



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

D3.5 – Cost Service

Work Package	WP 3, Infrastructure as a Service
Lead Author (Org)	Benjamin Depardon (UCit)
Contributing Author(s) (Org)	Philippe Bricard, Brian Amedro (UCit)
Reviewed by	Elisabeth Ortega (HPCNow!), Davide Pastorino (Do IT Systems)
Approved by	Name (organization)
Due Date	30.08.2022
Date	21.10.2022
Version	V1.2

Dissemination Level

- | | |
|-------------------------------------|--|
| <input checked="" type="checkbox"/> | PU: Public |
| <input type="checkbox"/> | PP: Restricted to other programme participants (including the Commission) |
| <input type="checkbox"/> | RE: Restricted to a group specified by the consortium (including the Commission) |
| <input type="checkbox"/> | 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	06.09.2022	Benjamin Depardon (UCit)	TOC
0.2	15.09.2022	Benjamin Depardon (UCit) Brian Amedro (UCit) Philippe Bricard (UCit)	Introduction, definitions and models
0.3	20.09.2022	Philippe Bricard (UCit)	Platform modelling
0.4	21.09.2022	Benjamin Depardon (UCit)	Performance estimations, conclusion
0.5	23.09.2022	Benjamin Depardon (UCit)	Proofreading
0.6	26.09.2022	Philippe Bricard (UCit)	Integration & proofreading
0.7	27.09.2022	Benjamin Depardon (UCit)	Prepare for review
0.8	28.09.2022	Elisabeth Ortega (HPCNow!)	First revision
0.9	30.09.2022	Benjamin Depardon (UCit)	Taking into account comments
1.0	13.10.2022	Davide Pastorino (Do IT Systems)	Review
1.1	13.10.2022	Benjamin Depardon (UCit)	Final version
1.2	21.10.2022	Corentin Lefevre (Neovia)	Version approved by the Management Board

Disclaimer

This document contains information which is proprietary to the HEROES Consortium. Neither this document nor the information contained herein shall be used, duplicated or communicated by any means to a third party, in whole or parts, except with the prior consent of the HEROES Consortium.



Table of Contents

Executive Summary	5
1 Introduction	6
2 Definition of a HEROES Marketplace	7
2.1 Overview	7
2.2 HEROES Marketplaces use cases	8
3 Pricings and costs estimations	10
3.1 Pricing models	10
3.2 HPC platform modelling.....	11
3.2.1 Compute.....	12
3.2.2 Storage	13
3.2.3 Visualization	13
3.2.4 Data Transfers	14
3.2.5 Service	14
3.3 Performance estimations	15
3.4 Cost estimations	16
4 Conclusion.....	20

List of Figures

FIGURE 1. "PHYSICAL" MARKETPLACE	6
FIGURE 2. HEROES MARKETPLACE	8
FIGURE 3. HEROES MARKETPLACE FOR A LARGE CLIENT	9
FIGURE 4. HEROES MARKETPLACE FOR A PUBLIC HPC CENTRES	9
FIGURE 5. INSTANCE JSON REPRESENTATION	13
FIGURE 6. STORAGE JSON REPRESENTATION.....	13
FIGURE 7. DATA TRANSFERS JSON REPRESENTATION	14
FIGURE 8. SUPPORT JSON REPRESENTATION.....	15
FIGURE 9. INSTANCE BENCHMARK RATIOS JSON REPRESENTATION	16
FIGURE 10. OKA FILTERING CAPABILITIES TO DEFINE WORKLOADS.....	16
FIGURE 11. CLOUDSHAPER COST ESTIMATION	18
FIGURE 12. CLOUDSHAPER DETAILED COST ESTIMATION FOR COMPUTE	18
FIGURE 13. DETAILS ON COSTS AND ENERGY CONSUMPTION FOR A SELECTED WORKLOAD	19



TERMINOLOGY

Terminology/Acronym	Description
BYOL	Bring Your Own License
CRP	Cloud Resource Provider
CSH	CloudSHaper
CSP	Cloud Service Provider
DoA	Description of Action
EC	European Commission
GA	Grant Agreement to the project
HPC	High-Performance Computing
HPL	High Performance Linpack
KPI	Key Performance Indicator
MCDA	Multiple-Criteria Decision Algorithm
VO	Virtual Organization

9



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 contains the descriptions and the principles of the “Cost Service” which aims at defining interfaces and tools to allow computing clusters to publish their costs through the HEROES platform.

We present both the initial pricing models that are considered and a set of implemented methods & tools that allows to provide estimates and track costs when using resources through the HEROES platform. The solution allows dynamic queries to gather up-to-date information used to calculate the cost of using the resources of HPC Centres or Cloud Resource Providers (CSP), allowing for both static and dynamic pricing. The Cost Services also features a “spot” model (e.g., “best effort preemptible queues” on unused resources, with a lower price) to allow dynamic updates of resources cost.



