Project Title	Hybrid Eco Responsible Optimized European Solution
Project Acronym	HEROES
Grant Agreement No.	956874
Start Date of Project	01.03.2021
Duration of Project	24 Months
Project Website	heroes-project.eu

D3.1 – Architecture Design

Work Package	**WP 3.1, Architecture Design**
Lead Author (Org)	**Davide Pastorino (Do IT Systems) & Jorik Remy (UCit)**
Contributing Author(s) (Org)	**Anaëlle Dambreville (UCit), Benjamin Depardon (UCit), Benoit Marchand (UCit), Jose Torres (HPCNow!), Julita Corbalan (BSC), Sablin Amon (UCit)**
Reviewed by	**Ana Guttierez (HPCNow!), Jose Torres (HPCNow!), Vincent Bosquier (UCit), Philippe Bricard (UCit)**
Approved by	**Management Board**
Due Date	**31.08.2021**
Date	**03.09.2021**
Version	**V4.0**

Dissemination Level

X	PU: Public
	PP: Restricted to other program 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)

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Versioning and contribution history

Version	Date	Author	Notes
0.1	12/07/2021	Davide Pastorino (Do IT Systems)	TOC, document structure draft
0.2	13/07/2021	Davide Pastorino (Do IT Systems)	First draft of Data Transfer Module (par. 2.4)
0.3	15/07/2021	Jorik Remy (UCit)	First draft of Identity Management module (par. 2.1) and first draft of Workflow submission and job management (par. 2.6)
0.4	22/07/2021	Davide Pastorino (Do IT Systems)	Updated par. 2.4, first full draft of Deployment Module (par 2.2)
0.5	23/07/2021	Jorik Remy (UCit)	First draft of User Interface & UI module (par. 2.10) + Updated par. 2.1 + Updated par. 2.6 + Updated tables, figures, terminology and structure
0.6	23/07/2021	Sablin Amon (UCit)	First draft of Container Management & Orchestration (par. 2.5)
0.7	26/07/2021	Anaelle Dambreville (UCit)	First draft of Decision module & Performance metadata (par. 3.7)
0.8	27/07/2021	Davide Pastorino (Do IT Systems)	First draft of the Executive Summary, Introduction (par. 1) and Global

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

			Architecture Overview (par. 2); renamed Data Transfer Module to Data Management Module.
0.9	28/07/2021	Davide Pastorino (Do IT Systems)	First draft of the Logging & Accounting Module (par. 3.8)
1.0	29/07/2021	Benoit Marchand (UCit)	Updated part 3.10
1.1	02/08/2021	Benjamin Depardon (UCit)	Added a new section with definitions. Added comments/revisions.
1.3	03/08/2021	Jorik Remy (UCit)	Update of the Identity Management Module (module 1) and update of the Workflow & Job Management (module 6)
1.4	05/08/2021	Jorik Remy (UCit)	Update of the Application and Container Management & Orchestration (module 5) Added comments/revisions.
1.5	06/08/2021	Jorik Remy (UCit)	First draft of Actions in Definitions part. Add bibliography + move externals sources of module to the bibliography
1.6	06/08/2021	Jose E Torres (HPCNow)	Review and first changes on the Workflows module section.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

1.7	07/08/2021	Jose E Torres (HPCNow)	Added content to the Workflows module section: minors
1.8	09/08/2021	Benoit Marchand (UCit)	Updated User interface and API module. Updated bibliography.
1.9	09/08/2021	Jose E Torres (HPCNow)	Added content to the Workflows module section. Introduction + assumptions. Answered some comments
2.0	10/08/2021	Jorik Remy (UCit)	Updated schemas, added comments to the document + Update figure table + Update style + Updated Identity Management + Updated Application and Container Management & Organization
2.1	10/08/2021	Jose E Torres (HPCNow)	Minor changes on Workflows schemas and Module 6 content. Closed some comments.
2.2	11/08/2021	Anaelle Dambreville (UCit)	Update of Decision module & Performance metadata (par. 3.7)
2.3	11/08/2021	Jorik Remy (UCit)	Update Introduction + Updated Global Architecture Overview + Update

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

			Actions + add comments to the document & resolve other ones + Update bibliography + Update list of figures & table of contents
2.4	11/08/2021	Jose E Torres (HPCNow)	Updated Workflows section: formatting, added schema to Internal Interfaces section + module interaction, added content to the Expected outcomes section and Uncertainties section.
2.5	11/08/2021	Benoit Marchand (UCit)	Updated a few paragraphs within Section Executive summary, 1 ; 2 ; 3 ; and the introduction of section 4. Also updated content for section 4.1 and 4.10
2.6	12/08/2021	Benoit Marchand (UCit)	Comments and update on section 4.2. Update of section 4.10.
2.7	12/08/2021	Jose E Torres (HPCNow)	Updated: Uncertainties section
2.8	12/08/2021	Jorik Remy (UCit)	Updated: Terminology, Bibliography, all internal interfaces & interactions

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

			schemas, reference to titles and figures
2.9	12/08/2021	Jorik Remy (UCit)	Added: First draft of the conclusion
3.0	12/08/2021	Jorik Remy (UCit)	Updated: "Bibliography" becomes "References and Applicable Documents" + "Terminology" becomes "References and Applicable Documents" + Check of all figures legends and references + Introduction to the section 3 (Definition) at the end of the section 2
3.1	13/08/2021	Jorik Remy (UCit)	Updated: Section 5 (Conclusion) + Style + References
3.2	13/08/2021	Anaelle Dambreville (UCit)	Updated conclusion
3.3	16/08/2021	Jorik Remy (UCit)	Updated definitions + global architecture overview texts and schemas
3.4	16/08/2021	Benoit Marchand (UCit)	Updated conclusion
3.5	17/08/2021	Jorik Remy (UCit)	Updated minor details, add abbreviation, self-comment, and resolution
3.6	23/08/2021	Jose Torres (HPCNow)	Updated minor details and

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

			comments corrections
3.7	23/08/2021	Jorik Remy (UCit)	Updated minor details, figure references, and resolution
3.8	24/08/2021	Davide Pastorino (Do IT Systems)	Added Headnode definition in par. 4.2.1
3.9	25/08/2021	Benjamin Depardon (UCit)	Fixed refs + formatting
4.0	03/09/2021	Corentin Lefèvre (Neovia)	Final version approved by the Management Board

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Table of Contents

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

List of Figures

List of Tables

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

References and Applicable Documents

[1] AWS (Amazon Web Services) official website, August 2021, https://aws.amazon.com/
[2] OCI (Oracle Cloud Infrastructure), August 2021, https://www.oracle.com/cloud/
[3] Keycloak official website, August 2021, https://www.keycloak.org/
[4] OpenID Connect official website, August 2021, https://openid.net/connect/
[5] OAuth 2.0 official website, August 2021, https://oauth.net/2/
[6] SAML V2.0 Standard, OASIS standards consortium, August 2021, https://wiki.oasis-open.org/security/FrontPage
[7] Active Directory Domain Services, Microsoft, August 2021, https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/active-directory-domain-services
[8] LDAP, August 2021, https://ldap.com/
[9] SILL (Socle Interministériel de Logiciels Libres), 2021, https://www.mim-libre.fr/wp-content/uploads/2021/06/sill-2021.pdf
[10] CloudInit, project website, August 2021, https://cloud-init.io/ https://docs.fluentd.org/quickstart/td-agent-v2-vs-v3-vs-v4
[11] TD-Agent (TreasureData Agent), FluentD Documentation, https://docs.fluentd.org/quickstart/td-agent-v2-vs-v3-vs-v4
[12] Terraform, project website, August 2021, https://www.terraform.io/
[13] Ansible, project website, August 2021, https://www.ansible.com/
[14] EAR, August 2021, https://www.eas4dc.com/
[15] RClone, project website, August 2021, https://rclone.org/
[16] MinIO, project website, August 2021, https://min.io/
[17] Ryax, official website, August 2021, https://ryax.tech/
[18] FluentD, official website, August 2021, https://www.fluentd.org/
[19] Elasticsearch, OpenDistro for Elasticsearch, August 2021, https://opendistro.github.io/for-elasticsearch/
[20] Kibana, Elastic website, August 2021, https://www.elastic.co/kibana/
[21] SSPL / ELv2 license, Elastic website, https://www.elastic.co/blog/elastic-license-v2
[22] FastAPI, FastAPI documentation, August 2021, https://fastapi.tiangolo.com/
[23] RabbitMQ, official website, August 2021, https://www.rabbitmq.com/
[24] Node.js, NodeJs documentation, August 2021, https://nodejs.org/en/about/
[25] ReactJS, ReactJS documentation August 2021, https://reactjs.org/
[26] Angular, project website, August 2021, https://angular.io/
[27] Django, project website, August 2021, https://www.djangoproject.com/
[28] Vue.js, project website, August 2021, https://vuejs.org/
[29] NGINX, official website, August 2021, https://nginx.org/
[30] Elastic Common Schema (ECS) Reference, Elastic website, August 2021, https://www.elastic.co/guide/en/ecs/current/index.html
[31] Introduction to Elastic Common Schema (ECS), Elastic website, August 2021, https://www.elastic.co/guide/en/ecs-logging/overview/master/intro.html

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

[32] Using AWS Price List API, Amazon Web Services Documentation, August 2021, https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/price-changes.html
[33] Azure Retail Prices overview, Microsoft documentation, August 2021, https://docs.microsoft.com/en-us/rest/api/cost-management/retail-prices/azure-retail-prices
[34] rsyslog, August 2021, https://www.rsyslog.com/
[35] OKA, OKA official website, August 2021, http://oka.how/
[36] Predict-IT, UCit official website, August 2021, https://ucit.fr/index.php/predict-it/
[37] AWS CLI, AWS Documentation, https://aws.amazon.com/cli/
[38] OCI CLI, Oracle Cloud Infrastructure Documentation, https://docs.oracle.com/en-us/iaas/Content/API/Concepts/cliconcepts.htm
[39] Kurtzer, G. M., Sochat, V., & Bauer, M. W. (2017). Singularity: Scientific containers for mobility of compute. PloS one, 12(5), e0177459.
[40] Oracle Cloud Infrastructure Documentation - Accessing List Pricing for OCI Products, August 2021, https://docs.oracle.com/en-us/iaas/Content/GSG/Tasks/signingup_topic-Estimating_Costs.htm
[41] Perf Wiki – perf: Linux profiling with performance counters, August 2021, https://perf.wiki.kernel.org/index.php/Main_Page

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

List of Acronyms and Abbreviations

Terminology/Acronym	Description
WP	Work Package
AI	Artificial Intelligence
API	Application Programming Interface
ID	Identifier
CSP	Cloud Service Provider
DoA	Description of Action
EC	European Commission
GA	Grant Agreement to the project
HPC	High-Performance Computing
IaaS	Infrastructure as a Service
IAM	Identity Management
KPI	Key Performance Indicator
ML	Machine Learning
OS	Operating System
DB	Database
UI	User Interface
SAML	Security Assertion Markup Language
LDAP	Lightweight Directory Access Protocol
AD	Active Directory
CRUD	Create Read Update Delete
AWS	Amazon Web Services
OCI	Oracle Cloud Infrastructure
EAS	Energy Aware Solution
EAR	Energy Aware Runtime
MSR	Model-Specific Register
ECS	Elastic Common Schema
O. S	Operating System
CLI	Command Line Interface
SSH	Secure Shell

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Executive Summary

The HEROES project is aiming at developing an innovative European software solution allowing industrial and scientific user communities to easily submit complex Simulation and ML (Machine Learning) workflows to HPC (High Performance Computing) Data Centres and Cloud Infrastructures. It will allow them to take informed decisions and select the best platform to achieve their goals on time, within budget and with the best energy efficiency.

This document describes the high-level architecture of the HEROES platform and its main components. The architecture is designed to deliver AI (Artificial Intelligence) and HPC workflows as a Service to end-users and third-party services. Access to the platform is done through web services APIs, allowing any authorized user and software tool to run a workflow on selected target platforms. Both workflow execution and data management are handled by the solution in a transparent manner. The platform also embeds a decision module that will help users in the selection of the computing platform of choice (HPC centre or Public Cloud) and to submit their jobs more easily – ideally in a transparent way – while taking into account their constraints (performance, energy and costs...). The solution will allow HPC Centres to provide access to their resources with an associated cost and allow other application and resource providers to integrate and publish their applications and workflows as well.

For the purpose of this document, the HEROES platform has been divided into several modules, each of them performing a specific set of tasks within the platform.

The description of each module in this document includes:

- A list of assumptions made during the design process, aimed at explaining design choices
- An overview of the module and its purpose
- A schematic view of the module's dependencies and interactions with other modules of the platform as well as a brief description of the main interactions
- A list of external dependencies such as third-party software that the module will eventually use
- A description of the module expected behaviour, exemplified through high-level flow diagrams
- A list of uncertainties that will need to be addressed during the implementation

While this document contains all essential information that will be used to drive the implementation tasks during the development of the HEROES platform in the next years, a certain degree of flexibility with the architectural design is to be expected due to the many unknowns still present at this time.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

1 Introduction

The purpose of the HEROES project is to provide an end-to-end distributed computing platform leveraging existing HPC Centres and Cloud Service Providers (CSP). The ML and simulation workflows require a great amount of computing power that can only be delivered by High Performance Computing (HPC) facilities. These environments are complex by nature: distributed, heterogeneous (hardware and software), and require many skills to master and use them. We propose to abstract the complexity of accessing such platforms and deliver the access to simulations through an HPC as a Service platform. Complex placement decisions must be taken while executing these workflows, and so the HEROES project will include a decision module that will leverage metadata gathered on all computing platforms and workflows to take informed decisions and select the best placement for the workflow's jobs. The HEROES project is also aiming at simplifying publication of computing resources (clusters and cloud resources) on one side and ease their consumption on the other side. The project will support HPC/ML workflows and will make use of AI/ML technology to take informed and guided decisions on placement and configuration of those workflows. The resulting software solution (referred to as "HEROES" hereafter) will be based on existing solutions, or solutions under development by the partners of the consortium.

HEROES, as described in this document, will be an HPC software solution with the following objectives:

- Allowing HPC Centres to present and easily publish on a marketplace environment the resources they are willing to share and the terms and conditions they provide for them
- Integrating transparently major Cloud Service Providers
- Supporting composable HPC/ML workflows based on containerized components.
- Providing an interface allowing the end-user to transparently run transparently workflow components, independently from the heterogeneous complexity of the underlying infrastructure
- Delivering recommendations for the best possible resource to use regarding user specific HPC/ML workloads considering time-to-result, resource costs and energy consumption

HEROES will allow users to securely access different kinds of resources, via an intuitive and easy-to-use web portal:

- Containerized applications
- Resource managers
- HPC resources published by Cloud Service Providers (CSP)
- HPC resources published by HPC Centres

HEROES will keep track of actions performed by users, and will consolidate their consumptions in a billing dashboard, allowing to sell computing hours and access to applications in an easy way.
HEROES will provide both a high-level interface, and the underlying connectors to infrastructures needed to build a distributed AI and HPC platform.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

2 Global Architecture Overview

The first high-level design of the HEROES architecture was based on logical blocks fitting the requirements and the needs of the platform. The choice of the modularity as an architectural basis has been made to manage the technical complexity of such a solution leading us to the global architecture as shown in Figure 1.

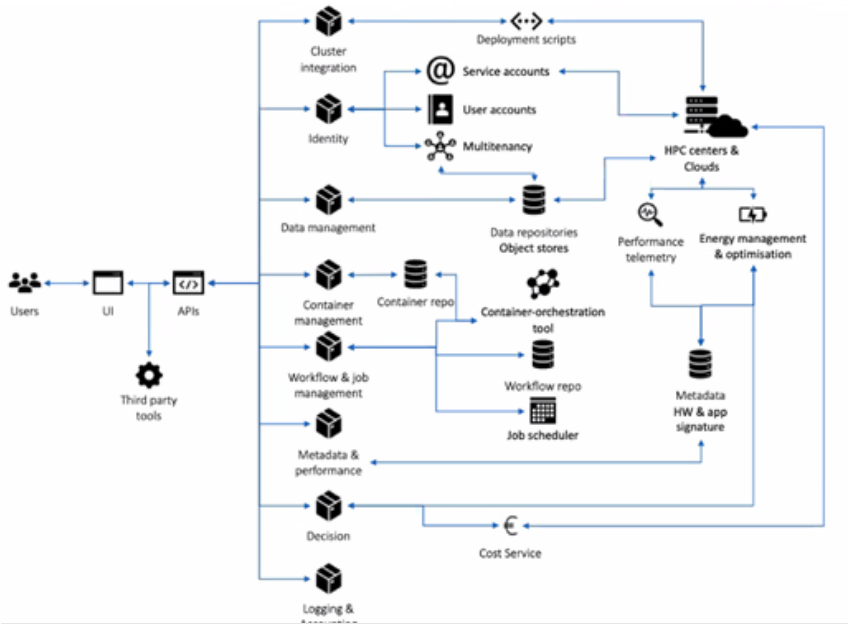

Figure 1 - First version of the overall design

Subsequently, we proceeded to group distinct systems of our architecture into ten "Modules", presented in Figure 2.

