Big Data technologies and extreme-scale analytics

Multimodal Extreme Scale Data Analytics for Smart Cities Environments

D5.8: HPC infrastructure and resource management for audio-visual data analytics – final version †

Abstract: This deliverable presents the High Performance Computing (HPC) infrastructure utilised by the MARVEL project to deploy the audio-visual data analytics framework. This document presents a comprehensive overview of the resources delivered by the Poznan Supercomputing and Networking Center (PSNC) to the consortium for the processing and storing of data. Computing resources for running applications come from domains like HPC and cloud, where virtual machines with framework services are set up according to pilot demands. The document also discusses the MARVEL Data Corpus, which aims to provide substantial volumes of anonymised and annotated audio-visual datasets for scientific and industrial communities. Moreover, the report addresses organisational and operational aspects, explaining the procedures for accessing and managing resources as well as obtaining technical support.

Contractual Date of Delivery	31/12/2023
Actual Date of Delivery	03/01/2024
Deliverable Security Class	Public
Editor	*Pawel Bratek (PSNC)*
Contributors	PSNC, FBK, GRN, ZELUS, INTRA
Quality Assurance	*Ilias Seitanidis (INTRA)* *Kostas Poulios (STS)*

† *The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957337.*

The *MARVEL* Consortium

Part. No.	Participant organisation name	Participant Short Name	Role	Country
1	FOUNDATION FOR RESEARCH AND TECHNOLOGY HELLAS	FORTH	Coordinator	EL
2	INFINEON TECHNOLOGIES AG	IFAG	Principal Contractor	DE
3	AARHUS UNIVERSITET	AU	Principal Contractor	DK
4	ATOS SPAIN SA	ATOS	Principal Contractor	ES
5	CONSIGLIO NAZIONALE DELLE RICERCHE	CNR	Principal Contractor	IT
6	INTRASOFT INTERNATIONAL S.A.	INTRA	Principal Contractor	LU
7	FONDAZIONE BRUNO KESSLER	FBK	Principal Contractor	IT
8	AUDEERING GMBH	AUD	Principal Contractor	DE
9	TAMPERE UNIVERSITY	TAU	Principal Contractor	FI
10	PRIVANOVA SAS	PN	Principal Contractor	FR
11	SPHYNX TECHNOLOGY SOLUTIONS AG	STS	Principal Contractor	CH
12	COMUNE DI TRENTO	MT	Principal Contractor	IT
13	UNIVERZITET U NOVOM SADU FAKULTET TEHNICKIH NAUKA	UNS	Principal Contractor	RS
14	INFORMATION TECHNOLOGY FOR MARKET LEADERSHIP	ITML	Principal Contractor	EL
15	GREENROADS LIMITED	GRN	Principal Contractor	MT
16	ZELUS IKE	ZELUS	Principal Contractor	EL
17	INSTYTUT CHEMII BIOORGANICZNEJ POLSKIEJ AKADEMII NAUK	PSNC	Principal Contractor	PL

Document Revisions & Quality Assurance

Internal Reviewers

1. *Ilias Seitanidis (INTRA)*
2. *Kostas Poulios (STS)*

Revisions

Version	Date	By	Overview
0.8	03/01/2024	PC	Final approval and submission
0.7	02/01/2024	Pawel Bratek	Addressed comments from STS (IR) and FORTH (PC)
0.6	02/01/2024	Manolis Michalodimitrakis, Despina Kopanaki	Comments on the first draft from FORTH (PC)
0.5	29/12/2023	Kostas Poulios	Comments on the first draft from STS (IR)
0.4	28/12/2023	Pawel Bratek	Addressed comments from INTRA (IR)
	27/12/2023	Ilias Seitanidis	Comments on the first draft from INTRA (IR)
0.3	21/12/2023	Pawel Bratek	First draft of the deliverable for Internal Reviewers
0.2	09/11/2023	Pawel Bratek	Revised ToC
	01/11/2023	Dusan Jakovetic	Comments on the ToC from UNS (STPM)
	24/10/2023	Ilias Seitanidis	Comments on the ToC from INTRA (WPL)
0.1	13/10/2023	Pawel Bratek	Initial ToC

Disclaimer

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
AT	Audio Tagging
AV	Audio-Visual
AVAD	Audio-Visual anomaly detection
AVCC	Audio Visual Crowd Counting
BDVA	Big Data Value Association
DFB	Data Fusion Bus
DMP	Data Management Platform
DMT	Decision-Making Toolkit
DNS	Domain Name System
E2F2C	Edge-to-Fog-to-Cloud
EC	European Commission
FDR	Fourteen Data Rate
GB	Gigabyte
GPFS	General Parallel File System
GPGPU	General-Purpose computing on Graphics Processing Units
HDD	Hierarchical Data Distribution
HDFS	Hadoop Distributed Files System
HPC	High Performance Computing
HTTPS	Hypertext Transfer Protocol Secure
ICT	Information and Communication Technologies
IoT	Internet of Things
IP	Internet Protocol
KPI	Key Performance Indicator
MAN	Metropolitan Area Networks
MDS	Meta Data Server
ML	Machine Learning
MPP	Massive Parallel Processing
MQTT	Message Queuing Telemetry Transport
MVP	Minimum Viable Product
PB	Petabyte
PFLOPS	Peta Floating-Point Operations Per Second