1 Introduction

Central to the HEROES platform is the notion of Marketplace for computing resources used for HPC and AI workloads. In the introduction part of “The Anatomy of the Grid”, Ian Foster, Carl Kesselman and Steve Tuecke defined the Grid concept as “...coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations...” and then “...This sharing is, necessarily, highly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs. A set of individuals and/or institutions defined by such sharing rules form what we call a virtual organization (VO)”.

In this document, we draw an analogy between a VO implemented through the deployment of a HEROES platform and physical Markets (see Figure 1) through similar guiding principles:

1. Markets are organised: a set of rules is defined by a “**Mayor**” who decides in which place the market will happen and when it will take place. On top of this, several additional rules are set for exhibitors (“**Vendors**”) to be authorized to show their products such as access fees, booth location, signage, pricing transparency... Other rules are set for visitors (“**Clients**”) such as access fees, dress codes.... Eventually additional and more complex rules are set at the transactional level between “Vendors” and “Clients” to bring additional clarity in the exchange such as setting commissions on transactions.
2. To participate Vendors must agree to the terms as they register to exhibit in the marketplace, but they typically are free to set pricing and specific terms for their goods or services.
3. Clients come to the marketplace and select products according to their wants and needs if the Vendor terms suit them.

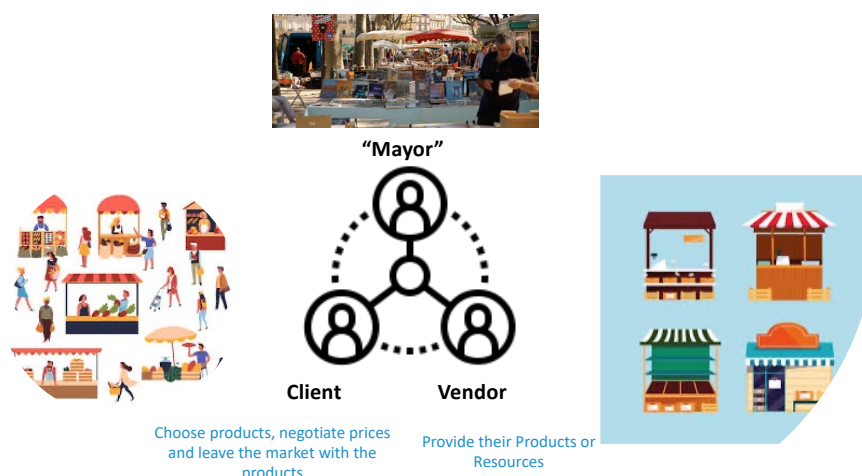


Figure 1. "Physical" Marketplace

The next section presents how this notion of Marketplace applies with similar notions in the context of the HEROES platform.

2 Definition of a HEROES Marketplace

The notion of a “HEROES marketplace” is an extension through the creation of a Cost Service of the Market principle. The HEROES platform is meant to be used in different contexts for which the notion of marketplace can apply, but with different meanings and implementations. In this section, we define what a HEROES marketplace could be, and how it applies to different use cases.

2.1 Overview

Following the analogy that was presented in Section 1, we can define a HEROES marketplace as follows (see Figure 2):

1. The “Mayor” is in our case the HEROES Platform “**Administrator**”. He is responsible for setting up the conditions that will allow Vendors to share their HPC resources and Clients to use them to execute their workflows. The HEROES platform acts as an intermediary between them and present the terms of the services so that they can be selected.
2. The “Vendors” are “**HPC resource providers**” who wish to “provide” them to third parties at their own terms and conditions (e.g., queue with GPU does not have the same availability and price than non-accelerated compute resources, preemptible resources could be sold at a “spot” price much cheaper than the “guaranteed”/on-demand resources). These conditions can vary over time.

Vendors can either be the users themselves (as their own clusters are usable through the marketplace), but they can also be HPC centres (public or private) willing to give access to “their” resources, or Cloud Service Providers. Though we concentrate in HEROES on HPC resources, the notion of Marketplace could be extended to propose/sell pre-defined/packaged workflows (along with software licenses).

3. The “Clients” are persons or entities (e.g., companies, laboratories, universities...) who need to access HPC resources to execute their workflows. They use the HEROES marketplace to identify which resources best fit their needs (e.g., 500 cores for CFD workloads) and constraints (e.g., I need my results before Friday and I have a budget of 10k€, and my Corporate social responsibility enforces me to use the most energy-efficient resources), and then execute their workflows. They can have special conditions negotiated with the Vendors which can be configured in the platform.