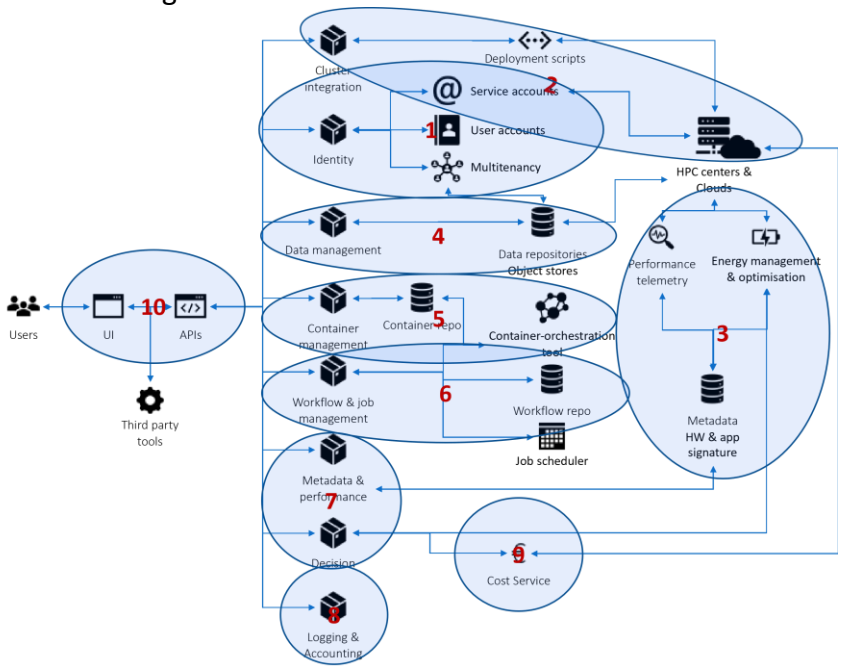

Figure 2 - Modular reorganization of the architecture

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Each module has been assigned an ID (Identifier) number and a name:

- Module 1 – Identity Management
- Module 2 – Deployment & Integration
- Module 3 – Energy Management & Optimization
- Module 4 – Data Management
- Module 5 – Application and Container Management & Orchestration
- Module 6 – Workflow & Job Management
- Module 7 – Decision Module & Performance Metadata
- Module 8 – Logging & Accounting
- Module 9 – Cost Management
- Module 10 – User Interface & API

The overall architecture design has been completed with additional details for each module and with basic infrastructural details, data transport & communication channels to be expected between each module, as presented in Figure 3.

The remainder of the document is organized as follows: Section 3 provides the vocabulary and definitions used throughout the document: it describes the users, resources, and actions definitions to meet the objectives set in the introduction (Section 1). Then, the details of each module including its assumptions, internal interfaces & interactions, external dependencies, expected outcomes & behaviour as well as uncertainties are presented in Section 0.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

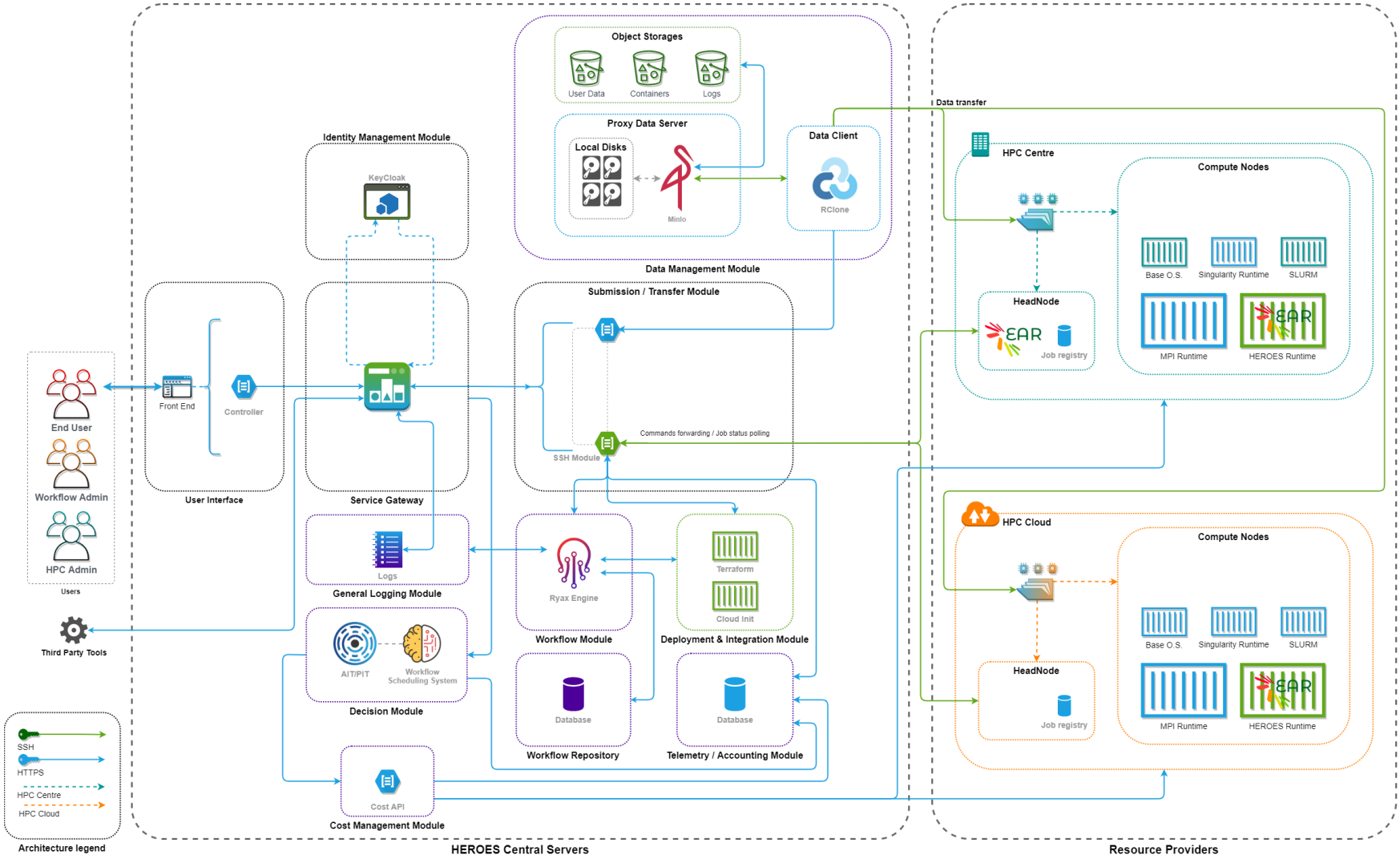

Figure 3 - Detailed Architecture Design

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

3 Definitions

In this section, we define a few terms that will be used throughout the following sections. In particular, we define who will be the actors (type of **Users**, **Roles** and **Accounts**) that will interact with the HEROES platform, and the resources that HEROES will rely on.

3.1 User, Roles and Account

The HEROES platform links two different entities: on one side we have the resource providers (data centres, CSPs), on the other side the end-users who will use the resources. We define the following elements.

3.1.1 User

A **User** is a physical (person) or logical (program) entity that interacts with the HEROES platform through its various interfaces (web, APIs). To do so, the User needs a valid **User Account** that grants him access and rights within the HEROES platform.

3.1.2 User Account

A **User Account** is a logical object that links a User to the HEROES platform: it is stored in a Directory that holds authentication and authorization information.

A **User Account** is part of an **Organization** (tenant) that is the legal entity (e.g., enterprises, universities, physical person – referred to as "single-user") responsible for administrative actions (e.g., billing, contract signing...) for one or more User Accounts.

Within the Organization, a User Account belongs to 1 or multiple **Projects** (groups). A Project is a group of users who share a common usage of the HEROES platform (e.g., technical: workflows, data, budget constraints...).

Each User Account is associated with 1 or multiple **Roles** in the Organization. A Role defines a set of **Policies** that grants access to a set of **Features**. Example of Roles:

- Engineer / Researcher
- Workflow admin
- Application packager (containers)
- User / Project administrator
- Billing admin (including group / project, budget)

A specific Organization is also defined: the **HEROES Organization**. It represents the initial Organization responsible for managing the HEROES platform. On top of the previous Role examples, some additional Roles can be defined for this Organization:

- Solution administrator (hardware / software platform HEROES management)
- Tenant / Organization administrator

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- Infrastructure admin (CSP & HPC Centre access management)
- Billing admin

3.1.3 Service Account

A **Service Account** is an account ("technical object") used by modules of the HEROES platform to access and interact with the resources offered by a Resource Provider (HPC Centre, CSP). For example:

- For a CSP: API keys (private key, secret key)
- For an HPC Centre: account with login + password or ssh (secure shell) keys or certificate…

A Service Account needs to comply with a set of minimum requirements and have enough rights on the resource provider to allow the HEROES Platform to submit jobs, create Cloud resources, have access to a set of resources with restrictions

- CSP: subset of services, limits on number of instances…
- HPC Centre: queue(s), limits on number of allocated cores/jobs, fair share…

A Service Account can be shared between multiple Organizations, or specific to a single Organization.

3.1.4 Provider Account

A provider account is an account that allows a **Provider** to interact with the HEROES platform and propose access to its computing resources (**Resource Provider** – e.g., an HPC Centre) or to workflows (**Application Provider**) to **Organizations** under certain conditions (price, availability, SLA…). The resources and the applications can be published by their Providers and become accessible to any Organization accepting the conditions through the marketplace environment.

For example, for an HPC Centre to enrol in the HEROES solution and provide access to its resources:

- It creates a Provider Account with details on resources it provides, at what price…
- This account is then used to integrate the cluster in the solution and make it available to selected Organizations

3.1.5 API Access

Every type of account communicates with HEROES through APIs (on top of which interfaces can be developed – e.g., web portal). It is needed for many use cases, for example for our services in the cloud / HPC centres to report back to the HEROES solution on certain events (e.g., end of job…) or to report metrics.

As most HPC Centres will not allow to have outgoing network connections, we will rely mainly on a "pull model" to gather information about running jobs or accessing data: the HEROES

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

platform will contact, through SSH, the HPC Centre to execute commands, gather information, for all its interaction with the Centre. The API access to HEROES platform will mainly be for third party tools, or between HEROES modules.

3.2 Resource providers

We consider two types of resource providers: HPC Centres and CSP

3.2.1 HPC Centre

HPC Centres are owned and managed either by a public organization (e.g., university) or a private company. Each cluster present in the HPC Centre can be part of HEROES either entirely or a subset of the available resources, the owner decides which part of the HPC Centre is made accessible to HEROES:

- Which cluster
- Which computing queue in each cluster
- At what price for each queue

HEROES will require access to the HPC Centre (a S**ervice Account**) to interact with it: a "standard unprivileged account" (without any "root" or "administrator" privileges) on the HPC Centre will be the main requirement to technically connect the HPC cluster to the HEROES platform.

3.2.2 Cloud Service Provider

The goal of HEROES is to be CSP agnostic. The solution will allow to deploy resources and submit jobs to any CSP. The Cloud offers greater flexibility than HPC Centres in terms of types and access to resources. Though HEROES could deploy specific and independent lightweight resources in the CSPs, we plan on deploying full-fledged clusters for consistency with HPC Centres.

The initial model is that HEROES will own the account required to access the CSP and deploy resources, but future evolutions will allow the incorporation of private CSP accounts within Organizations.

We will initially concentrate on two CSP during the project: AWS (Amazon Web Services) [1] and OCI (Oracle Cloud Infrastructure) [2].

3.3 Actions

An **Action** is an interaction within the HEROES platform through its various interfaces (web, APIs). To do so, each role must have a list of possible actions. These actions can relate to simple user tasks or tasks dedicated to administrator roles.

The actions available for the Organization within the platform HEROES must be differentiated from the management actions of the HEROES platform.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The actions which will be available for the **Organization** include but are not limited to:

- The management of users, groups, and their attributes within the Organization
- The management of roles, including creation / deletion / modification / attribution of roles within the Organization
- The billing access and billing management within the Organization
- The access to the user personal data, including:
    - Access to its own home
    - Access to its organization data
- The interaction with workflow and jobs within the Organization, such as:
    - The authorization to submit jobs
    - The authorization to package and manage application
    - The authorization to define workflow

Actions that belong to external **Providers**:

- **Resource** Provider:
    - Present and publish the resources they are willing to share and the terms and conditions they provide for them with a pricing
- **Application** Provider:
    - Present and publish the applications they are willing to share and the terms and conditions they provide for them with a pricing in the Marketplace

Actions that belong to the **overall management of the HEROES platform** include but are not limited to:

- Management of any Organization
- Management of the Service Accounts
- Management of the Marketplace

3.3.1 Table of actions by roles

Table 1 presents example of Actions that Users can do depending on their Role in the HEROES platform.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Table 1. Example of possible actions by roles

Role / Actions	Submit job/workflow	Data access (home)	Manage data access	Manage workflows	Manage users/groups and roles	Package application	Publish applications marketplace	Publish resources in marketplace	Billings access and cost management
Client Organizations									
Engineer, Researcher	X	X							
Workflow admin	X	X		X					
Application packager	X					X			
Project admin			X		X				
Billing admin									X
Heroes Organization									
Solution admin	X	X	X	X	X	X			X
Tenant admin, Organization admin					X				X
Infrastructure admin									
Billing admin									X
Providers									
Resource Providers								X	
Application Providers	X					X	X		

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4 Details of each module

In this section, we dive into each module to illustrate their purpose and how we intend for them to work. We will cover in more details the assumptions we hold to be true regarding each of them as well as the dependencies we have already identified and the uncertainties that remains at this stage. We will also illustrate in this part how we intend for each module to interact with the others.

4.1 Module 1 – Identity Management

The Identity Management Module purpose is to perform secure authentication for users (e.g., end-users such as Engineer, Researcher, or providers such as Application providers) so that they can access services provided by the HEROES platform. This module will also provide and control authorization for the authenticated users depending on their role and privileges while they use HEROES' services.

The Identity Management is based on the following two mechanisms:

- The authentication allows user and provider accounts to identify themselves onto the platform
    - Ascertain the legitimacy of a user's or provider's access request
    - Validate the correct credentials given by user or provider accounts, giving back a token to propagate the identity
    - Store the users in internal solution or optional external identity providers
- The authorization, allowing requests to be executed depending on the available actions' permissions granted to the user / provider accounts
    - It validates correct privileges from authenticated user or provider accounts to authorize requests and access to the HEROES platform services

4.1.1 Assumptions

At the time of writing this document, the following assumptions are made:

- The accounts within the HEROES platform must be decoupled between user accounts and service accounts
    - Service accounts (internal accounts) are a dedicated account for a specific service of the HEROES platform
    - User accounts (organization accounts) are a part of an organization and are linked with billing
- The organizations will be independent from each other

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.1.2 Overview

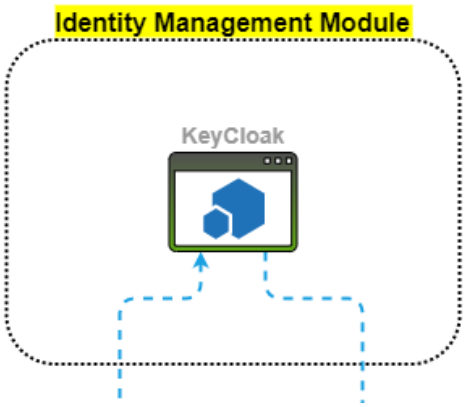

Figure 4 - Identity Management module

The needs of the Identity Management have been identified as:

- The Identities within the HEROES platform must be managed by separated Organizations
  - The Identity Providers must be internal to the platform by default, each Organization could have optional external identity providers
  - Each Organization might be able to use its own identity management systems with optional secured user federation including optional secured user federation using standard protocols
- The authorization must depend on roles, groups or project as defined in Section 3.1
- The solution must be able to scale according to the organizations and users within the HEROES platform

To meet the identified needs, the best choice seems to be a solution named Keycloak [3]. This Identity and Access Management solution (IAM) recommended by the French government through its "Interministerial Digital Department" [9] is a high-performance opensource solution providing API client authentication and multi-tenancy. It allows easy management over authentications, permissions and access to the HEROES services and platform.

The solution matches with our needs and assumptions for the following reasons:

- Keycloak provides authentication and authorization for API client
- The solution is based on standard protocols and provide support for OpenID Connect [4], OAuth 2.0 [5] and SAML [6]
- The solution has "Organization" system where the Organizations are called "realms"
  - Each "realm" (Organization) can have its own Users / Groups / Roles
  - Each "realm" can synchronize with external identity providers to create a User federation with SAML [6] (Keycloak can validate credentials from external stores and pull in identity information).
- Support AD (Active Directory) [7] and LDAP [8] to connect to existing user directories
- Provides high-performance: able to scale and stay highly available for the end-users and services

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.1.3 Internal interfaces & interactions

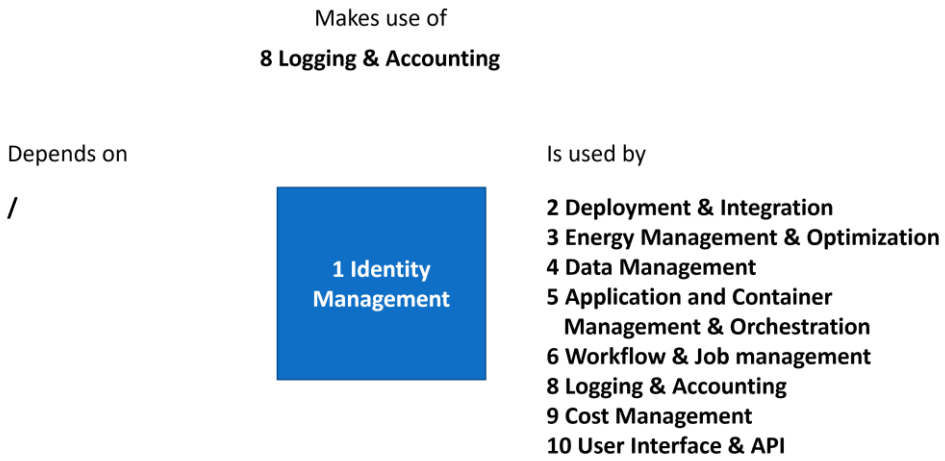

Figure 5 - Identity Management module internal interfaces & interactions

The IAM module will expose its functions via API to the other modules of the HEROES platform.

Internal and external authentication and authorization will be managed by the Identity Management Module:

- Internal services can interact with the IAM module and request an authorization check at any time.
- External access control will be centralized by the Service Gateway (see User Interface & API module Section 4.10) that will handle the authentication and authorization requests.

