as the inaugural deliverable emerging from the efforts of Work Package 4 within the EO4EU project. It intricately delves into multiple facets integral to the successful establishment and functioning of the EO4EU platform.

One pivotal aspect addressed in this document is the comprehensive listing of the various services and components constituting the EO4EU platform. Through a meticulous exploration, the document provides a clear and insightful understanding of the purposes each element serves within the broader framework of the project. Furthermore, a fundamental focus of this deliverable is to offer an initial snapshot of the infrastructure requirements essential for the deployment of the platform's components and services. This foresight ensures a foundational understanding of the technological prerequisites, setting the stage for subsequent phases of development.

An in-depth overview of the available infrastructure earmarked for hosting the EO4EU platform is presented, bringing attention to three primary infrastructures: WEkEO, CINECA Cloud, and CINECA High-Performance Computer (HPC). The strategic allocation of these infrastructures is a critical aspect of the document, influencing the hosting strategy for various components or groups (Tiers) of components and services. The document further elucidates a well-defined methodology for achieving a multi-cloud infrastructure. This strategic approach, encapsulating all available infrastructure resources, ensures not only flexibility but also efficiency in the deployment of EO4EU platform components and services.

In terms of hosting strategy, the document provides explicit details on the infrastructure chosen for each component or group, accompanied by a rationale behind these decisions. Notably, the Platform Controller finds its hosting on the WEkEO infrastructure, while CINECA Cloud emerges as the designated host for the majority of other components, leveraging a Kubernetes Multi-Cluster. For the critical Machine Learning/Inference Server service, instrumental in model training, CINECA HPC is selected to deliver optimal performance. To encapsulate the collective needs of all components, the document furnishes a summarized version of infrastructure requirements. This holistic perspective ensures a balanced and efficient allocation of resources, setting the stage for a seamless integration of services within the EO4EU platform.

Conclusively, the document not only delineates the intricacies of infrastructure and service definitions but also lays a robust foundation for the forthcoming phases of the EO4EU project. The detailed insights into computational, memory, storage, and GPU capabilities, presented in the concluding section of the document, serve as a valuable reference for evaluating and optimizing the performance of the platform.