The HEROES Platform “Administrator”

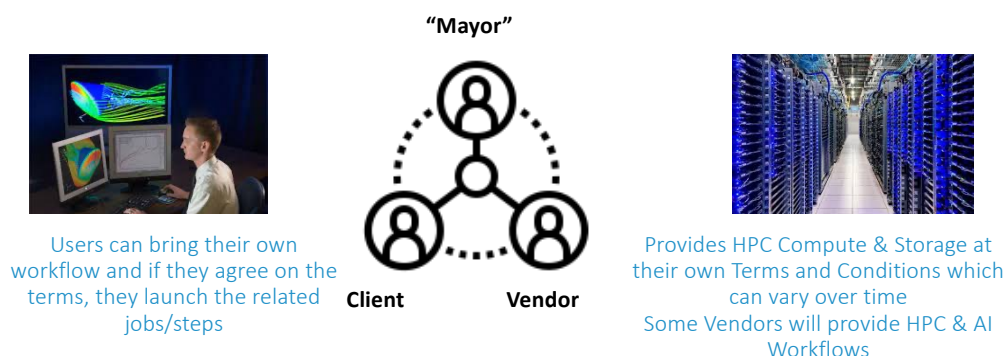


Figure 2. HEROES Marketplace

Note 1: the definitions given above are quite generic. We present in Section 2.2 selected use cases for which this notion of HEROES Marketplace can be applied.

2.2 HEROES Marketplaces use cases

We have identified two main use cases in which the HEROES marketplace could be instantiated.

Hybrid HPC deployment for Large Industrial/Private “Client”

Large clients often have compute resources distributed over several clusters (see Figure 3): either their own (on-premises) or provided as services (the use of Public Cloud to process non confidential data is currently a common scenario in the industry).

These resources are not necessarily interconnected nor shared between users. The deployment of a HEROES marketplace will help them to distribute the load of their HPC/AI workflows more transparently. In this scenario, payment for the use of the resources might not be always delegated to the end user or its department, however the information on resource cost could be at least made available to the end user business unit. The deployment of HEROES in this case would be single tenant (Client Organization).

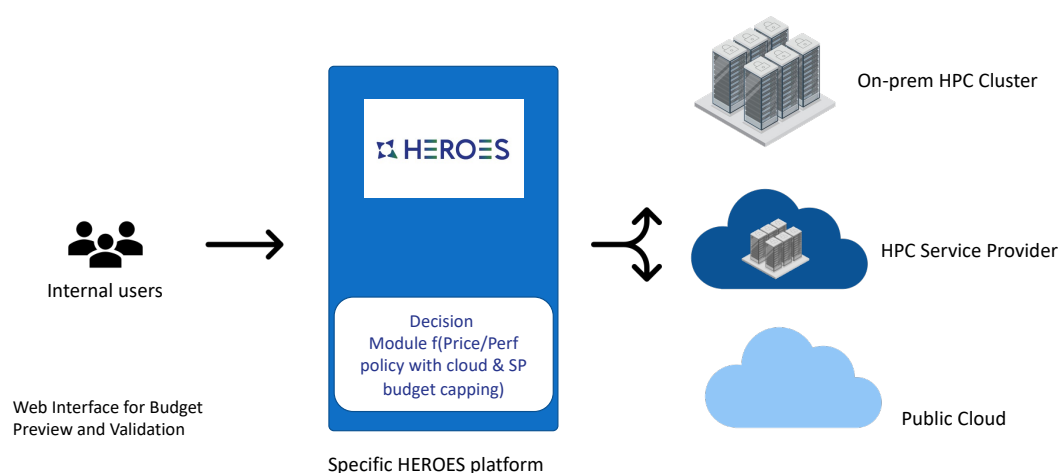


Figure 3. HEROES marketplace for a large client

Public HPC Centres

Public HPC Centres have the necessity to provide resources with external universities and with commercial enterprises within predefined context. Providing HPC capabilities to SMEs in order to accelerate innovation pace is one common scheme. A HEROES Marketplace could be deployed and managed by a central federating entity (e.g., EuroHPC) to provide a centralized access points to end-users from multiple organizations (see Figure 4) in a more dynamic way than the hour allocation process permits. Each HPC Centre would act as a Vendor in this context, and would publicize their costs, performances, and energy efficiency. An example of service offered to the end-users could be to help them select the most energy efficient HPC centre at the time of submission.