4.1.4 External dependencies

Third-party Software:

- Keycloak: Secured authentication and authorization solution.

4.1.5 Expected outcomes & behaviour

The Identity Management module will be responsible for handling authentication and authorization control whether it concerns internal request made by HEROES' services or external request from a user using the published APIs.

The following workflows schemas will present the expected behaviour of the module when confronted to a user's request to log in onto the HEROES platform and then proceed to call upon one of the published API.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.1.5.1 Authentication Workflow

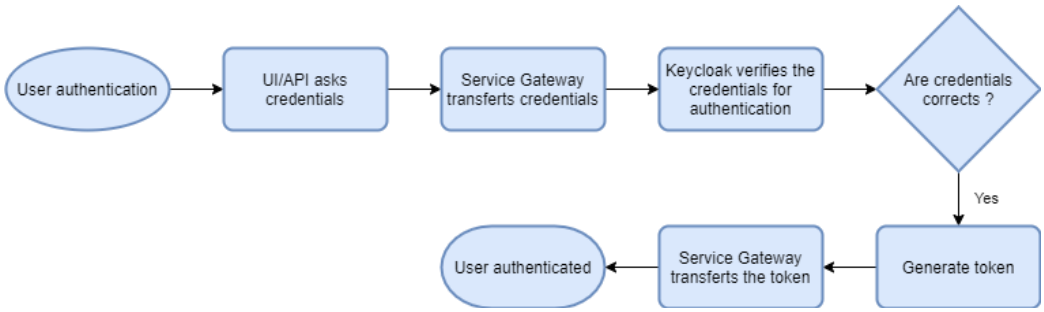

Figure 6 - Authentication workflow

The Authentication Workflow (as defined in Figure 6) starts by a "user's" authentication request and ends by the authentication of the user:

- "Users" (user or provider accounts) sends authentication request with valid credentials to the Service Gateway
- The Service Gateway transfers the request to Keycloak
- Keycloak returns a token to propagate the "User" identity if credentials are valid.

4.1.5.2 Authorization Workflow

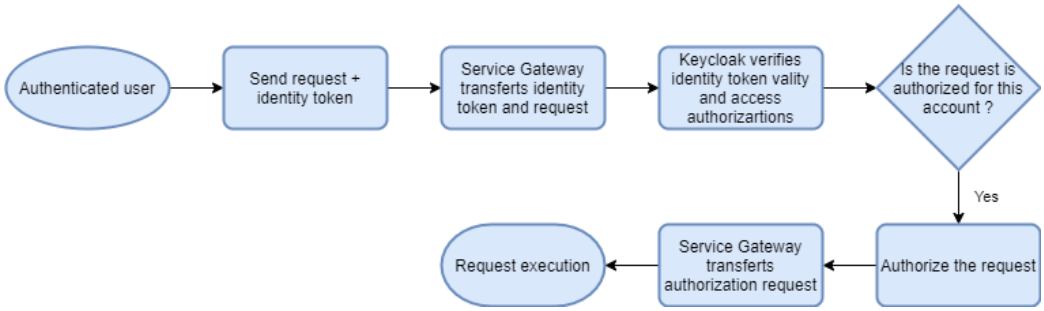

Figure 7 - Authorization workflow

The Authorization Workflow (as defined Figure 7) starts by a "user's" request from an already authenticated user and ends by the execution of the request:

- "User's" (user or provider accounts) sends request with valid identity token
- The Service Gateway transfers the request to Keycloak
- Keycloak verifies if the token is valid and if the owner of the token is authorized to perform the request
- Execute the request if the owner of the token is authorized

4.1.6 Uncertainties

The current uncertainties of the IAM module include but are not limited to:

- The ability for each module to work directly with Keycloak as an authentication and authorization service

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.2 Module 2 – Deployment & Integration

The purpose of the Deployment & Integration Module is to ensure that the remote components of the HEROES infrastructure are created, configured and ready to take in charge the Users' workloads.

The Deployment & Integration module will make use of automated and semi-automated components.

The fully automated components will mainly be used in workflows to activate / deactivate compute nodes and other short-term resources.

The semi-automated components will be used in workflows to provision static and long-term infrastructure of the HEROES platform, such as:

- Per-Organization Clusters in the Cloud
- HEROES Remote Services in HPC Centres
- HEROES Runtime in both Cloud and HPC Centres

4.2.1 Assumptions

At the time of writing this document, the following assumptions are made:

- The integration of remote resources in the HEROES platform will make use of a remote Headnode, defined as "a server (or VM, or container) identified to host the HEROES Runtime required to integrate a HPC cluster into the HEROES platform"
- HEROES will make use of remote HPC Centres and commercial CSP platforms for compute workloads (e.g., Amazon Web Services (AWS) [1], Oracle Cloud Infrastructure (OCI) [2])
- HPC Centres will limit access to their HPC clusters through access to a login node (used by HEROES as a Headnode) from specific IP addresses
- HEROES central servers (see Figure 3) will interact with commercial cloud platforms through VPNs
- The type of supported cloud instances will be limited to a small subset of the cloud platform offering
- Organizations using HEROES will be able to deploy a persistent infrastructure on the CSP platforms. This is to avoid deploy / undeploy of common resources with high deployment overhead (e.g., private network subnet, parallel filesystems, private license servers, etc.). Compute nodes will be on-demand to address the workload variations and limit cost & energy consumption.
- Single users of the HEROES platform will use short-lived, on-demand cloud compute nodes. The central HEROES infrastructure will manage scheduling and workloads directly.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.2.2 Overview

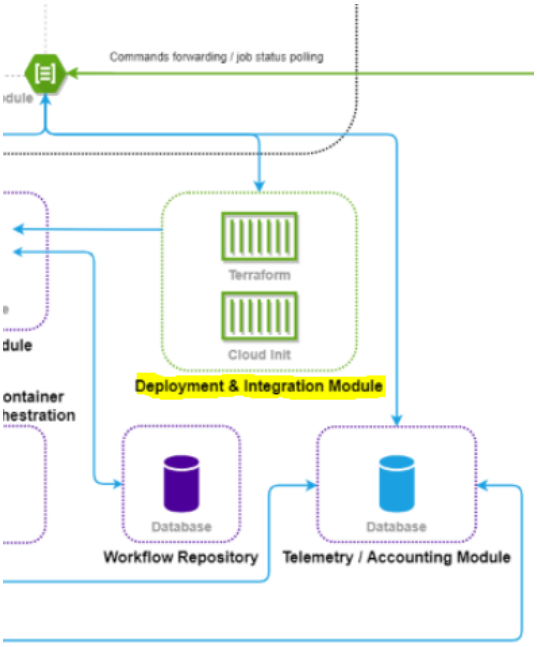

Figure 8 - Deployment & Integration Module architecture

The Deployment & Integration module will make use of software, scripts, configuration files and inputs from the Workflow Module to manage the remote components of the HEROES architecture.

For CSP resources, the Deployment & Integration Module will take care of:

- Activate the necessary quantity and type of instances (Terraform [12])
- Configure the "raw" OS (Operating System) of the instances with all the required parameters for full integration with the HEROES platform (CloudInit [10]), including the deployment of the HEROES Runtime (bundle of HEROES libraries, scripts, and tools)
- Ensure that the activated instances are ready for use (health check script)
- Perform termination of unused/unnecessary instances

For HPC Centres, the Deployment Module will take care of:

- Check the HPC Centre environment for prerequisites
- Deploy, activate, and configure the HEROES Runtime on the remote Headnode (see section 4.2.1)

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.2.3 Internal interfaces & interactions

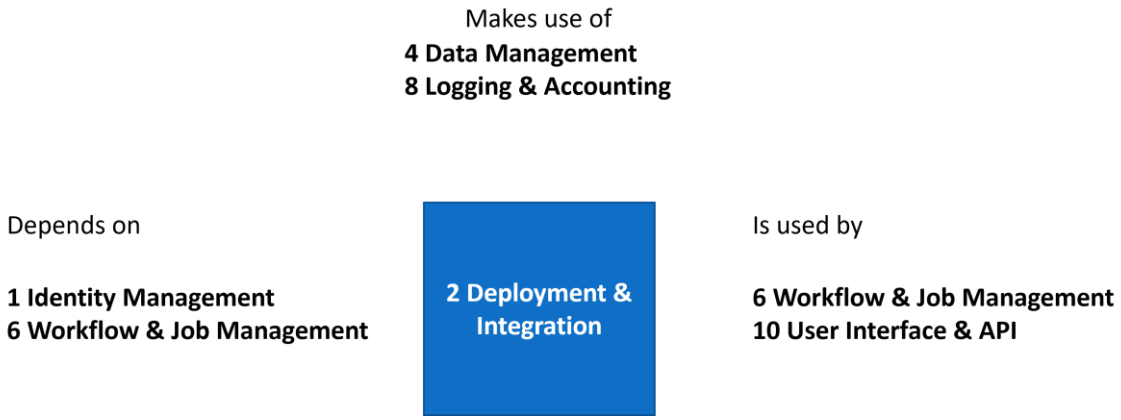

Figure 9 - Deployment Module internal interfaces & interactions

The Deployment & Integration module will interact with the other HEROES platform modules via API.

Identity Management:

- User identification (check User's id, role, group/project/organization)
- Authorization (check if the user has access to the remote platform)
- Endpoint authentication (check if the API call was initiated from a legitimate endpoint within the HEROES platform)

Workflow & Job Management:

- Receive tasks with detailed requirements to deploy / undeploy remote cloud instances
- Send events with status of all executed task

Logging & Accounting:

- Send events with details about all received requests/tasks
- Send events with status of all executed task

4.2.4 External Dependencies

Third-party Software:

- Terraform [12]: software used to deploy the remote cloud infrastructure with a consistent configuration
- CloudInit [10]: software used to perform consistent configuration of remote cloud instances

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- Ansible [13]: software for the automation of installation and configuration processes of the various components of the HEROES Runtime platform
- Cloud Platform CLI (Command Line Interface) tool / API: software used to interact with the remote cloud platform (e.g., AWS CLI [37] for AWS [1], OCI CLI [38] for Oracle Cloud Infrastructure [2])

4.2.5 Expected outcomes & behaviour

The Deployment & Integration Module main objective is to implement / execute tasks that will be sent from the Workflow Management module to ensure that the resources are available for receiving ML/HPC jobs. The following types of incoming tasks are expected:

- Deploy / undeploy cloud compute resources
- Check if HPC Centre HEROES resources and services are active and correctly configured

The Deployment & Integration Module will also contain a library of Ansible [13] playbooks that will be used to install and configure the HEROES Runtime both in the cloud platforms and in the HPC Centres.

4.2.5.1 Deployment workflows

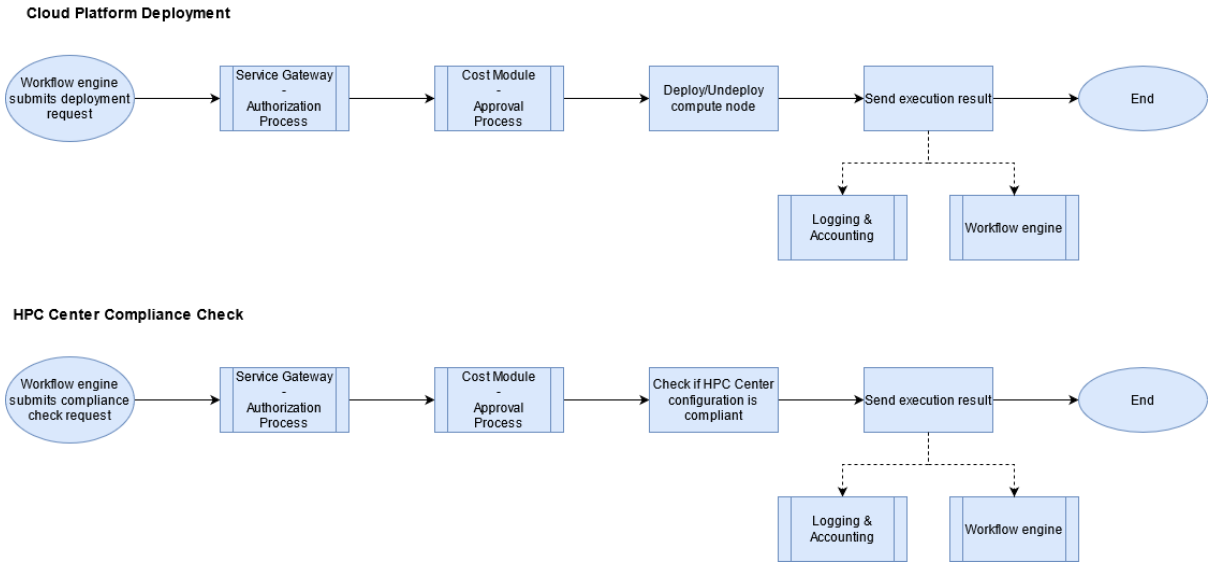

Figure 10 - Cloud Platform Deployment workflow & HPC Centre Compliance Check workflow

The **Cloud Platform Deployment workflow** will take care of:

- Start / Stop compute nodes on existing Private Cloud Clusters within an Organization scope

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- Deploy / undeploy temporary compute nodes for single users and organizations without an existing Private Cloud Cluster. *(Please note that this will cover all types of different compute nodes, including but not limited to CPU nodes, GPU nodes, Remote Visualization nodes.)*
- The above tasks will be accomplished using the Cloud Platform CLI and APIs, mainly through scripts. The scripts will be exposed to the Workflow module either directly or via APIs.

The **HPC Centre Compliance Check workflow** will take care of:

- Check that the HEROES Runtime is active on the HPC Centre Headnode (see section 4.2.1)
- Check that the HEROES Runtime is configured as expected
- The above tasks will be accomplished using Ansible roles, containerized applications and services, scripts. The Ansible roles will be exposed to the Workflow module either directly or via APIs.

4.2.5.2 HPC Centre Enrolment workflow

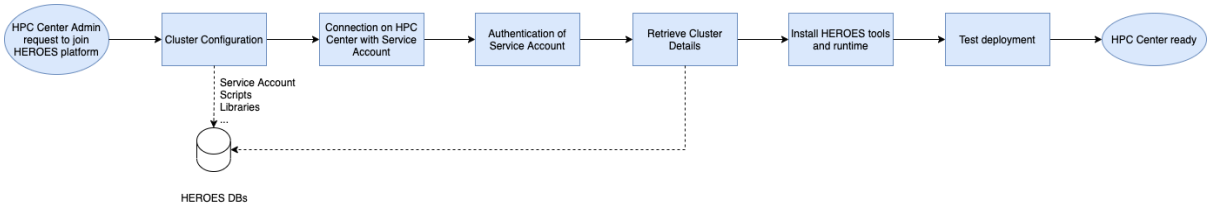

Figure 11 - HPC Centre Enrolment workflow

The HPC Centre Enrolment workflow will be "manually" executed by the HPC Centre admins, supported by the HEROES Support / Admin Team.

The HEROES Runtime package will contain the components needed to integrate the HPC Centre with the HEROES platform. All installation & configuration tasks will be handled with Ansible [13] roles to ensure that all components will be configured as designed.

4.2.5.3 Cloud Platform Integration workflow

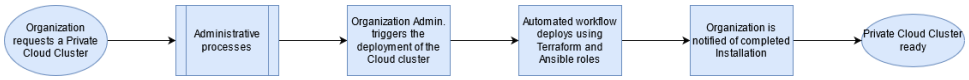

Figure 12 - Cloud Platform Integration workflow

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The Cloud Platform Integration workflow will be manually triggered by an "administrator" of the client Organization to create a complete Private Cloud Cluster on behalf of an Organization. Creation of static cloud infrastructures will be handled by an automatic procedure based on Terraform. All installation & configuration tasks of static cloud instances and / or resources will be handled by automatic procedures, based on Ansible roles, to ensure that all components will be configured as designed. Temporary compute nodes will be handled with the Cloud Platform Deployment workflow (see Section 4.2.5.1).

4.2.6 Uncertainties

At the time of writing this document uncertainties include, but are not limited to:

- Terraform and Ansible playbooks could be automatically or manually run by HEROES Support / Admin Team to create Private Cloud Clusters. The on-premise deployment on HPC Centre could be manually run by HPC Centre admins, supported by the HEROES Support / Admin Team, to install the required components on their premises and perform prerequisite checks. On the other side the HEROES Runtime will be pushed by the HEROES platform whenever needed.
- Which CSP platforms will be involved (initial target are AWS [1] and OCI [2])
- Configuration and type of communication channels between the HPC Centres Headnodes and the HEROES platform (aside from the assumed SSH channel)
- Configuration and type of communication channels between the HPC Centres Headnodes and the public internet (e.g., to recover software from public repositories)

4.3 Module 3 – Energy Management & Optimization

Energy monitoring, management and optimization is key for an energy efficient system. In the HPC context, reducing the energy consumption also implies the additional requirement of minimizing the performance impact. The EAR [14] framework is going to be used as the energy management framework for energy monitoring, management, and optimization. EAR will be one of the data sources that will be used by the Decision Module presented in Section 4.7: EAR will take care of "local" energy optimization (at a job level) whenever possible (see Assumptions below), and the Decision module will take global decisions on where to place the tasks depending on the energy requirements.

4.3.1 Assumptions

We have designed the HEROES architecture with the following assumptions regarding energy management and optimizations:

- We must support scenarios where EAR is already installed or not installed on the target HPC Centre. In all the scenarios we should be able to provide, at least, monitoring, and local optimization when EAR is fully deployed (which requires additional privileges).