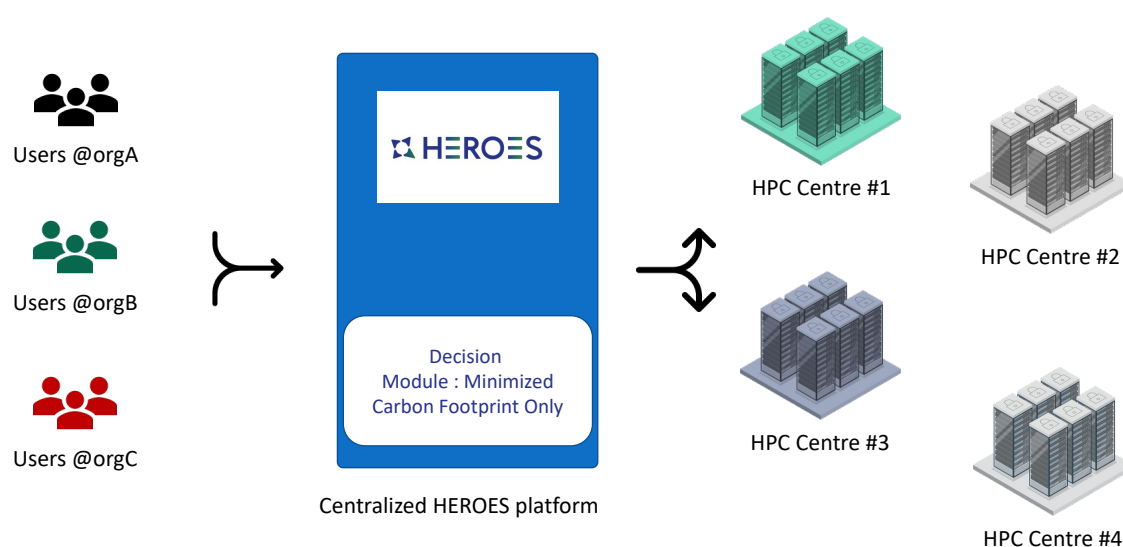


Figure 4. HEROES marketplace for a public HPC Centres

The HEROES platform doesn't integrate billing or payment mechanism which are tools typically used by Service Providers. Through the development of a specific "Data Enhancer" it can provide to a Service Provider billing system the accounting information. Its main function is to provide the HPC Resources enrolment with the definition of their costs and specific terms of use.

3 Pricings and costs estimations

3.1 Pricing models

Multiple pricing models can be found on the market. CSP and HPC centres have multiple ways of selling access to and utilisation of their resources, here are the most applicable to a HEROES context:

- **On Demand Pricing:** this is the standard model for many resources. The Buyer pays a certain price to get access to the resource for a "potentially" infinite duration (within the constraints set by the Vendor, e.g., queue limits). Except when a hardware failure occurs, the resource is available until the end of the job/task. When the Buyer has finished using the resource, it is freed and can be bought and used by another Buyer. Total cost is based on the actual consumption. The more popular example is the CPU*Hour cost from Cloud Providers or Compute Centres where additional services are embedded in the unit cost.
- **Preemptible Pricing:** a preemptible resource is a resource that usually cost less than the "on-demand" ones, but their availability is not guaranteed. They can be reclaimed by the Vendor for their own use or to sell them at another price (e.g., On-demand) to another Buyer. This model is sometimes referred to as "SPOT Pricing" or "Best-Effort Pricing". Additional terms need to be defined in the case of a pre-emption during execution. Cost is based on the actual consumption and is typically in use from Public Cloud providers. Due to the large number of compute cores typically required by HPC workloads, this model need to be considered carefully and limited to access to Datacentres with enough resources compared with job requirements.
- **Reserved Pricing:** in this model, the Buyer is ensured to have access to a specific type of resource for the duration of his engagement: the type of resource is always available to the Buyer, though the specific underlying hardware may change at each request to use it. Total cost is not based on the actual consumption, but on the engagement of the Buyer for a specific capacity during the contractual period. Pricing usually scales up as the engagement gets shorter, however Vendors tend to include some flexibility in the model with the capability to give some flexibility on the total amount of resources available at a given time. This model is commonly applied by HPC Centres for Private/Public partnerships and is a useful contributor to resource funding.

- **Dedicated Pricing:** This model is similar to Reserved, but as the hardware is dedicated to the Buyer, the flexibility is removed from the model. This model is the typical one used for on-premises clusters. The current Energy Crisis (both price increases and forthcoming shortages) is driving some of the traditional HPC market to efficient Datacentres located where energy is available and affordable.

In HEROES, we decided that the initial prototype would consider only the On-demand and Preemptible (also denoted afterwards as SPOT) pricing models. These are the most common in both the HPC centres and CSP, though they don't necessarily coexist, and some providers may only provide one of the two (i.e., usually On-demand). It is possible to model the fix cost of an on-premises cluster to fuel the decision module with a comparison point.

We present in the following sections the details of the models we designed to represent the costs of using a specific platform and the tools we have implemented to both estimate and track down these costs. The initial implementation has been done within OKA¹ (a data science platform for HPC developed by UCit), within a plugin called CloudSHaper and through specific tools called Data Enhancers.

3.2 HPC platform modelling

To estimate the running cost for a given workload, we need a model for its computational environment. This model permits to detail consumption on numerous aspects of a workload execution:

- Computing node allocation time
- Required storage on each tier
- Remote visualization strategy
- Data transfer volumes
- Cluster service
- Support plan

Each of these aspects, will correspond to a particular resource type, and a price for the cost estimation.

Thus, to describe a computational environment, we introduce the *HPC Platform Model* in CloudSHaper. Our *HPC Platform Model* relies on the following core components:

- Compute
- Storage
- Visualization
- Service
- Data transfers

These components are generic enough to encompass the need to describe both on-premises HPC Centres and clusters deployed in a CSP.

¹ <https://oka.how>



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

An instance of an *HPC Platform Model*, to describe an HPC cluster, is called an *HPC Cluster Configuration*. One can create multiple *HPC Cluster Configurations*, for each environment he has access to, such as:

- An on-premises cluster or an HPC Centre
- A cloud cluster hosted on AWS
- A cloud cluster hosted on Azure

An *HPC Cluster Configuration* can be accessible in multiple *Regions*, allowing to describe a model once and providing the capability to describe multiple clusters with multiple pricings. In a CSP, the region directly maps with their notion of Region and Availability Zones (i.e., a geographical location in which the CSP has datacentres, e.g., eu-west-1 for AWS), for an HPC Centre this notion can be used to describe either multiple offerings or to split the pricing per queues for example.

Each *HPC Cluster Configuration* will be used by CloudSHaper as a guideline to breakdown the costs of a given workload and provide a cost estimation to run it on the targeted environment.

To allow CloudSHaper to project resource consumptions of a workload on various *HPC Cluster Configurations*, our *HPC Platform Model* relies on a series of generic properties for each core component.

3.2.1 Compute

Compute allows to define a node allocation strategy for a workload, depending on an optimization criterion, such as execution time, cost or even reproducibility. These strategies rely on the definition of the workload to study (list of jobs with their characteristics in terms of node, CPU, RAM, execution time) to estimate the global cost of running it on a target platform. Current implemented strategies are:

1. **Lift and shift:** same the placement of the jobs on compute nodes with similar performances, trying to find the less expensive nodes.
2. **Optimized performance:** tries to find an allocation of the jobs on faster nodes, trying to lower the number of instances to reduce communications and to reduce the execution time of the jobs, even if this is more expensive.
3. **Optimized cost:** tries to place the jobs on cheaper nodes, trying less or more nodes than what was specified for the jobs (keeping the same number of CPU, and the same amount of RAM), the cost will be lower, but the execution time of the jobs might be longer.

This compute model relies on the notion of *Instance*. An *Instance* allows to define all the kind of computing nodes which are allocatable on the environment. These instances can be used for various purpose such as compute, head/service node or remote visualization.

Each instance has the following generic properties that are used by the algorithms when trying to match the needs of the jobs to the correct node type: *Name, family, tenancy (shared or dedicated to prepare for future pricing models such as Reserved, see Section 3.1), network*



bandwidth, operating system, vcpu, memory, gpu, locations and costs (per hour of usage). Figure 5 presents the JSON representation of instances in CloudShaper.

```
{
  "c5n.18xlarge": {
    "memory": 192.0,
    "tenancy": "Shared",
    "networkPerformance": "100 Gigabit",
    "operatingSystem": "Linux",
    "vcpu": 72,
    "gpu": 0,
    "region": {
      "eu-west-1": {
        "on_demand_price": 4.392,
        "price_unit": "Hrs",
        "spot_price": 1.2367
      }
    }
  },
  ...
}
```

Figure 5. Instance JSON representation

3.2.2 Storage

Storage allows to define a solution to store data. Depending on its usage, this solution must be one of those kinds of storages:

- Network File System (such as NFS, CIFS, Azure Files, etc.)
- Parallel File System (such as LUSTRE, BeeGFS, AWS FSx for Lustre, etc.)
- Object Store (such as AWS S3, MinIO, etc.)

Pricing model for the storage is based on the volume of data requested for a 1-month period (prorated to the actual workload duration).

```
{
  "eu-west-1": {
    "name": "EU (Ireland)",
    "on_demand_price": 0.023,
    "price_unit": "GB-Mo"
  },
  ...
}
```

Figure 6. Storage JSON representation

3.2.3 Visualization



Visualization allows to estimate resource consumption for remote visualization. Depending on screen resolution, refresh rate and achievable level of image compression, remote visualization will consume CPU time on the remote visualization host and network bandwidth.