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- We must be able to provide different levels of monitoring, accounting and optimization depending on what we are allowed to measure

4.3.2 Overview

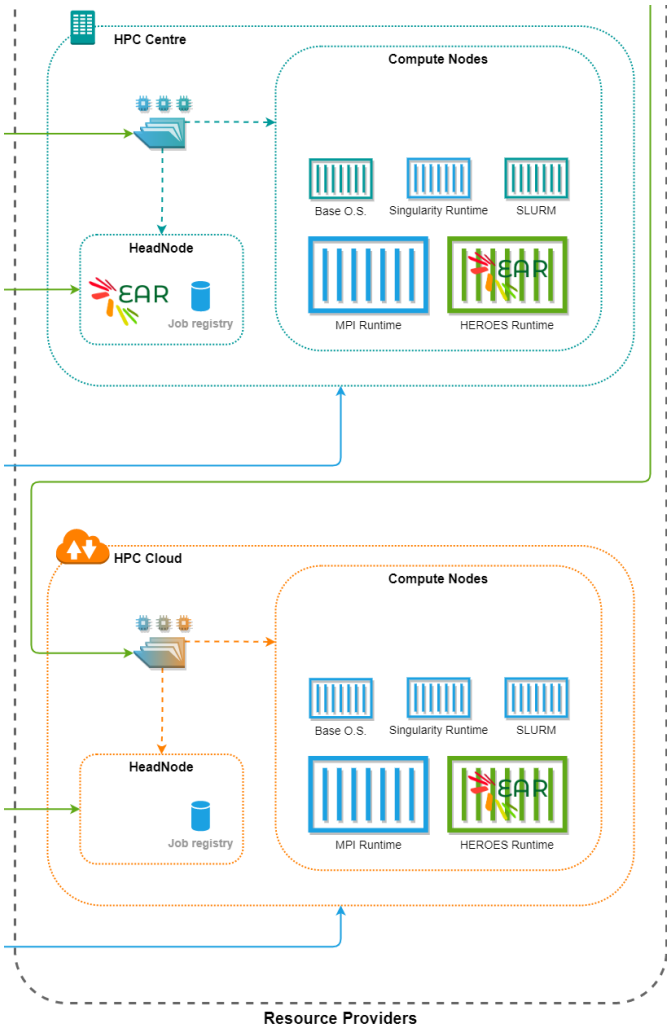

Figure 13 - Energy Management integration in HEROES

Figure 13 shows the integration of the energy management in HEROES. EAR will be deployed automatically in cloud nodes. In HPC Centres it will depend on whether EAR is already installed or not, and if we have sufficient rights to install it entirely or only for monitoring. In any case, each provider will specify which "version" must be used: from the most lightweight where EAR will be running as a runtime, to the most powerful where EAR will be running as a service, and we will be able to gather system metrics and optimize dynamically the energy consumption by changing the CPU frequency. EAR data is reported in an SQL database (DB). During the project we will extend EAR to report data with different technologies to make it possible to transport this data to HEROES' metadata DB, see Section 4.7.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Even though in the picture EAR is shown as a single component for simplicity, it includes the following components (see Figure 14):

- EAR library (EARL): It's a runtime library responsible for application monitoring and energy optimization. EAR library will be extended to support cloud environments with limited access. The EAR library computes the ***application signature*** with different granularities. The application signature is a set of metrics including all the information collected by EAR library. This set of metrics can be used to classify the application, to model it, etc. EAR computes application signatures per node and per regions, where one particular region includes the whole application. Application regions, loops, are automatically detected by the library at runtime.
- EAR daemon (EARD): It is a Linux service executed with root privileges. EAR library relies on EARD to gather privileged metrics and report data. The EAR library will be extended to support scenarios where EARD will not be present (cloud and HPC Centres where EAR is not officially installed)
- EAR DB manager (EARDBD): EARDBD reports data to the DB server. It will be only present on HPC Centres with EAR already installed. It can be used as a proxy to report data to HEROES' DB.
- EAR Global Manager (EARGM): It is a Linux service for global monitoring and optimization. It will be only present on HPC Centres where EAR is already installed. It could be used to report global cluster status.

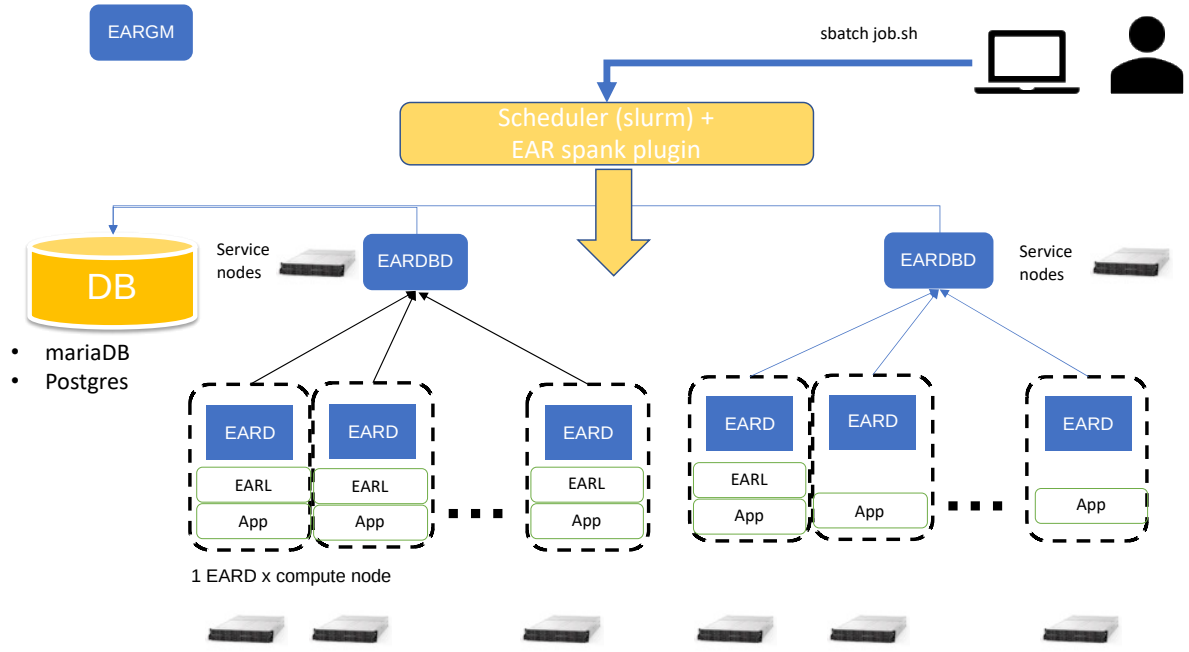

Figure 14 - EAR architecture

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.3.3 Internal interfaces & interactions

Interaction from cloud or HPC nodes to outside will be done though SSH to guarantee the deployment in HPC Centres with strong connectivity limitations.

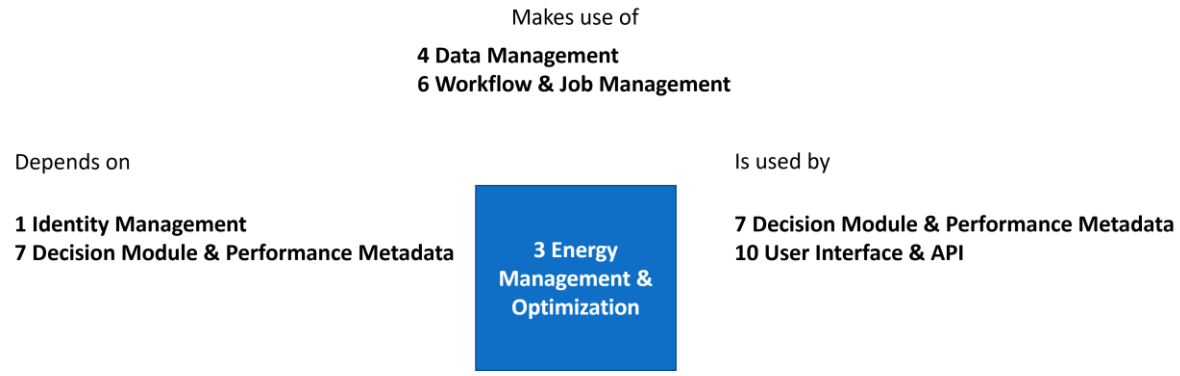

Figure 15 - Energy Management & Optimization internal interfaces & interactions

EAR will gather 3 types of data that will be used for HEROES accounting and the Decision module:

- **Job accounting**: Performance and energy metrics corresponding with HEROES jobs. This information is gathered at runtime, and it can be reported at application end or periodically depending on system access limitations. It will include the data for job accounting (identification and resource consumption) and processed information with the application runtime characteristics that can be useful for the decision module. Ideally, this information could be the runtime signatures of the application but, depending on the availability and the amount of data we could expect to send and store a kind of application profile and not all the signatures computed.
- **System monitoring**: System monitoring is done by EAR at intervals specified by the sysadmins. In HPC Centres it is important to report as many details as possible for system analysis and reporting. However, in the case of HEROES, what is more relevant is to know the system status. For HPC Centres where EAR is already installed, we will aggregate the information for the whole cluster in a single record, either based on EARGM information or aggregated metrics from the DB. For HPC Centres where EAR is not installed, we will provide a light version of the service.
- **Events**: Events are generic information potentially associated with a job. What is relevant for HEROES are those events from the system that could provide relevant information for the decision module such as systems with lots of failures, or job events such as energy optimization problems (cases where the energy model fails on its predictions). This kind of hint can be useful for the decision module to discard a given centre.

EAR will depend on Deployment (see Section 4.2) and Data Management (see Section 4.4) modules and will be the main source of data for the Decision module (see Section 4.7).

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Logging & Accounting:

- Send events with details about all received requests / tasks
- Send events with status of all executed task

Identity Management:

- Service identification (check Service's id, role)
- Authorization (check if the service has access to the remote platform)
- Endpoint authentication (check if the API call was initiated from a legitimate endpoint within the HEROES platform)

4.3.4 External dependencies

Having a full EAR installation requires a DB server and some basic OS kernel modules such as MSR (Model-Specific Register) or perf [41]. However, we don't have dependencies from other tools or services, and thus the requirements can be lowered in a "lightweight" deployment.

4.3.5 Expected outcomes & behaviour

As stated before, the expected outcome for this module is:

- Dynamic and transparent energy optimization for running jobs. EAR will dynamically analyse running jobs and will select the optimal frequency. HEROES will apply a default policy for non-expert users and will give the opportunity to select a specific energy optimization policy to energy-aware users. Anyway, system specific settings will be considered. Energy optimization policies will be applied when supported by system limitations.
- Automatic job accounting: Metrics gathered by EAR will be reported to HEROES DB. Not all the data will be reported, only data needed for HEROES job accounting and for the decision module. Given a job could generate lots of signatures at runtime, we will select relevant data such as 1 signature per application phase and phase duration.
- System monitoring: EAR will report system status and relevant metrics / events to the decision module to characterize our ecosystem.

4.3.6 Uncertainties

At the time of writing this deliverable, it is unclear what will be possible to offer in the EAR-light versions. It will depend on specific configurations for clouds and HPC centres. Anyway, we will automatically detect the limitations and we will offer application monitoring, system monitoring (if possible). and energy optimization (if possible). We assume application monitoring (for accounting) will be always possible with more or less details.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.4 Module 4 – Data Management

The Data Management module purpose is to perform secure, encrypted, authenticated data transfers between the various storage endpoints within the HEROES platform.

4.4.1 Assumptions

At the time of writing this document, the following assumptions are made:

- HPC Centres will provide storage for HEROES platform data
- Storage within HPC Centres will be accessible only with SSH / SCP data transfers (potentially using this as an encapsulated protocol, as is the case with RClone [15] – see below)
- Users will be able to directly access data only from the HEROES web interface
- Users will be able to access and interact only with data stored in the HEROES Central Archive

4.4.2 Overview

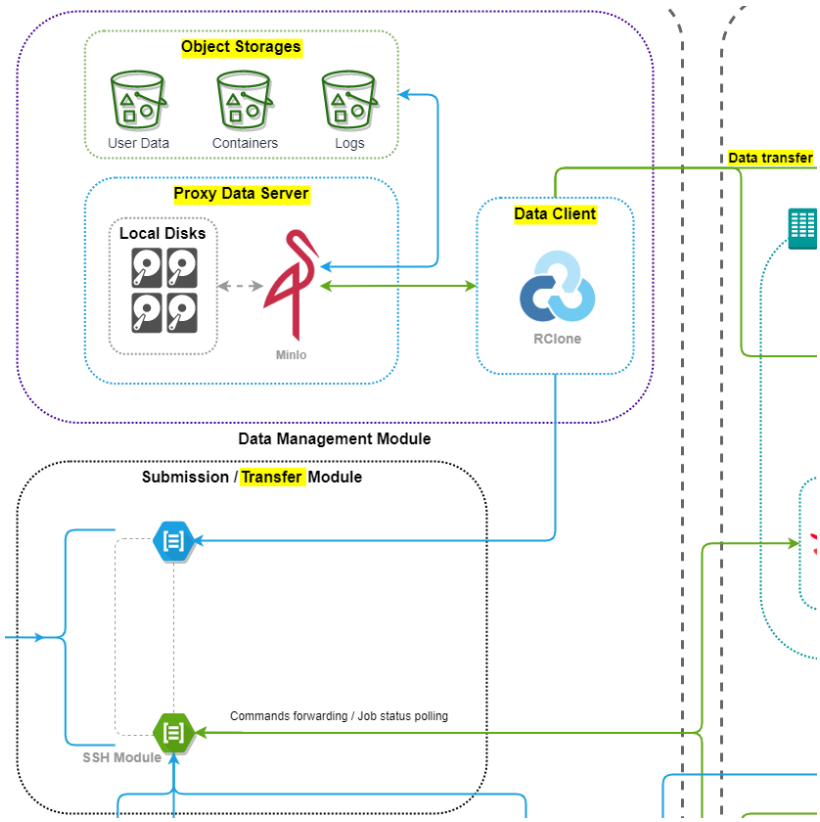

Figure 16 - Data Management Module Architecture

The Data Management module architecture is composed by:

- **Transfer Module API**: main interface to call for data transfer / access execution

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- **Data Client**: based on RClone, will take care of actual data transfer execution
- **Proxy Data Server**: scalable storage component based on MinIO; used to directly manage block devices attached to one (or more) cloud instance(s) and integrate object storage services managed by the cloud platforms
- **Object Storages**: backend services managed by the cloud platforms and accessed through the Proxy Data Server

Data inside the HEROES platform are categorized as:

- **User Data** - data that:
  - have been uploaded to the platform by the Users
  - have been created as direct output of a workflow / job
  - have been manipulated directly by the Users within the platform
  - Users can retrieve from the platform.
- **Internal Data**:
  - data that have been generated within the platform, by the platform's components (e.g., logs)
  - data and metadata that have been created as collateral output of a workflow / job (e.g., debug logs, performance metadata)
  - data that are created within the platform by the platform administrators (e.g., scripts)
  - data that are part of the platform (e.g., HEROES runtime, configuration files)
- **Public Data**: data that are publicly accessible from the HEROES platform website (e.g., brochures, web pages…)

The Data Management module will perform automated transfers between the following storage endpoints:

- **HPC Cloud**: local storage on cloud instances; shared storage within an organization's private cluster inside a cloud platform.
- **HPC Centre**: shared storage within the HPC Centre infrastructure
- **Central Archive**: HEROES Platform endpoint data storage, managed with MinIO [16] and accessed through the Proxy Data Server

All the above endpoints are part of the HEROES platform. The Central Archive will be a unique endpoint for the whole architecture; there will be as many HPC Cloud and HPC Centre endpoints as needed.

The Data Management module will oversee all automated User Data and Internal Data movement operations within the HEROES platform, with the following exceptions:

- User Data that are uploaded via the main HEROES interface to the platform's Central Archive
- User Data that are downloaded via the main HEROES interface from the platform's Central Archive

The above User Data transfers will be managed from the UI Module by making use of direct access to the Proxy Data Server with MinIO's own APIs.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Please note that manual, Internal Data transfers will be possible, but only for Platform Administrators with direct access to the infrastructure. All manual data transfer should be performed over SSH, or other TLS encrypted protocol.

4.4.3 Internal interfaces & interactions

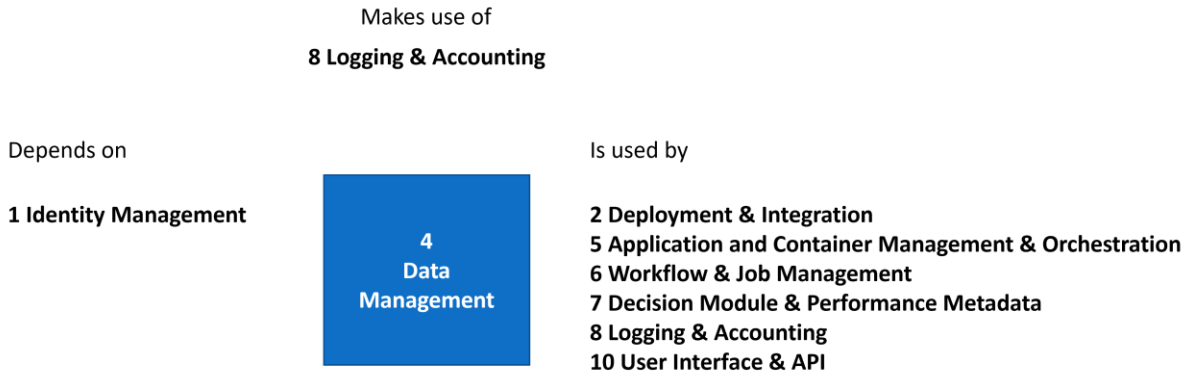

Figure 17 - Data Management Module internal interfaces & interactions

Identity Management:

- User identification (check User's id, role, group / project / organization)
- Authorization (check if the user has access to data transfer source and targets)
- Endpoint authentication (check if the API call was initiated from a legitimate endpoint within the HEROES platform)

Storage endpoints:

- Source and target endpoints must be defined within the HEROES platform and uniquely identified with a cryptographic certificate

Logging & Accounting:

- Send events with details about all received requests / tasks
- Send events with status of all executed task

The Data Management module will expose its functions via API to the other modules of the HEROES platform.

Endpoint authorization will be managed by the Identity Management Module.