Thus, three strategies can be applied:

1. **Light**, for typical 2D application with low refresh rate
2. **Medium**, for 2D and lightweight 3D applications
3. **High**, for intensive 3D application

The algorithms will try to find *Instances* that fits the remote visualization needs and will compute the associated costs of the *Instance* and the *Data transfers* associated with the streaming of the “remote screen” (based on assumptions about the size of screen and the average bandwidth need in each strategy for an h264 interactive video stream).

3.2.4 Data Transfers

Data Transfers allows to estimate the cost of sending data outside the platform. This is usually the case on CSP, not necessarily on HPC Centres. The model allows for either a flat pricing (per GB transferred) or a stepped pricing based on the amount of data transferred.

```
{
  "eu-west-1": {
    "name": "EU (Ireland)",
    "on_demand_price": [
      {
        "begin_range": "153600",
        "cost": "0.0500000000"
      },
      {
        "begin_range": "10240",
        "cost": "0.0850000000"
      },
      {
        "begin_range": "51200",
        "cost": "0.0700000000"
      },
      {
        "begin_range": "0",
        "cost": "0.0900000000"
      }
    ]
  },
  ...
}
```

Figure 7. Data Transfers JSON representation

3.2.5 Service

Service allows to include general resource requirement for the whole cluster. This includes:



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

- the head node for the job scheduler and resource management
- the network components
- the support offered to the Buyer

Note: Software licenses are not considered in this model. We consider that the Buyers will provide their own software licenses (Bring Your Own License – BYOL – model).

Within the service sphere, a specific part is dedicated to the Support offered by the Vendor. *Support* allows to define support level available for the whole cluster. This is especially required for cost estimation. Multiple *Support* pricing models are supported: either fixed price (see *min_cost* in Figure 8) or based on the global cost of using a platform for the selected workload (with steps to adapt to different ranges of costs, see *cost_steps* in Figure 8).

```
{
  "developer": {
    "cost_steps": [
      {
        "begin_range": "0",
        "percentage": "0.0300000000"
      }
    ],
    "min_cost": "29"
  },
  "business": {
    "cost_steps": [
      {
        "begin_range": "80000",
        "percentage": "0.0500000000"
      },
      {
        "begin_range": "10000",
        "percentage": "0.0700000000"
      },
      {
        "begin_range": "0",
        "percentage": "0.1000000000"
      }
    ],
    "min_cost": "100"
  }
},
```

Figure 8. Support JSON representation

3.3 Performance estimations

In order to provide estimations on the cost of running a workload we need to have performances estimations for each job of a workload, based on the strategy for the selection



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

of the target *Instances* (see Section 3.2.1). These benchmarks can be provided for each *Instance* type and combines two ratios (see Figure 9): *perf_ratio* considering the CPU performance, and *node_ratio* considering the network performance. These ratios were computed through the execution of a standard HPC benchmark: High Performance Linpack² (HPL). The ratio is the comparison of the result of each individual *Instance* with a selected “reference *Instance*”.

```
{
  "c5.xlarge": {
    "perf_ratio": 0.91181259,
    "node_ratio": 0.9
  },
  ...
}
```

Figure 9. Instance benchmark ratios JSON representation

The format of these benchmarks and the way the algorithms use them will change in the next version of the Decision module that will be presented in “D4.2 Decision module prototype integrated in HEROES”, to integrate multiple dimensions (CPU, RAM, Network, GPU...), and combine the result of these benchmarks through Multiple Criteria Decision Algorithms (MCDA).

3.4 Cost estimations

The main goal of CloudShaper is to provide costs estimations for running an HPC workload on a target HPC platform (on-premises or in the cloud). The estimations are based on the HPC platform modelling presented in Section 3.2 and some information about the workload that needs to be estimated. This first implementation leverages the capability of OKA to ingest accounting logs of job schedulers to get the information about the workload. Once the historical information about the jobs ingested, OKA allows the definition and selection of a workload through the use of advanced filtering capabilities (see Figure 10): this gives us the details on the types of jobs that need to be estimated (their duration, number of CPU and memory needs...).