All Data Transfer operations will be logged using the Logging Module API functions, whether successfully completed or not.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.4.4 External dependencies

Third-party Software:

- RClone: software used to transfer data; used in the main program of the Data Management module
- MinIO: object storage platform for hybrid cloud; used for the backend storage on HPC Cloud and HEROES Central Archive

4.4.5 Expected outcomes & behaviour

The Data Management Module will be the core of the following workflows.

4.4.5.1 Service Data Management Workflow

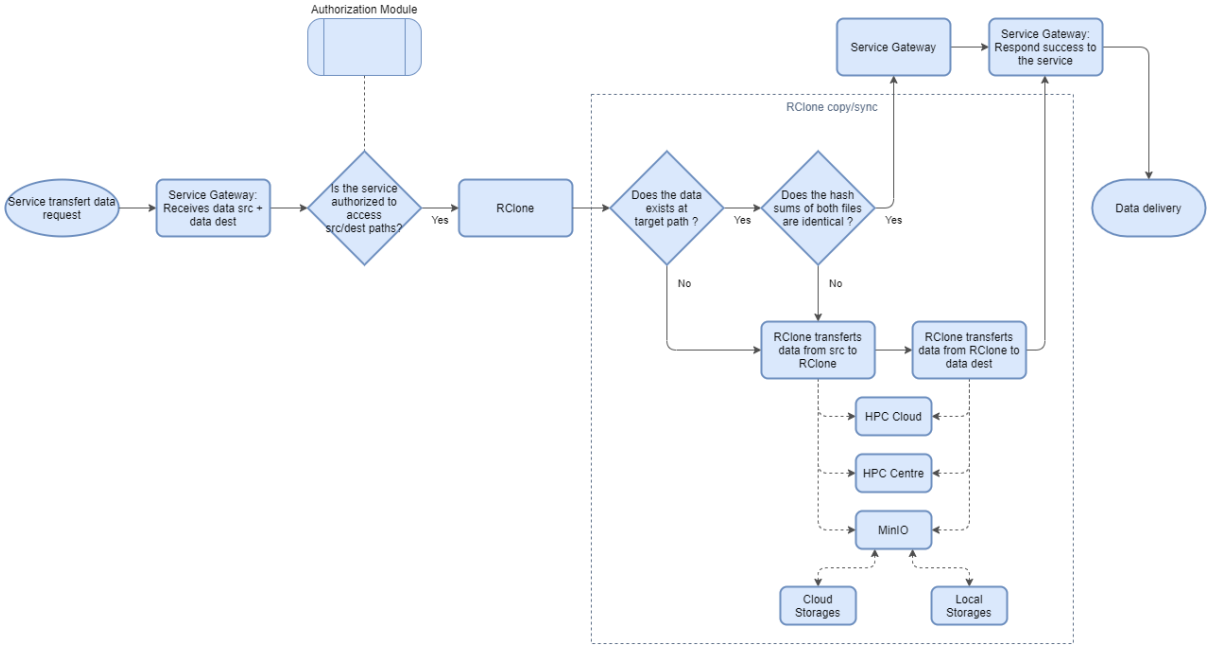

Figure 18 - Service Data Management workflow

The Service Data Management workflow takes care of all automated User Data and Internal Data movement operations within the HEROES platform.

The main component is RClone: by making use of its features the workflow first checks if the target data exists on the target as an exact copy. If not, the source data is transferred to the target. The operation result is then sent to the Service Gateway together with any error code or extended message.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.4.6 Uncertainties

At the time of writing this document uncertainties include, but are not limited to:

- Type and size of storage endpoints
- Usage of quotas for enforcing a limit in user data consumption
- Duration of a data transfer (depending on size of data, the available bandwidth, and its distance between the transfer points)

4.5 Module 5 – Application and Container Management & Orchestration

The Application and Container Management & Orchestration purpose is to perform secure and versioned management of the various containers within the HEROES platform.

4.5.1 Assumptions

At the time of writing this document, the following assumptions are made:

- The user application containers will be based on Singularity [39]
- The application container images will be built by the customer at the customer's site and then uploaded to the HEROES platform
- "Application packager" role will be able to publish his applications to make it accessible to specific organization, project, group, user or public as target.
    - This role may be given to:
        - Customer organization administrators in charge of the application of their organization
        - Application Provider in charge of application publishment of their organization
- Users who wish to publish their applications must test the correct application behaviour before publishing to the platform

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.5.2 Overview

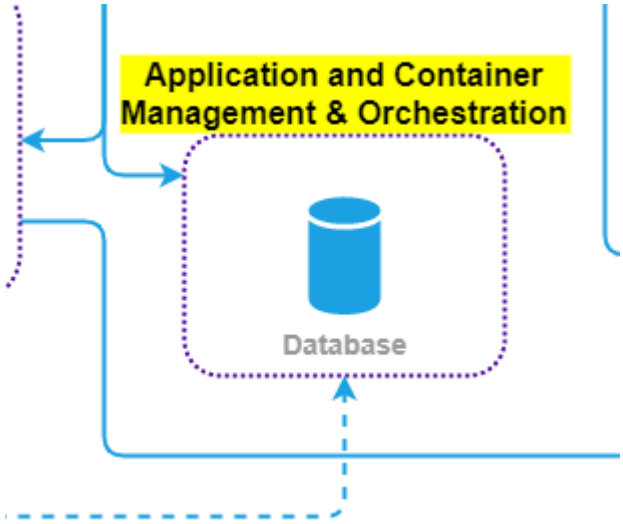

Figure 19 - Application & Container Management & Orchestration architecture

The needs of Application and Container Management & Orchestration have been identified as:

- The storage of the application information, including:
  - Its name.
  - Its owner (the name / unique id of the application packager).
  - Its metadata and parameters.
  - Its publishment information, including:
    - Its price and licence.
    - The specific organization, project, group, user or public as target.
  - Its publishment status
- The needs of security:
  - Each application / container needs to be encrypted at rest, in transit and while running to ensure confidentiality of the containers.
  - Prevent “spam bombs / botnets” by blocking the outgoing traffic from the containers to ensure the availability of the service
  - The containers must be immutable to ensure their integrity

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.5.3 Internal interfaces & interactions

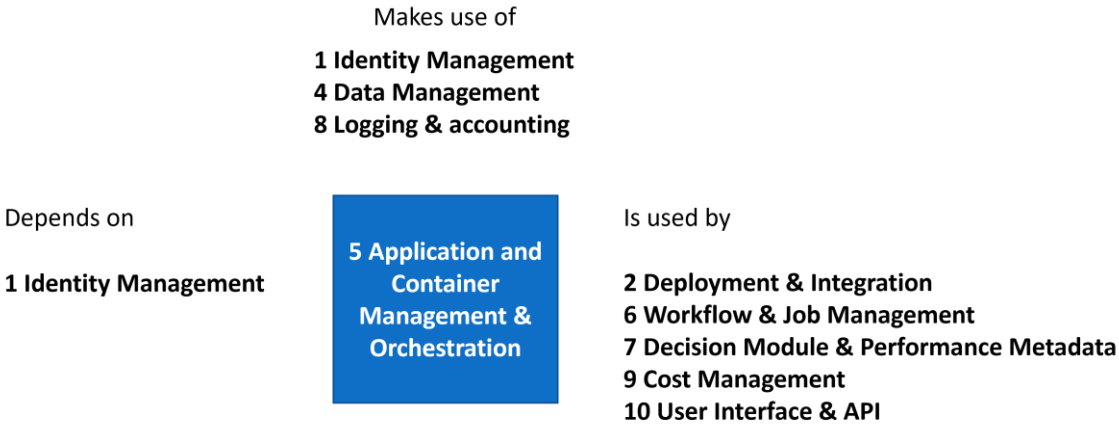

Figure 20 - Application and Container Management & Orchestration internal interfaces & interactions

The Container Management & Orchestration will interact with the other HEROES platform modules via API.

Identity Management:

- User identification (check User's id, role, group / project / organization)
- Authorization (check if the user has access to the remote platform and to this service)
- Endpoint authentication (check if the API call was initiated from a legitimate endpoint within the HEROES platform)

Data Management:

- Uploading container from the User to the platform
- Container storage within the HEROES platform including:
  - Versioning of the container
  - Tags linked to the stored container

Logging & Accounting:

- Send events with details about all received requests / tasks
- Send events with status of all executed tasks

4.5.4 External dependencies

Third-Party:

- Singularity is a container platform. It allows to run containers that package up pieces of software in a way that is portable and reproducible.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.5.5 Expected outcomes & behaviour

The Application and Container Management & Orchestration Module main objective is to allow the creation, management, and orchestration of application. The following types of incoming tasks are expected:

- Create / Delete application
- Update application
- Publish / Unpublish application

The application packager (organization application administrator or Application Provider) can create an application within the HEROES platform using a container.

The application creation includes:

- The application information will be stored in a database within the Application and Container Management & Orchestration module.
- The storage of the application's container will be managed by the Data Management module

After its creation, application will become visible for modifications or publication for its owner.

The owner of an application must make functional tests within the HEROES platform with its application before each new application publishment.

After publishment, the cost and license of the published application will be available for the Cost Management module.

4.5.5.1 Application referencing flow within the platform

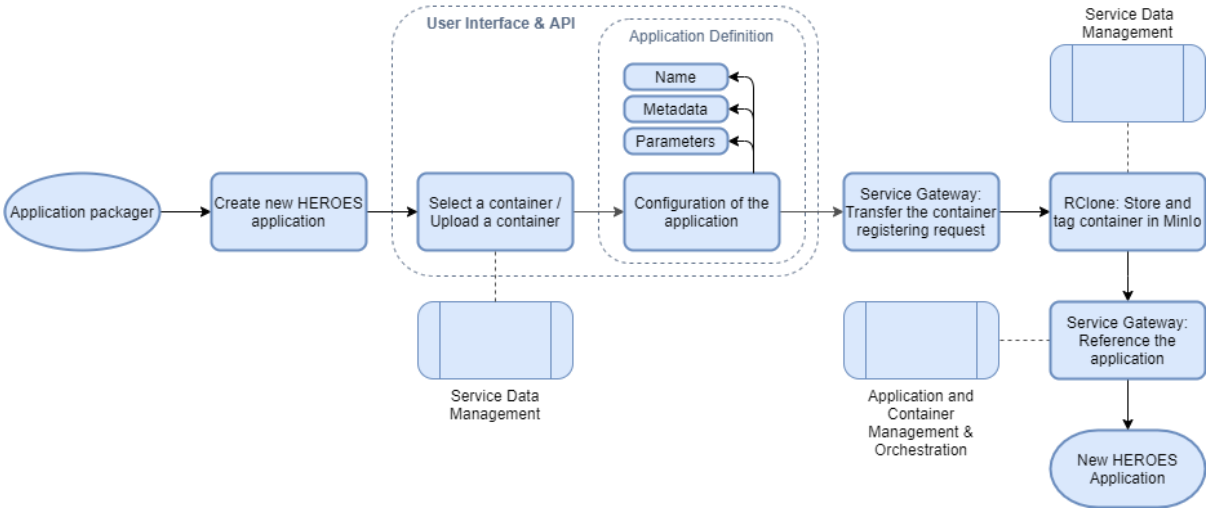

Figure 21 - Application referencing flow within the platform

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The Application referencing flow (as defined in Figure 21) starts by an “Application packager” creating a new HEROES application and ends by a new HEROES application in the platform:

- “Application packager” sends a request of new HEROES application through the UI or with API including:
  - A selected or uploaded container for the new application.
  - A configuration for the application, including but not limited to: Name, Metadata and Parameters.
- The “Service Gateway” transfers the container registering request to the Service Data Management which will store and tag the container.
- At the end, the Service Gateway references the application in the Application and Container Management & Orchestration module.

4.5.5.2 Application updating and publishing flow within the platform

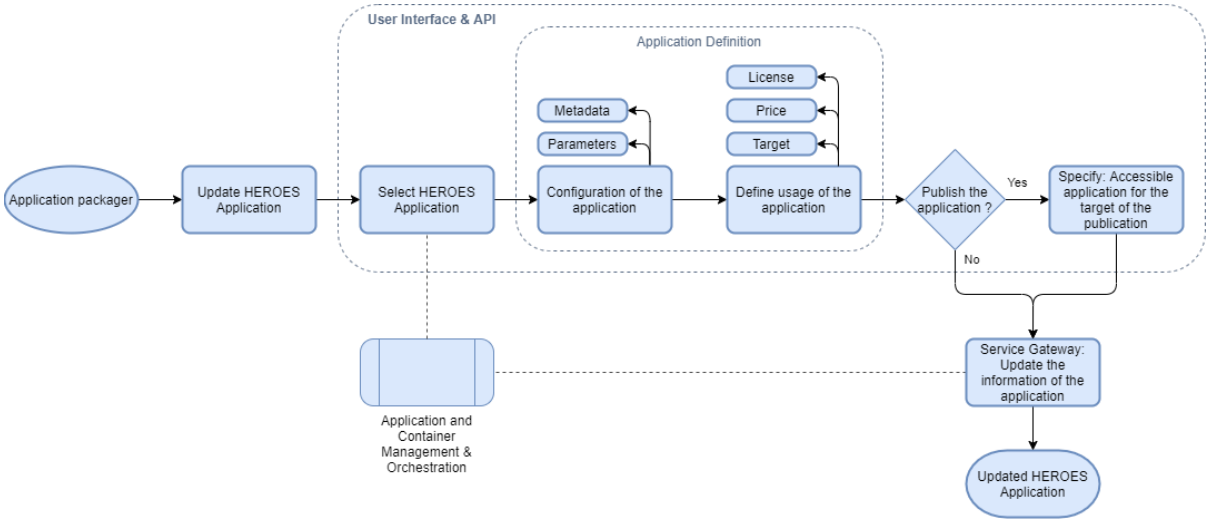

Figure 22 - Application updating and publishing flow within the platform

The Application updating and publishing flow (as defined in Figure 22) starts by “Application packager” updating a new HEROES application and ends by an updated HEROES application in the platform:

- “Application packager” sends a request of HEROES application update through the UI or with API including:
  - The selected application name / id to identify the correct application.
  - Optional modifications of the application configuration (Metadata and Parameters).
  - Optional license, cost, and target as required parameters for the application publishment.
  - Optional application publishment with its required parameters.
- At the end, the Service Gateway updates the information of the application in the Application and Container Management & Orchestration module.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.5.6 Uncertainties

The current uncertainties include but are not limited to:

- How will user receive the role of “Application packager”? Do we dissociate an application packager from a customer and from Application Provider?
- Should published applications be restricted to use by organizations, groups of users, or can they be public for everyone?
- Should published applications be restricted to use by organizations, groups of users, or can they be public for every organization within the HEROES platform?

4.6 Module 6 – Workflow & Job Management

The Workflow and Job management module purpose is to provide an easy interface for develop, orchestrate, execute, and register secure workflow submissions and pipeline executions.
The Workflow and Job management module is composed mainly by the Ryax [17] engine tool that provides a complete data engineering platform enabling functionalities for end users to deploy, run, and scale their models on production infrastructures.
The Workflow and Job management module could execute HEROES architecture internal processes and scientifical pipelines deployed by the end-users.

The needs inside the HEROES platform for a Workflows and Job Management module are:

- Automate, orchestrate, and execute scientific and engineering workflows customized and parametrized by the end-users.
- Automate, orchestrate, and execute workflows for HEROES architecture processes.
- Different levels of Workflows will be designed for the different stages of the pipeline.
- Export and import workflows from a Workflow Repository to reuse previously developed code.
- Monitor the workflows execution.
- Trigger the Deployment Module to generate and start the remote infrastructure, conducting every step of the workflow through HEROES technology to the optimal hardware needed.
- Obtain workflow placement recommendations based on the Decision Module to select the best environment to deploy the models.
- Save time on engineering workflows, automating, and optimizing the methodology to manage resulting files.
- Deploy containers with state-of-the-art tools to analyse data and to monitor the models and predictions.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.6.1 Assumptions

At the time of writing this document, the following assumptions are made:

- HEROES platform will need the definition, orchestration, and utilization of different types of workflows to manage automatically complex scientific processes and internal architecture procedures.
- Automatic process can be implemented, using users' scripts (batch postprocess) that can be edited to be launched in the Cloud or HPC Centres.
- A Deployment and Integration module will provide tools and interfaces to trigger the creation and monitoring both of on-premises HPC resources or the provisioning of cloud clusters over multiple CSPs if needed.
- A polling system from the Service Gateway will allow the HEROES platform to monitor the jobs status and execution on metrics for the end user.
- Workflow configuration will be accessible by HEROES. This will allow users with sufficient privileges to design different workflows, according to their process requirements.
- Modularity and extensibility of the solution for the workflows management will allow the development of new functionalities in a modular way, to easily allow the addition of new features or the support of third-party components.
- The Workflows module should propose execution based on performance and efficiency relying this selection on information gathered by the Decision Module.
- The platform should be easy-to-use by the end-users using a graphical interface, as they are not computer scientists so the interfaces should be intuitive to use it and obtain the best experience from the HEROES platform.
- Accounting application will be implemented and coupled to HEROES (e.g., to show the number of launched workflows per day, which application is being used the most, the used CPU resources...)