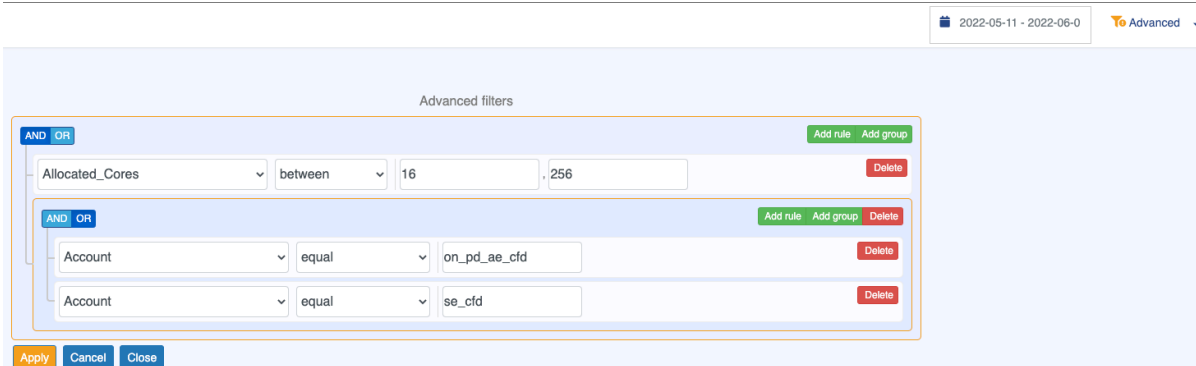


Figure 10. OKA filtering capabilities to define workloads

² <https://netlib.org/benchmark/hpl/>



As said in Section 3.1, the current prototype only supports On-Demand and Preemptible (SPOT) pricing models: the default model being On-demand, the SPOT model can be specified per cluster depending on its availability. The price list for all components presented in Section 3.2 can be automatically retrieved by CloudSHaper through the implementation of specific plugin, or through the provision of JSON files. Currently two plugins have been implemented for 2 CSP: AWS and Azure and rely on their pricing APIs which require to have a valid account with the relevant access rights to their platform. The addition of an on-premises HPC cluster can easily be done by setting up the correct model configuration and either providing the JSON files on a periodic basis or by implementing a specific plugin if the cluster has APIs or ways to automatically retrieve the price lists.

Once the configurations are setup, we can query CloudSHaper to have an estimation of the cost of a specific workload (set of jobs) on a target platform for multiple scenarios/strategies. CloudSHaper automatically tries to map the needs of the jobs to target *Instances* taking in to account the selected strategy (Lift & Shift, Optimized Performance, or Optimized Cost) – it can do so thanks to the information provided as input from the accounting logs of the job schedulers (but the same information could be provided through an API). The information about which type of storage(s) and their sizing and performance, or if the end-users need to access remote visualization capabilities are unknown to CloudSHaper and need to be provided through the interface. This allows to build different scenarios depending on the “size” of the target infrastructure that is required compared to the cost this platform will incur. The result of the estimation is provided with both a high-level view (see Figure 11) through the 5 categories presented in Section 3.2 (Service, Compute, Storage, Visualization and Data-Transfer), but also provides detailed information about what each category contains (see example for Compute in)



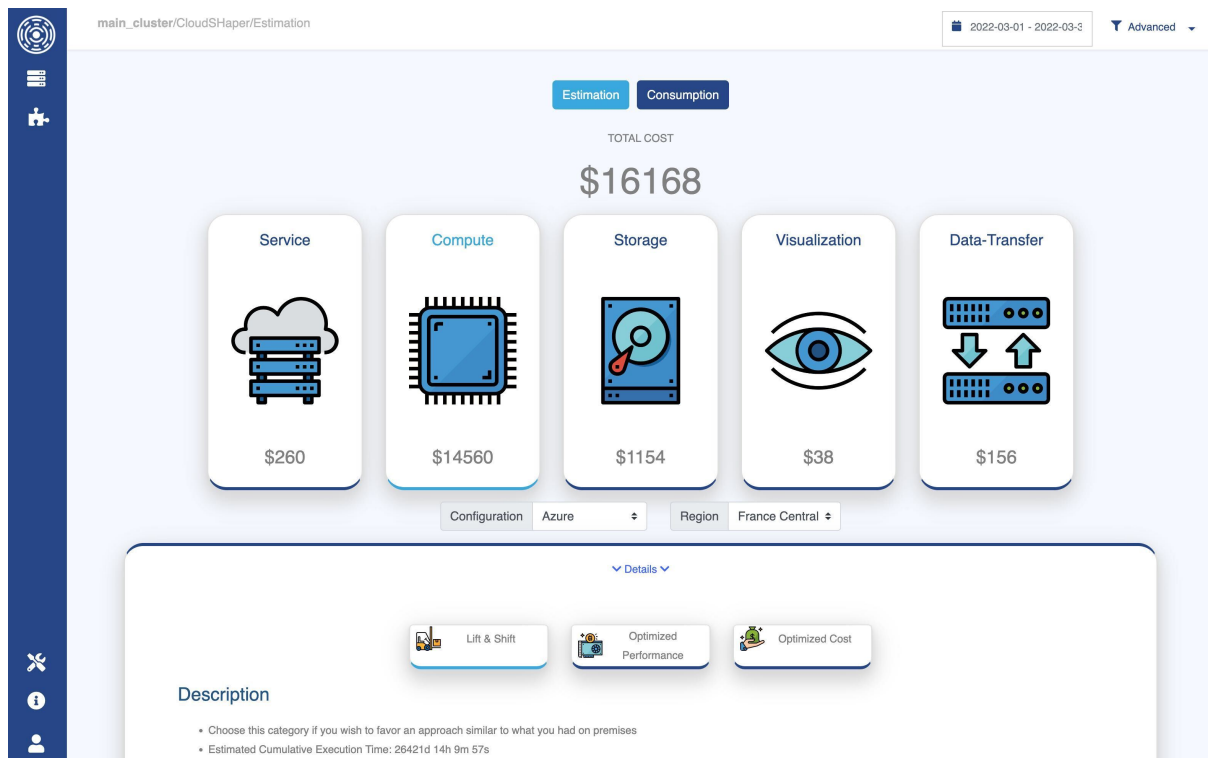


Figure 11. CloudSHaper cost estimation

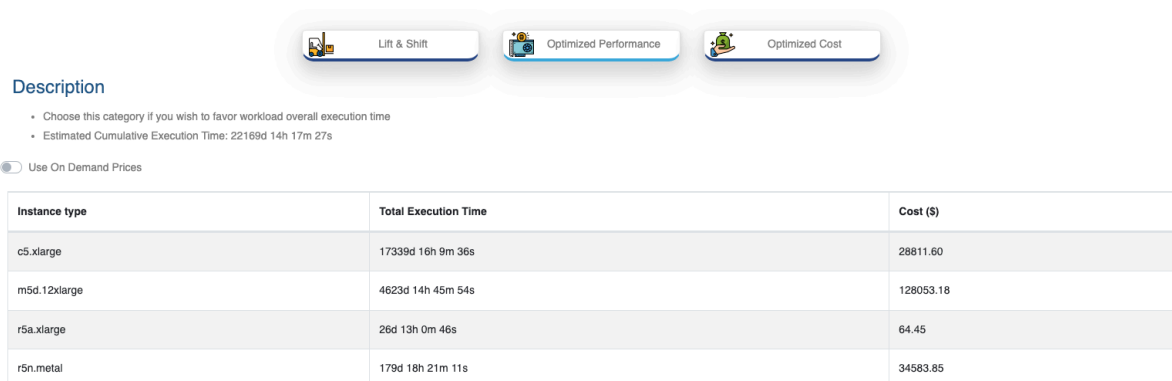


Figure 12. CloudSHaper detailed cost estimation for Compute

Costs estimation help the users or the Organization Administrators to select the platform that best fits their needs in terms of costs. This estimation can be either directly provided for example prior to creating a Cloud cluster (to support a whole set of jobs/workflows belonging to a defined workload), or through the decision module within the placement algorithms. Once a job has run, the actual cost needs to be retrieved and stored in OKA database for later invoicing and to serve as new input in the decision module. OKA provides a flexible way for

that through the use of Data Enhancers³: Python scripts that can be embedded in OKA to add information about jobs, such as the cost. For each HPC Platform enrolled in HEROES, a Data Enhancer can be created to compute/retrieve for each job the actual cost. Once computed, this cost is then directly accessible in OKA, just like the information about the consumed energy (coming from EAR, see deliverable D4.1 - Updated Energy Aware Runtime) – see Figure 13.



Figure 13. Details on Costs and Energy consumption for a selected workload

³ See https://doc.oka.how/admin_guide/configuration/data_enhancers.html



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

4 Conclusion

This deliverable presents the notion of Marketplace as defined within HEROES, and the associated Cost Service to define, estimate and track pricing and costs of using HPC platforms both on-premises and in the Cloud. The Cost Service prototype has been implemented in OKA through the CloudSHaper plugin. It allows to estimate the cost of a selected workload in different scenarios.

Next steps for this Cost Service include the capability to add detailed information about the cost of the energy consumption and the carbon footprint. These topics now become of utmost importance for many datacentres. Recent events have increased the pressure of both topics:

- The rise of energy costs has had a huge impact on the capability of HPC Centres to operate at full capacity. Prices being sometimes multiplied by a factor of 2 even forced some of them to cancel the renewal of clusters due to their incapacity to sustain the cost of operating the cluster.
- The increased dryness and spread of wildfires have demonstrated once more the urgency to consider the impact of our actions on the climate. Taking into account the carbon footprint of using HPC Platforms and helping end-users in using them more efficiently is mandatory. The Cost must not be seen only as the direct monetary cost, but also the global impact (cost) running these workflows has.