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.6.2 Overview

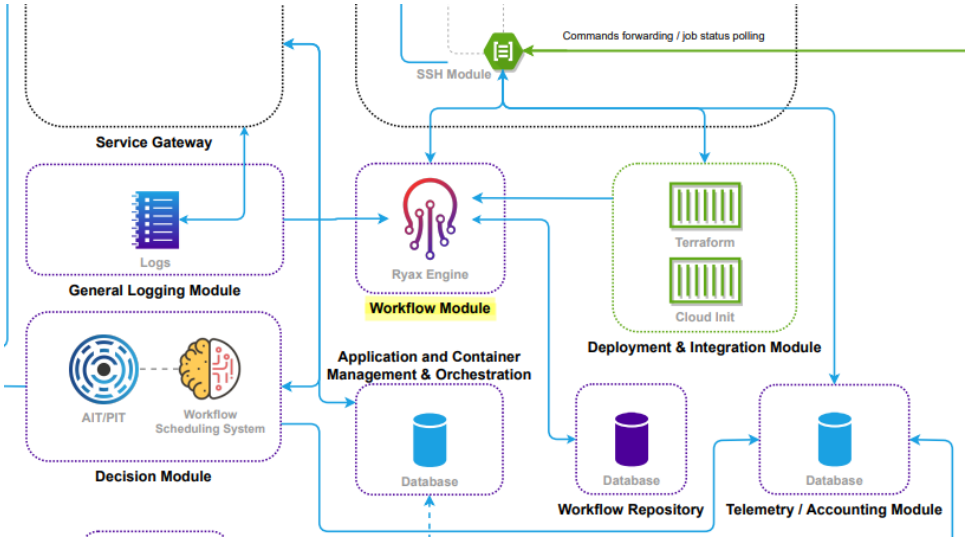

Figure 23 - Workflow & Job Management Module architecture

The Workflow and Job Management module will provide an internal service for the definition and execution of workflows: it will rely on the other HEROES modules to access computing resources (CSP and HPC Cluster) and transfer data (for its **Architecture Workflows**), and will dispatch each task of the workflows to the selected target infrastructure (for the **Engineering and scientific user workflows**). This workflow allows the automation and simplification of complex processes inside the platform for the end-user.

The Workflow module is composed by:

- **Ryax Engine** [17]: main tool to manage, automate and store all the pipelines and processing workflows inside the HEROES architecture. The tool interacts with the rest of HEROES modules to achieve the pipeline completion.
- **Workflow Repository**: a registry to store and classify available workflows for being used on the platform.

Workflows within the HEROES platform are categorized as:

- **Architecture workflows**: internal workflows to interact with different parts inside the architecture
  - Obtaining placement strategies by the Decision Module
  - Triggering the deployment, configuration of the Deployment & Integration Module for compute resource deployments
  - Creation and upload of new Singularity containers (models) to Application and Container Management & Orchestration Module

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- **Engineering and scientific user workflows**: workflows defined inside the infrastructure to trigger and deploy scientifical processes inside the compute resources
    - Workflow with a **single job**
        - Executed on the same cloud cluster or HPC Centre.
    - Workflow with **multiple jobs**
        - On the same cloud cluster or HPC Centre
        - On multiple cloud cluster(s) or / and HPC Centre(s)

The computational parts of the workflows described previously can be executed on different infrastructures or resources following different steps:

- **HPC Centre workflow** (on-premises computational resources)
    - Check compute resources availability
    - Push job dependencies (if not already present) in a directory specific to the user and job
        - Singularity container
        - HEROES Runtime
        - Input data
    - Job submission to the job scheduler
        - Execution using Singularity container in a secured manner (e.g., directory access, filesystem mount...)
        - HEROES Runtime execution in the container to monitor the job and gather metrics
    - Periodic job status polling
    - When the job ends
        - Retrieve results and output data, and store them in central HEROES data repository
        - Retrieve job metrics, and store them in the logging module

- **HPC Cloud workflow** (cloud provided computational resources)
    - Check cloud compute resources availability
    - Cloud Cluster compute nodes provisioning if needed, through the Deployment & Integration module if needed
    - Push job dependencies (if not already present) in a directory specific to the user and job
        - Singularity container
        - HEROES Runtime
        - Input data
    - Job submission to the job scheduler
        - Execution using Singularity container in a secured manner (e.g., directory access, filesystem mount...)
        - HEROES Runtime execution in the container to monitor the job and gather metrics
    - Periodic job status polling
    - When the job ends

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- Retrieve results and output data, and store them in central HEROES data repository
- Retrieve job metrics, and store them in the logging module
- Cloud Cluster compute nodes termination if they are no longer used, through the Deployment & Integration module

4.6.3 Internal interfaces & interaction

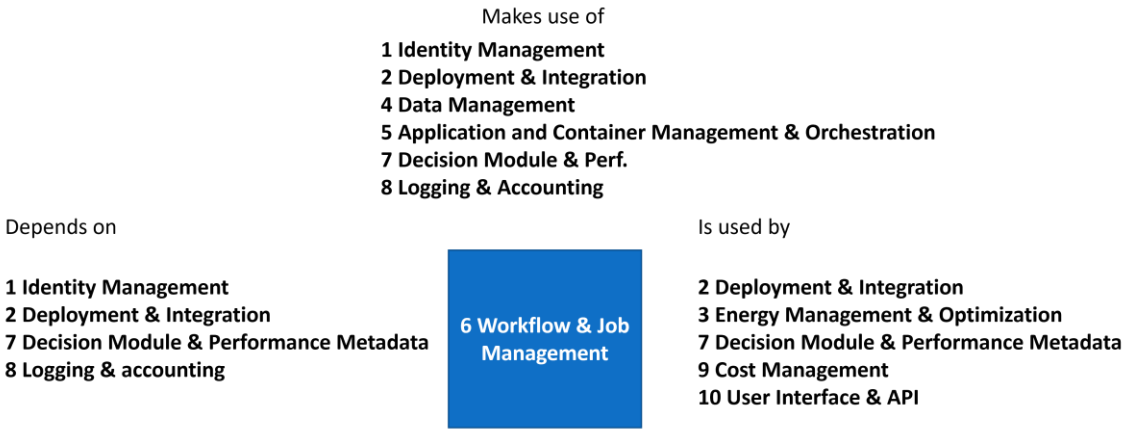

Figure 24 - Workflow and Job Management internal interfaces & interactions

The Workflow & Job Management module will expose its functions via API to the rest of the HEROES modules that need an interaction with it. Also, some steps on the workflows could call some module API endpoints or trigger customized scripts to interact to obtain other functionalities provided by other modules.

All the modules that need to interact with the Workflows and Job Management module API, requires to have a validation and authorization to make calls to the endpoint, providing a more secure protocol on the interaction.

The next section describes the main tasks performed by the modules in interaction with the Workflows and Job Management module:

- **Deployment and Integration Module:**
  - Forward commands securely through SSH
  - Script execution to deploy and configure clusters in CSPs

- **General Logging Module:**
  - Log workflow submission and job executions operations
  - Send events with details about workflows requests / tasks
  - Send events with status of all executed workflows
  - Gather information about the available infrastructures

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- **Submission and Transfer Module:**
  - Forward commands securely through SSH Module to Resource Providers
  - Send jobs to the Batch Scheduler through SSH Module to Resource Providers
  - Job status polling (check the status of submitted jobs) through SSH Module

Also, this module acts as intermediary to interact with other modules on the HEROES architecture, allowing the Workflows & Job Management module to obtain different information to complete the defined workflows.

The modules used through the Submission and Transfer Module are:

- **Data Management Module:**
  - Data control and data staging jobs with trough Data Management Module

- **Decision Module:**
  - Provides information for workflow placement strategies: selecting the "best" resources to run each part of the workflows, ensuring to meet constraints and SLAs (costs, performance, energy).
  - Provides information for job submission optimization: select the best combination of submission parameters (runtime, number, and type of requested resources…).
  - Provides constraints checking and cost and performance projections.

- **Identity Management Module:**
  - Interaction through Service Gateway
  - Access and authorization managed by the IAM module.
  - User identification (check User's id, role, group / project / organization)
  - Authorization (check if the user has access to the application / data)
  - Privileges validation
  - Endpoint authentication (check if the API call was initiated from a legitimate endpoint within the HEROES platform)

- **Workflow Registry:**
  - Store workflows available or developed by users
  - Provides a catalogue with the workflow's execution metadata, parameters, strategy, etc.

4.6.4 External dependencies

As the workflow submission and job management module is a quite complex part in the architecture, we require a tool set that covers all our needs and requirements bringing it all functionalities required to complete the different workflows and the integration with the HEROES architecture.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Third-party Software:

- Ryax is a data engineering platform enabling data teams to deploy, run, and scale their models on production infrastructures.
    - Allow workflows management via GUI / API / CLI
    - Allow integration with other tools / software via custom modules
    - Able to parallelize workflows steps basing their execution on containers over Kubernetes orchestrator
    - Extendible by Python code, allowing to create custom functions / modules
    - Easy interface to construct and deploy advanced data science workflows
    - Able to import Git repositories to export / import custom unitary modules or workflows

4.6.5 Expected outcomes & behaviour

The Workflow & Job Management Module main objective is to allow the creation, management, and orchestration of custom workflows, both for engineering and architecture management. The following types of tasks are expected:

- Create / Delete architecture / engineering / scientific workflows
- Run a specific workflow interacting with the rest of the HEROES architecture
- Log all the states and changes of the workflow
- Give proper metadata about the workflows state and execution

The Workflow & Job Management main tool (Ryax) can easily create workflows for the HEROES platform using their own GUI or importing it as code from an external repository.

The workflows creation and later execution include some interaction with other modules that are part of the HEROES infrastructure, where:

- The execution of the workflow could be triggered by the user directly using the **User Interface** via a web browser.
- The user can define the data used by the workflow through HEROES web interface or APIs. Then, the Workflow & Job Management module interacts indirectly with the Data Management module to transfer data to the target infrastructures and used by the different workflow tasks.
- All the workflow submission parameters are forwarded by the **Service Gateway** to the Workflow Management Module. It then uses them through the **Decision Module** to get some information about placement strategy for the workflow.
- The user chooses if he continues or not with the workflow execution via web browser using the **User Interface**.
- When the target infrastructure is selected, the **Deployment and Integration Module** deploys the required resources if needed.
- The toolset is selected from an available Singularity container on the **Application and Container Management and orchestration module**.
- The job is submitted to the Resource Providers using the **Submission / Transfer module**.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- The job is monitored using the **Submission / Transfer Module** and the obtained metrics are placed in the **Telemetry / Accounting Module**.
- The metadata retrieved from the execution and workflow status is saved in the **General Logging Module**.

4.6.5.1 Internal architecture workflow

The Workflow & Job Management module can store and execute workflows out of the engineering scope, adding an extra functionality for manage internal processes for the HEROES architecture.

These workflows can be automated or can be executed manually by HEROES administrators, performing desired tasks to maintain the platform, or just adding extra functionalities.

An internal workflow on the HEROES platform can be:

- Execute an infrastructure workflow, triggering different internal modules on HEROES, creating the target computational resources on a CSP, ensuring that the cluster exists and is configured before executing the scientific / engineering workflow.
- Execute an infrastructure workflow triggering different internal modules on HEROES ensuring that the proper directory tree is created on the target filesystem and copying later the data from the Data Management module to ensure that the data needed for the scientific workflow exist before forwarding their execution.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.6.5.2 Job Submission Workflow

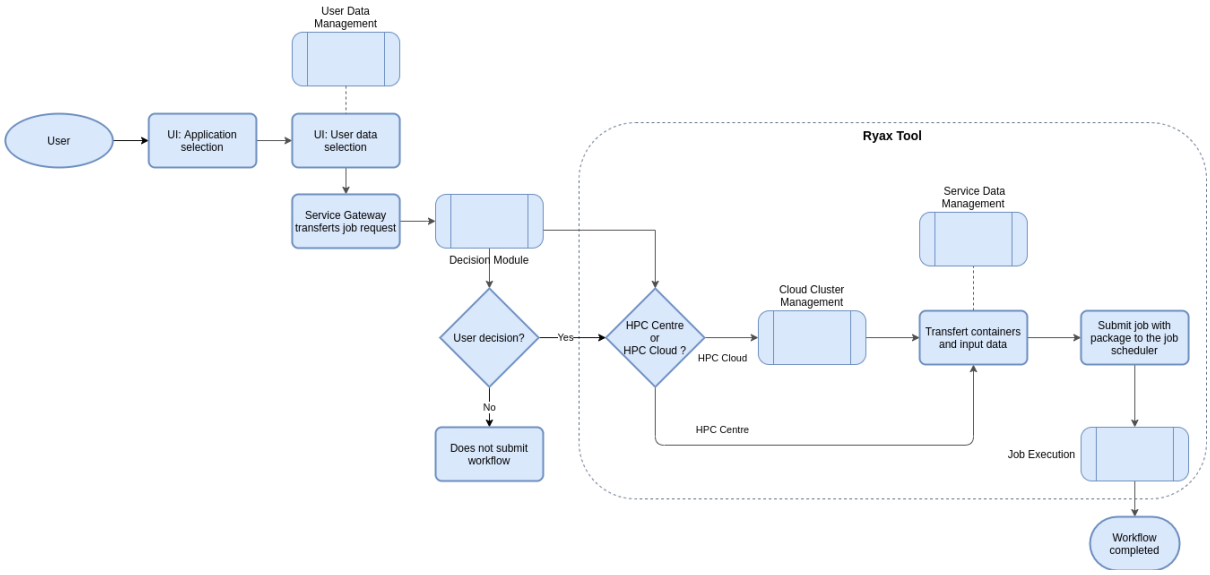

Figure 25 - Job Submission Workflow

The Job Submission Workflow as defined in Figure 25 starts by a user interaction and ends by a complete workflow being executed.

Those interactions can be summarized as:

- User application and data selection.
- The decision module analyses and predicts the optimal workflow decision depending on organization / project constraints on HPC Centre – HPC Cloud.
- User accepts or refuses the optimal workflow decision.
- A repetition of the analyses, prediction, and execution with its new optimal decision for each job of the workflow.
- Store the retrieved metrics and logs from the workflow's execution on the internal architecture databases / modules.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.6.5.3 Job Execution Workflow

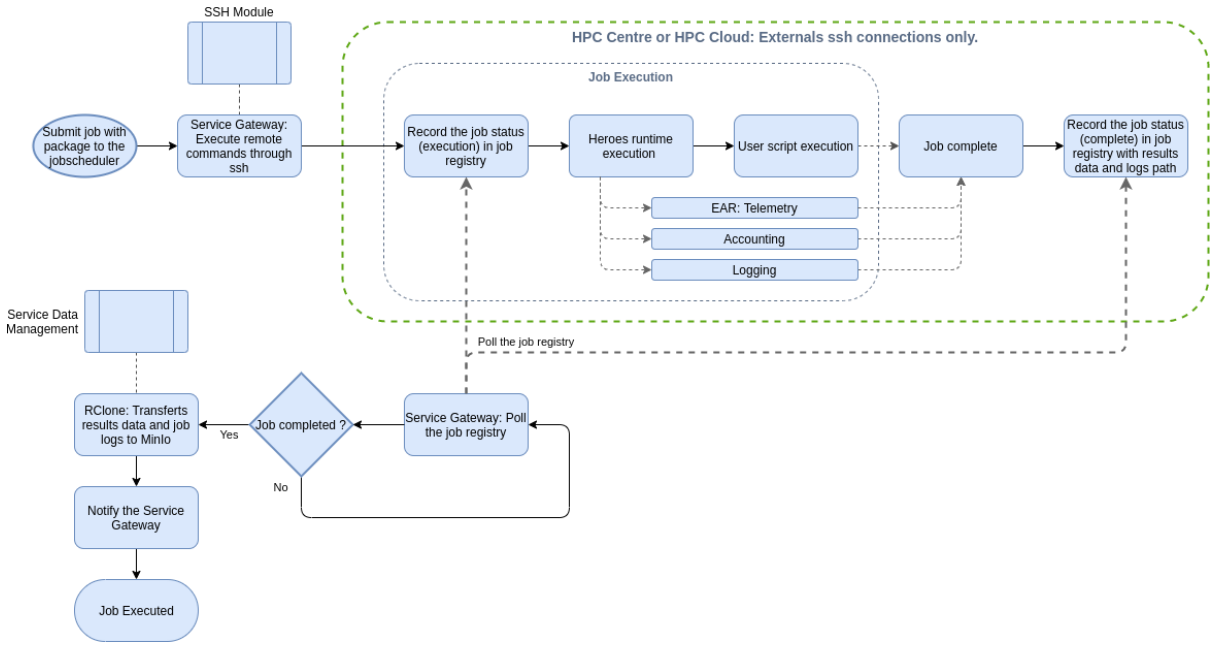

Figure 26 - Job Execution Workflow

The Job Execution Workflow as defined in Figure 26 is composed by:

- The job execution through the SSH Module to the HPC Centre – HPC Cloud (Resource Providers)
- The acquisition of metrics and valuable data from the job execution, as:
  - EAR Telemetry: hardware metrics read directly from the compute resources hardware as power consumption, frequencies, etc.
  - Accounting: information about compute resource consumptions, job durations, nodes involved, cores used, QOS, priorities, etc.
- Logging all the requests performed by the Workflows and Job module, obtaining a valuable metadata from it that later could be reused on the Decision Module (see Section 4.7)
- Monitoring of the job status using the Service Gateway polling, to obtain the job registries for each cluster to get the status of each job, trigger the next steps of the workflow when the job is completed.

After the job is finished, store all the logs, metrics and transfer all the data and results involved in the job using the Data Management Module.

4.6.6 Uncertainties

At the time of writing this document uncertainties include, but are not limited to:

- Job status polling: What solution will be used as job registry to store the job status of cluster(s).

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- Any defined budget (see Cost Management, Section 4.9) limit could be reached while the execution of workflow: What will be stopped in workflows and how it will be managed.
- Using remote visualisation to launch interactive sessions: the idea is to deploy a GPU node or use a visualization node on-premises resources to allow the use of graphical tools to review the results of the engineering workflows.
- How to control the versioning of modules and workflows created by the Ryax tools.

4.7 Module 7 – Decision Module & Performance Metadata

The decision module purpose is to select the best resources and platform to execute the users' workflows based on an optimization strategy.

The decision module is composed by:

- A job-level submodule that predicts cost and jobs resources (memory, execution time…) and power consumption. Predict-IT [36] (powered by OKA [35]) will be part of this submodule as the prediction-provider tool.
- A workflow-level submodule that evaluates the predictions and select the best platform to execute the workflow
- A database that stores metadata about all workflows and jobs executed by HEROES

4.7.1 Assumptions

At the time of writing this document, the following assumptions are made:

- Cloud cluster management: Each organization will have one cluster per CSP (with *x* types of instances, dynamic compute nodes and a file system)
- Platform management: Each platform will define their clusters availabilities for HEROES jobs
- Each organization will have access to a list of on-premises platforms and cloud clusters.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.7.2 Overview

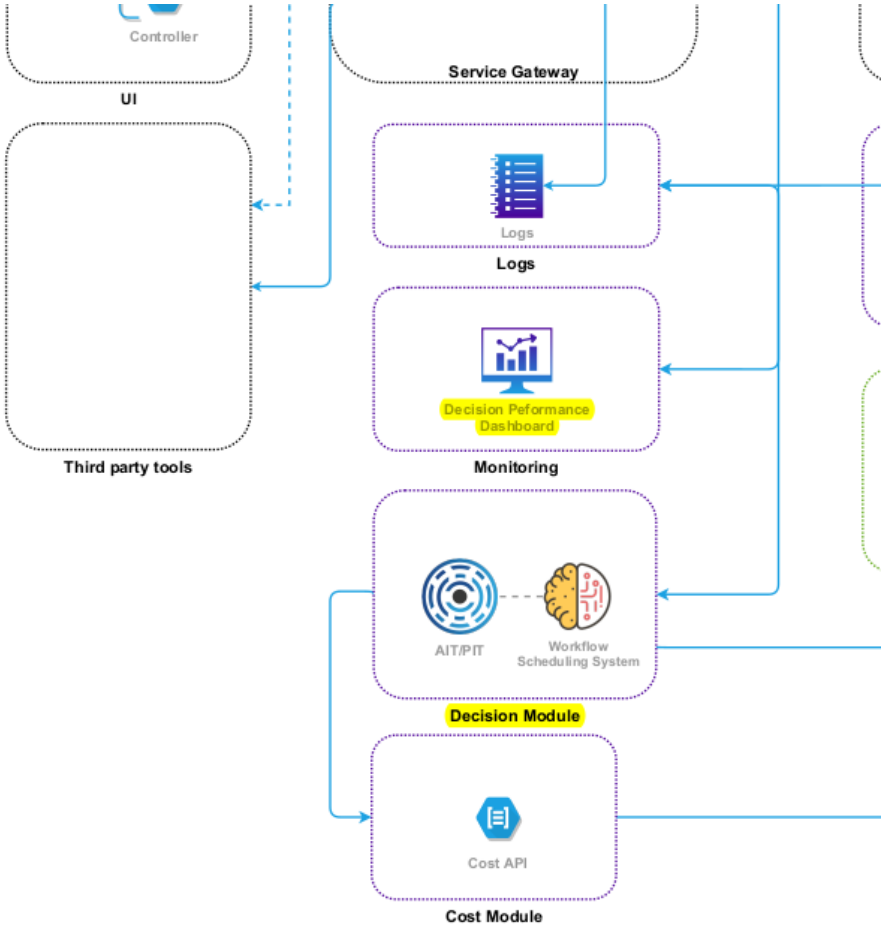

Figure 27 - Decision module

The decision module will be called multiple times during the workflow execution:

- At submission time it will predict jobs resources and the cost of the workflow execution on different platforms. These predictions gathered with the optimization strategy will be used by the decision module to propose the best platform on which to execute the submitted workflow
- Based on the user choice at submission time, the decision module will check the platforms occupancy to verify if another platform could be more suitable for each step of the workflow

The information will be shared to the UI / API and used to provide a dashboard which will allow the administrators to check the users' decisions and the decisions performances (predictions *vs.* the actual workload execution).

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.7.3 Internal interfaces & interactions

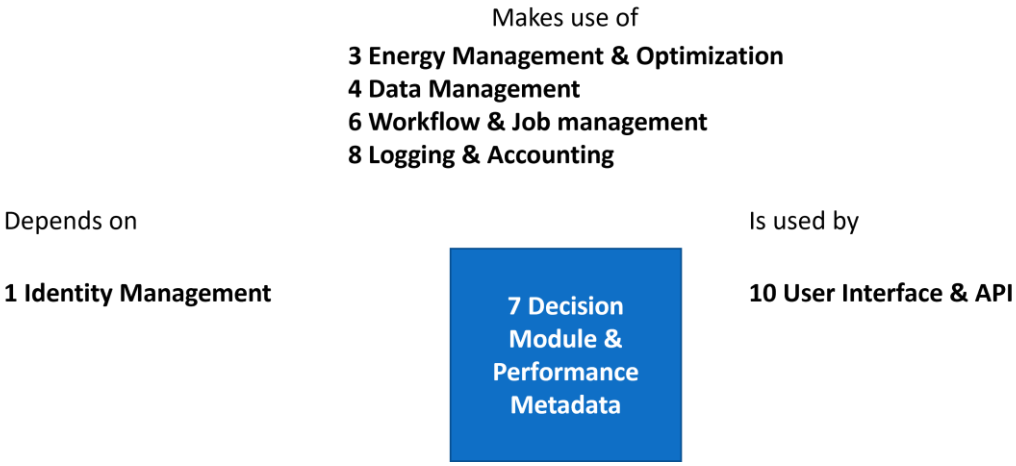

Figure 28 - Decision Module & Performance Metadata Module internal interfaces & interactions

The decision module will interact with the other HEROES platform modules via API.

Energy Management and Optimization – EAR:

- to get metadata about jobs power consumption

Data Management:

- to get accounting logs and other data useful for making predictions
- to store predictions and users' choices

Workflow & Job management:

- to get workflow submission request
- to receive workflow placement suggestions

Logging & Accounting:

- Send events with details about all received requests / tasks
- Send events with status of all executed task

User Interface & API:

- to request cost estimation of the workflow on different platforms using Cost API
- to provide the performance dashboard

4.7.4 External dependencies

Third-party Software:

- Elasticsearch [19]: used by OKA framework to store logs and predictions provided by Predict-IT.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.7.5 Expected outcomes & behaviour

The decision module will operate in four successive steps:

1. Parametrization by each organization
   Within an organization, the managers define a list of platforms available for each workflow. They also either choose an optimization strategy (e.g., best speed, best price, lowest power consumption with configurable weights) or allow each user to select his own strategy.
2. Prediction of jobs resources and cost
   For each job, accounting logs and energy data are used to predict the "optimum" job resources (memory, execution time) but also its power consumption. The cost API will then give an estimated price for executing the submitted workflow on the available platforms using the predicted resources.
3. Selection of the best platform
   Based on the selected strategy and other constraints (data localization, available licenses, GPUs…) the decision module will then select the "best" platform to execute the submitted workflow. This platform will be proposed to the user as the "best" solution, but he will still have the possibility to choose a different platform.
4. Workflow execution monitoring
   During the workflow execution (only if the workflow is composed of multiple jobs) the decision module will check at each step if the selected platform is still the best platform (if by the end of the workflow, the selected platform is overloaded, it will be wise to execute the last jobs of the workflow on a different platform).

4.7.6 Uncertainties

At the time of writing this document uncertainties include, but are not limited to:

- The workflow placement algorithm: currently, this submodule is not implemented yet and we do not have any data to test for the performance of this algorithm. This will depend on the data quantity and quality we will manage to gather during the project.

4.8 Module 8 – Logging & Accounting

The Logging & Accounting module will provide standardised tools for the other HEROES module to log events, messages, outputs from their own processes, jobs, and workflows directly or indirectly.
The module will also provide a centralized log storage, a sorting / filtering / analysis tool, and a configurable dashboard to visualize information extracted from the logged data.

4.8.1 Assumptions

At the time of writing this document, the following assumptions are made:

- a common log format will be established for all HEROES modules and related software components

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- logs will be pushed from modules to a central repository by making use of a dedicated agent
- access to logged data will be managed through role-based authorization
- dashboards for log and statistic visualization will have very limited editing capabilities (e.g., change only the timeframe of the displayed data)

4.8.2 Overview

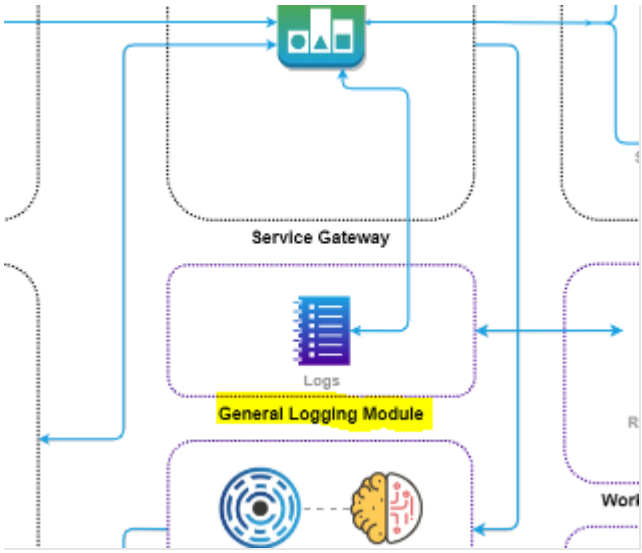

Figure 29 - Logging & Accounting Module in the HEROES architecture

The Logging & Accounting module architecture is composed of the following components:

- Distributed Log Collector, based on FluentD [18], that will oversee collecting all service / process / workflow / etc. events and messages on a single HEROES server and sending them to the central repository
- Log Storage & Analysis, based on Elasticsearch, that will be used to store, categorize, parse, and tag the collected log files
- Log Access & Presentation, based on Kibana [20] dashboards, that will be used to extract meaningful representations and information from the previously parsed log data

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The internal architecture of the Logging & Accounting module:

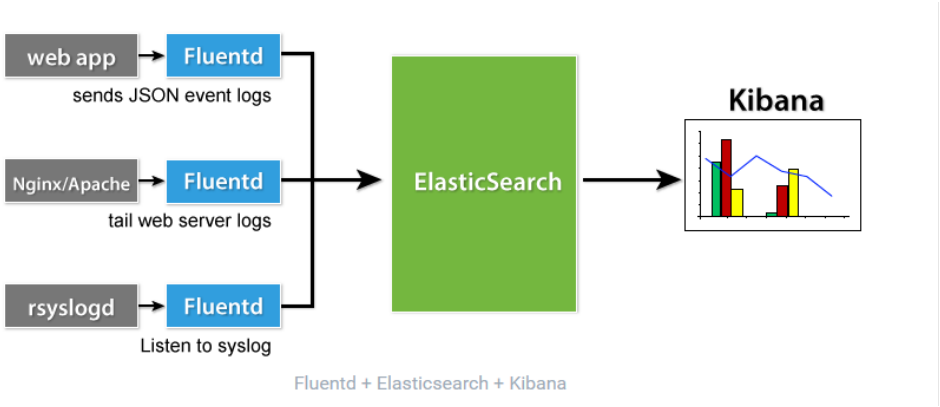

Figure 30 - Logging & Accounting Module internal architecture

All logs from web apps shall follow a common log format, following the Elastic Common Schema (ECS) guidelines [30] [31] and adding relevant fields such as:

- HEROES User
- HEROES Project
- HEROES Organization
- JobStep unique id
- Job unique id
- Workflow unique id
- etc.

4.8.3 Internal interfaces & interactions

Figure 31 - Data Management Module internal interfaces & interactions

Identity Management:

- User identification (check User's id, role, group / project / organization)
- Authorization (check if the user has access to data transfer source and targets)

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- Endpoint authentication (check if the API call was initiated from a legitimate endpoint within the HEROES platform)

User Interface and API:

- Pre-defined dashboards will be published through the HEROES UI by making use of the standard Kibana APIs, avoiding direct access to Kibana UI.

All HEROES Modules:

- All modules will log their events / messages in the standardized format (see above)
- All modules will have a specific configuration file (or stanza) for the FluentD daemon that will run on the server and will send the collected logs to the central log repository

4.8.4 External dependencies

Third-party Software:

- FluentD: software used to collect log data from different sources and send them to a central repository
- ElasticSearch: used to centrally store log data and to search, tag and analyse them.
- Kibana: used to publish data to users through the HEROES UI via API access.

Note: It is planned to use the Open Distro for Elasticsearch and Kibana to avoid issues with the SSPL/ELv2 license [21] that the original Elasticsearch introduced in the beginning of 2021.

4.8.5 Expected outcomes & behaviour

4.8.5.1 Log events / messages and store in central repository

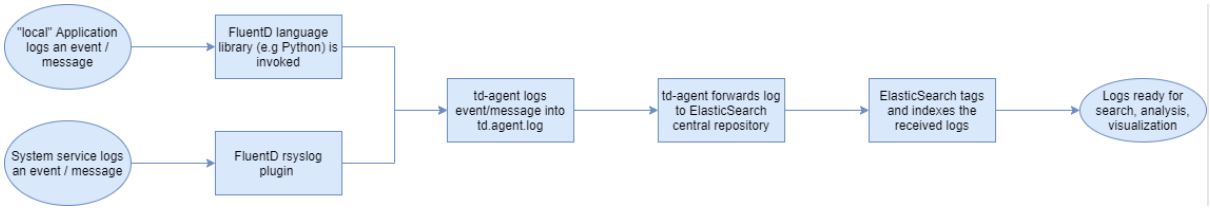

Figure 32 - Logging & storing workflow

Thanks to the many source plugins available for FluentD, a component of the HEROES platform will be able to log using native libraries (python, java, etc.) interacting directly with the FluentD td-agent (TreasureData Agent) [11] process. In case of system services, the rsyslog [34] source plugin will be used to forward any event / message that will normally be sent just to the OS (Operating System) logging facilities.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The td-agent process will then forward the logs to the central repository, where Elasticsearch will be able to tag, analyse and index the received data, that will then be ready for consultation, search, and further analysis with Kibana.

4.8.5.2 HEROES User access to logged information

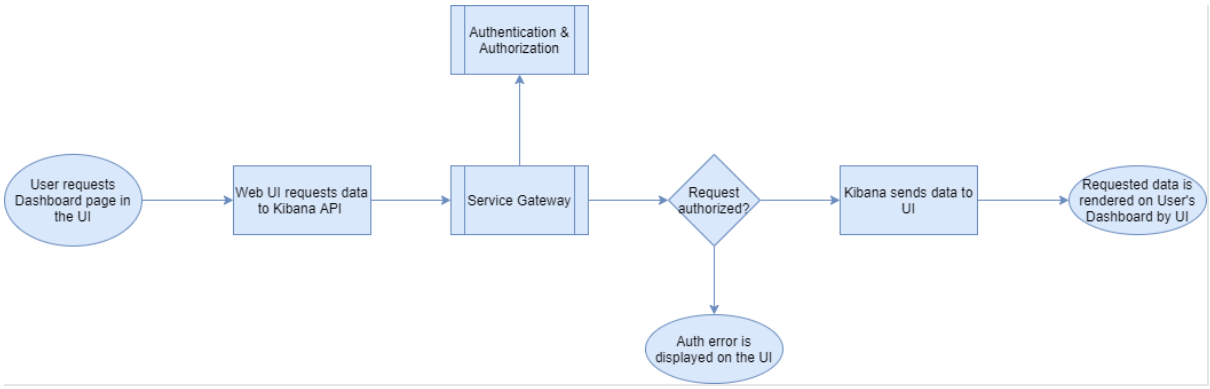

Figure 33 - User access to logged information workflow

HEROES User will be able to access only pre-digested information through the HEROES web UI. When a User requests a "Dashboard" page from the HEROES UI, the UI will send pre-configured requests to the Kibana API / interface through the Service Gateway. The Service Gateway will check the request to confirm / deny access to the requesting User. If access is granted, Kibana will send back data & graphs to the HEROES UI that will incorporate them in the User's "Dashboard".

Only HEROES administrators will be able to access the raw logs from the Kibana direct interface. Kibana shall be configured with SSO with the HEROES platform.

4.8.6 Uncertainties

At the time of writing this document uncertainties include, but are not limited to:

- Scale / capacity of the data logging & transfer of the whole HEROES platform
- If the HPC Centres will allow for td-agent to be installed somewhere within their premise; if not, FluentD will be configured in a "Pull" architecture instead of a "Push" one.

4.9 Module 9 – Cost Management

The Cost Management module will provide the following features:

- Resources provider business model:
  - will allow an HPC Centre to declare and manage the resources they want to publish within the HEROES platform and at what price

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

    - will fetch (through APIs) the current prices associated with CSP services to compute costs of using cloud resources
- Application / Workflow provider business model: will allow an application / workflow provider to declare and manage the costs and the licenses associated with their use.
- Budget management: each organization will be able to define budgets. The module will keep track of the actual consumption and make sure budgets are not exceeded.
- Billing & Credits:
    - The actual billing will not be made by the HEROES platform, but the interface will allow to see consolidated costs consumption per organization and projects.
    - Payment policies have not been defined yet, but two main options exist: pay per use or payment in advance / credits. Payment will not be directly handled by the HEROES platform, but if credits / prepayment exists, then they will be considered when tracking costs and managing budgets.

The module will provide multiple business models that will be defined by "WP5 – Dissemination / Business Model" document, such as "Ax+B": Fixed price (B) whenever a resource is used, plus a fee that depends upon the consumption (e.g., number of core-hours, number of nodes, storage…). The module will adapt to comply with the requirements of WP5.

4.9.1 Assumptions

At the time of writing this document, the following assumptions are made:

- WP5 will provide the considered business models and payment policies
- Actual billing and payment will be handled outside of the HEROES platform: no payment means, and automatic billing tool will be integrated.
- Only end-users with sufficient privileges will be able to access the Costs interfaces

4.9.2 Overview

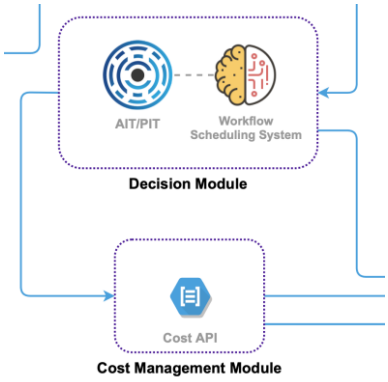

Figure 34 - Cost Management Module

The Costs Management module is composed of the following components:

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- A database that will store the information about Resources and Applications / Workflows business models, budgets, consumptions…
- APIs to register / update / delete new Resources and Applications / Workflows
- A budget management tool: responsible to control and enforce budgets vs. actual consumptions, and to reserve part of the budget whenever a workflow is executed.

4.9.3 Internal interfaces & interactions

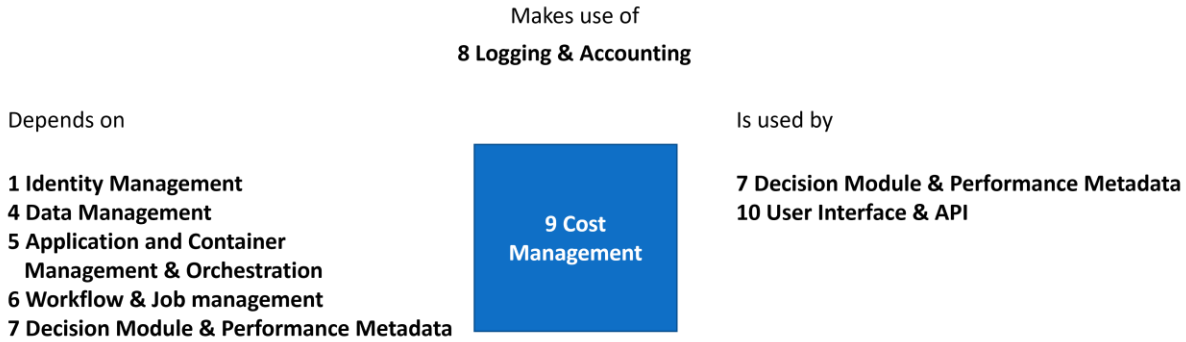

Figure 35 - Cost Management Module internal interfaces & interactions

Identity Management:

- User identification (check User's id, role, group / project / organization)
- Authorization: check if the user has access to Costs management, and which part (costs tracking, business model update…)
- Endpoint authentication (check if the API call was initiated from a legitimate endpoint within the HEROES platform)

User Interface and API

- Dashboards for costs tracking
- Budget management per organization and project
- Business model associated Resource and Applications, for providers

Decision module

- To predict the cost of running a workflow
- To access metadata about all the jobs to consolidate associated costs

Workflow module

- To reserve part of the budget once a user has submitted a workflow
- To track costs associated with the execution of the workflow

Data management module

- To track costs associated with data storage and transfers

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Logging & Accounting

- Send events about business models CRUD
- Send events about budget CRUD and monitoring

4.9.4 External dependencies

Third party software & services

- CSP Pricing APIs to retrieve the cost of using the services, accessed either through APIs or CLI (e.g., AWS Pricing APIs [32], Azure Retail Prices API [33], OCI pricing API [40])
- A relational database such as PostgreSQL to store budgets, business models and billing information…

4.9.5 Expected outcomes & behaviour

The Costs Module main objective is to:

- Provide means to manage business models for the applications / workflows and resources. The history of each business model will be kept for auditing.
- Provide a centralized accounting information for billing purposes (keep track of all consumed resources and associated prices).
- Provide means to declare and manage budgets within organizations

4.9.6 Uncertainties

At the time of writing this deliverable, the business models have not yet been defined. The solution will need to be flexible in this sense.

4.10 Module 10 – User Interface & API

The User Interface & API module purpose is to allow external access for end-users to the HEROES platform through easy-to-use UI and APIs.

It aims to give access to all HEROES capabilities as simply as possible so that we can provide a list of actions - as define at Section 3.3 - and therefore a way for users to interacts with the platform internal services.

This module is based on three elements:

- User Interface (UI): Front End to the platform
- Application Programming Interface: Back End to interact with internal modules
- Service Gateway: Single access point to published external APIs

4.10.1 Assumptions

At the time of writing this document, the following assumptions are made:

- Users will be able to directly access the UI with a web browser through https

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

- The access point for external APIs will be centralized using the Service Gateway
- Each internal module will provide its own APIs accessible only by the Service Gateway or by another module within the platform
- The control of a user authorization to access the platform's APIs will be handled by the Identity Management module
- The Service Gateway and / or external APIs will in some cases provide direct access to internal third-party tools' APIs to handle specific tasks related to internal module such as Data Management (e.g., data transfer via MinIO, RClone) or Identity Management (e.g., user management via Keycloak)

4.10.2 Overview

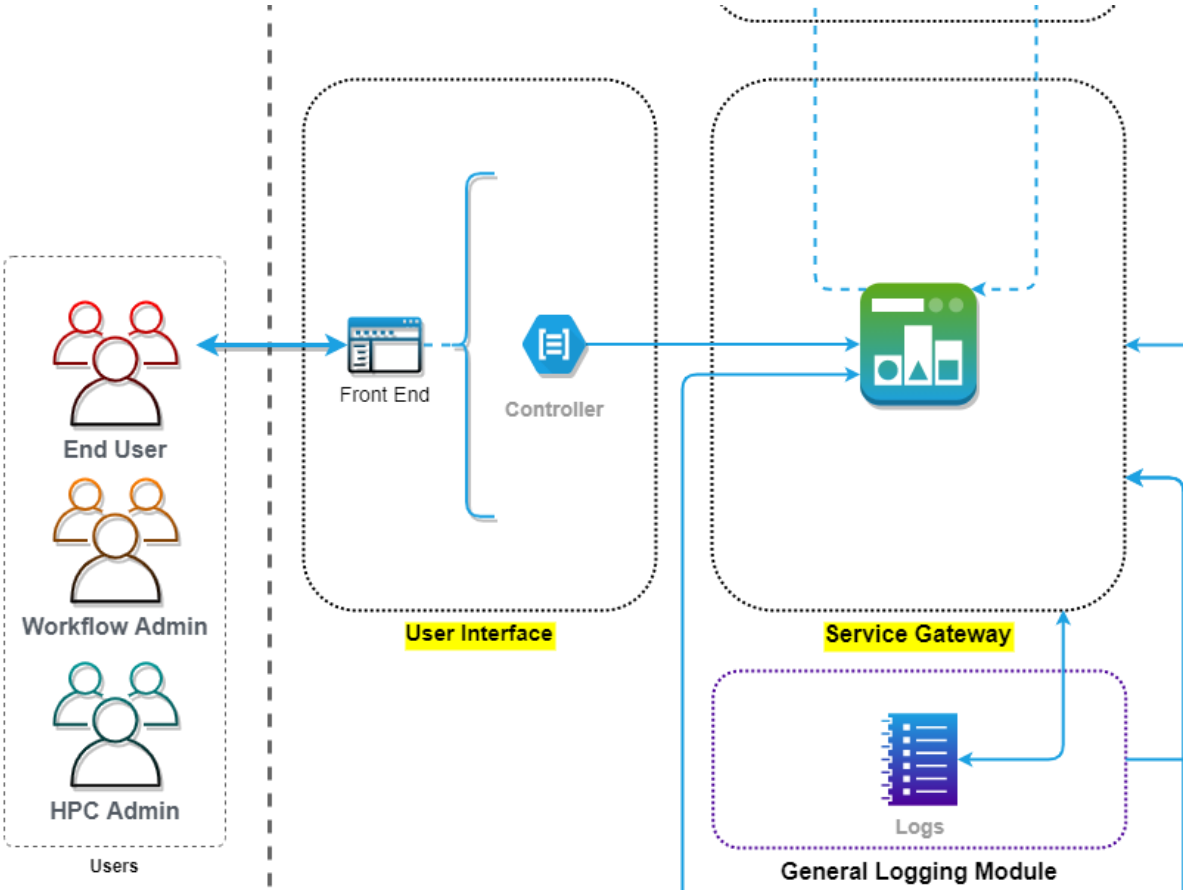

Figure 36 - User Interface and Service Gateway architecture

As shown in Figure 36 the User Interface & API module can be described as having two main components, the User Interface (UI) and the Service Gateway.

The UI was thought as the Front End of the platform. This part will provide access to multiple dashboards accessible to user or provider accounts based on their roles and privileges.
Those dashboards will allow users to have access to all data related to the - as defined in section 3.3.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

Available dashboards for each type of account include but are not limited to:

- Organization:
  - Global dashboard (e.g., handle users, groups, and their attributes)
  - Role dashboard (e.g., handle roles CRUD and attribution)
  - Data Management dashboard (e.g., access to organization data and own home)
  - Workflow Management dashboard (e.g., handle authorization over workflows and jobs)
  - Billing dashboard
- Providers:
  - Resource Provider:
    - Resources dashboard (e.g., list, add, publish resources and their pricing info)
  - Application Provider:
    - Applications dashboard (e.g., list, add, publish applications and their pricing info)
- Overall HEROES management
  - Management dashboard (e.g., access to monitoring, support info)

The Service Gateway on the other hand was designed to be the central and unique access point to the HEROES platform.
This component will handle all the external APIs that will be published and available to all users or third-party tools. All actions - as defined in section 3.3 - will be related to those high-level APIs.

4.10.3 Internal interfaces & interactions

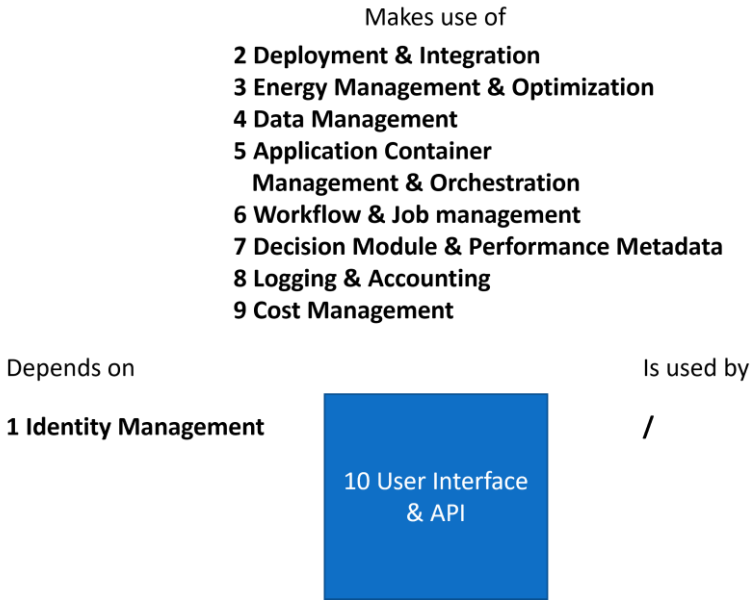

Figure 37 - User Interfaces and API Module internal interfaces & interactions

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

This module was thought as one that will be allowed to interact with all other modules as its purpose is to provide access to all functionalities that the platform will offer. At this stage, we cannot provide an exhaustive list of actions for this module however, the following list shows how this module will use some of the services at its disposal.

Common to all modules:

- Retrieve published data and metadata (e.g., to populate UI dashboards)
- Access internal services and execute internal functionalities through modules' APIs
- Configure and manage modules' content (e.g., CRUD, rights attribution)

Identity Management:

- User identification (e.g., check User's id, role, group / project / organization)
- Authorization (e.g., check if the user has access to data transfer source and targets)
- Endpoint authentication (e.g., check if the API call was initiated from a legitimate endpoint within the HEROES platform)

Data Management:

- Data Transfer to / from the platform to / from the end-user

Logging & Accounting:

- Send events with details about all received requests / tasks
- Send events with status of all executed task

4.10.4 External dependencies

Third-party Software:

- FastAPI [22]: Modern web framework for building APIs with Python
- NGINX [29]: Reverse proxy

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

4.10.5 Expected outcomes & behaviour

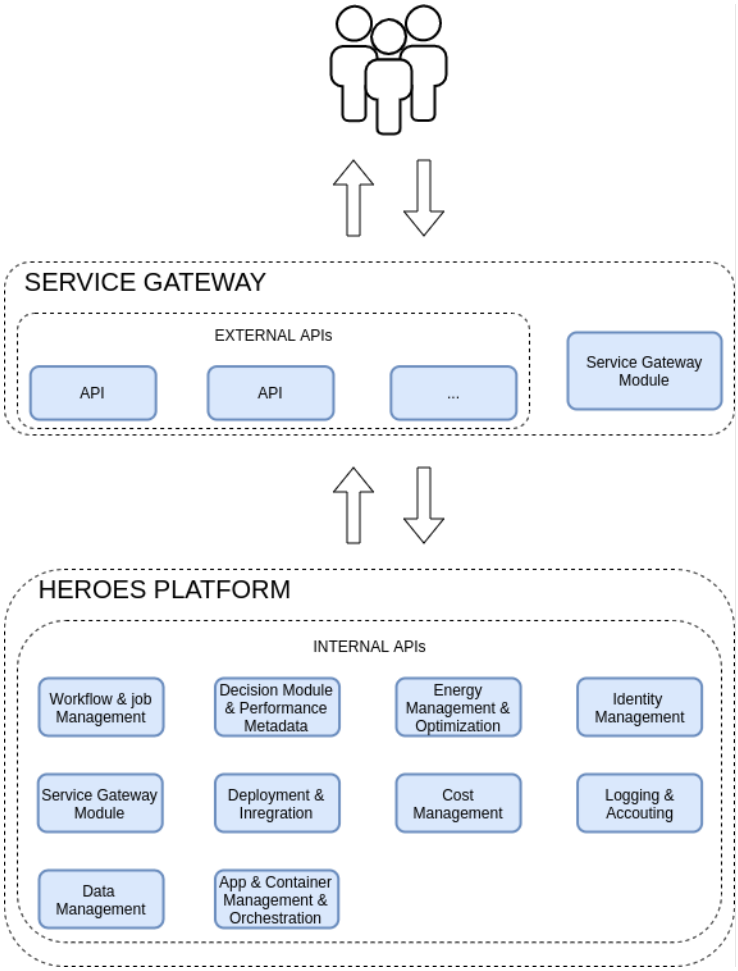

Figure 38 - API behaviour within HEROES platform

As presented in the previous sections, the aim of this module is to give access to internal HEROES's functionalities through simple external APIs.
To control the access to those APIs, we decided that they would be part of a Service Gateway component whose main purpose will be to handle routing of requests sent to the HEROES platform.

Based on those facts, the expected behaviour for this module is as follow.
Any call coming through the Service Gateway made by any user or third-party tool - either using the APIs directly or through the UI - will be subject to an authorization check made by the Service Gateway, using the Identity Management module (see Figure 7 in section 4.1 for detailed workflow example).

Then, assuming the user has the appropriate role and privileges, the requested API will proceed and itself send requests to use one or more of HEROES's internal services (see Figure 7 in section 4.1 for detailed workflow example).

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

The expected outcome will then depend on the actual original requested action. At this stage, we cannot define all the possible outcomes as the complete list of available actions has not been defined. However, all calls will at least result in a response being sent back with status and information regarding the execution of the requested action.

4.10.6 Uncertainties

Uncertainties include but are not limited to:

- How to handle data upload to the HEROES platform through the UI using RClone / MinIO APIs
    - Endpoint user transfer to HEROES storage though the Service Gateway
- What technology to use to develop the Front End
    - Node.js [24], ReactJS [25], Angular [26], Django [27], Vue.js [28]
- How to handle internal communication between modules:
    - RabbitMQ [23]: Lightweight and easy to deploy message broker
- The number, type, and content of each dashboard to be created for each HEROES's module
- The complete list of functionalities provided through internal APIs that will be available to the Service Gateway
- The complete list of actions to implement APIs for and publish using the Service Gateway

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.

5 Conclusion

In this document we presented what the architecture of the innovative software solution that the HEROES project aims to deliver should be. With the objective of creating a solution that would address both the needs of industrial and scientific users of the HPC community, we designed a modular system and described - with as much details as possible at this point - each of those modules to show, how they would work and how they would allow us to provide the functionalities that HEROES intends to offer.

The architecture design covered in this document demonstrates how we plan to build a fully encapsulated platform that would give end-users access to AI and HPC workflows as a Service. It shows how through a “single entry point” we aim to allow Applications and Resources Providers to publish their content on a marketplace environment on their own terms, conditions, and price. In turns, having those resources at their disposal, administrators in charge of scientific or industrial Organizations will be able to provide new types of workflows for their engineers and researchers therefore, creating high added values across multiple fields.

Diving deeper into the innovative features offered by the solution, we demonstrated how the platform would supervise the execution of any jobs or workflows over multiple HPC infrastructure: from HPC Centres to Clusters in the Cloud managed by multiple providers. Using one of the other innovations included in the platform, we illustrated how HEROES would help its users to take informed decisions while submitting those jobs and workflows so that they would make use of the best possible resources based on their type (e.g., ML, AI) and the priority set by the Organizations’ administrators (i.e., time-to-result, budget limitations and energy efficiency).

Throughout this design phase we looked at the available software and libraries that could be used as part of our different modules. While creating an innovative solution, HEROES does not intend to reinvent the wheel when it comes to basic components. Already proven technologies we thought could be a good fit for some of our requirements were put to the test to see if the features they offer could be useful to our project (e.g., Keycloak as part of the IAM module or RClone and MinIO to produce a Data Proxy Server). Those multiple tests helped us confirmed some of our assumptions but, also raised a few uncertainties in some cases that could not be cleared as it would have required further implementations that were not possible at this stage (e.g., the technical interaction between each API within the HEROES platform and the Identity Management module).

In its next step, the project will move-on to the implementation of a prototype to challenge the architecture that was designed during this first phase. We will use this development step to mock-up all core modules of the platform and attempt to answer as much of the remaining uncertainties as possible while doing so. The deliverable to come should allow us to demonstrate the HEROES platform functionalities giving us the possibility to handle workflows and job submission on conventional HPC infrastructures as well as CSP.

The HEROES project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 956874.