QDR	Quad Data Rate
RADOS	Reliable Autonomic Distributed Object Store
RAM	Random Access Memory
RDB	RADOS Block Device
REST	Representational State Transfer
SED	Sound Event Detection
SMP	Symmetric Multiprocessing
SSH	Secure Shell
TB	Terabyte
UI	User Interface
UPS	Uninterruptible Power Supply
ViAD	Visual Anomaly Detection
VCC	Visual Crowd Counting
VM	Virtual Machine
WP	Work Package

Executive Summary

This deliverable showcases the final version of the MARVEL HPC infrastructure and resource management for audio-visual data analytics, offering a comprehensive overview of hardware, software, and services provided by the Poznan Supercomputing and Networking Center (PSNC).

The HPC infrastructure requires specialised expertise to be used efficiently, and thus, the deliverable provides the knowledge of mechanisms for accessing and using these resources at PSNC. Furthermore, to ensure seamless access to compute resources within the MARVEL project and to carry out testing, integration, validation, and benchmarking procedures in an efficient manner, it is imperative to pay particular attention to the parameters of the offered infrastructure. The pilot and the pre-production testbeds are connected with specialised resources such as HPC and cloud, including GPGPUs, which enable taking full advantage of the AI-based algorithms for audio-visual analytics developed in WP3 (AI-based distributed algorithms for multimodal perception and situational awareness).

At the same time, PSNC, as a hosting centre and national network operator, has teams of experts in the areas of HPC, cloud, storage and security that fully support all aspects of computing and data infrastructures.

The delivered infrastructure comprises access to the EAGLE cluster and LabITaaS, functioning as a virtualised private cloud with an OpenStack web interface tool. This tool enables flexible allocation of computing resources and managing various class storage services connected with the HPC system.

The document also reports on infrastructure delivered for the Data Corpus, designed to provide a substantial volume of anonymised and annotated audio-visual datasets from MARVEL pilots to scientific and industrial communities. Based on the project agreements, PSNC will maintain the infrastructure for the Data Corpus for one year after the end of the project.

This document's structure is similar to the initial version, submitted in June 2022 as D5.3. Compared to D5.3, it includes relevant updates reflecting the current status at the end of the project. Additionally, the content has been expanded to include sections about achievements from the second part of the project. Specifically, the document describes implemented solutions for extending the Kubernetes cluster with GPU-equipped nodes at the cloud layer. Furthermore, it provides information related to HPC and cloud infrastructure monitoring and outlines contributions to developing a tool that facilitates monitoring the entire E2F2C framework's infrastructure.

1 Introduction

1.1 Purpose and scope of this document

The purpose of this document is to present the established infrastructure at HPC and cloud levels for the implementation of the MARVEL framework. The infrastructure consists of computing and memory resources that are linked to specific resources such as HPC and dedicated cloud virtual machines. By enabling access to GPGPU, the capabilities of AI-based algorithms for audio-visual analytics developed in WP3 are fully exploited.

In addition, the document discusses the aspects of planning and managing access to resources implemented in HPC, focusing on two access modes: batch and interactive. It also explains how to use the pre-installed software using the functionality of the module and install user-owned applications. Then it outlines the services offered by PSNC to users and provides guidance on contacting the support department in case of issues.

Cloud computing services are presented in terms of their specification (hardware configuration) as well as determined for the project demands resources. The document describes also implemented solutions for extending the Kubernetes cluster with GPU-equipped nodes at the cloud layer. Furthermore, it provides information on HPC and cloud infrastructure monitoring, along with contributions to developing a tool that facilitates monitoring the entire E2F2C framework's infrastructure.

Finally, it presents the infrastructure dedicated to the Data Corpus, designed to provide a substantial volume of anonymised and annotated audio-visual datasets from MARVEL pilots to scientific and industrial communities.

1.2 Intended readership

Deliverable D5.8 – "HPC infrastructure and resource management for audio-visual data analytics – final version" is a publicly accessible document showcasing infrastructure provided by PSNC for deploying the cloud layer of the MARVEL E2F2C framework.

The content found in this document aims to show to all stakeholders and potential users the HPC and the cloud infrastructure used for installing MARVEL services. Moreover, the resource management and rules are discussed along with their fulfilment which are required for the correct use of the offered resources.

Additionally, this document serves as a reference to any activities related to future cloud layer upgrades and technical documentation including among other, large-scale implementations, to be built upon the latest cloud advancements of the MARVEL.

1.3 Contribution to WP5 and project objectives

This deliverable has been composed within the context of 'WP5 – Infrastructure Management and Integration', and more specifically, it constitutes the final output of 'Task 5.1. HPC infrastructure' and 'Task 5.2. Resource management and optimised automatic usage of external computational and storage resources'. As part of the objectives of WP5, the document provides details of the provision and configuration of the HPC and cloud infrastructure for processing and storing data. These resources lay the foundation for high-throughput processing and storage, tailored to meet the specific needs of the MARVEL services. The configuration is designed to enable fast, complex analytics, integrating high-level services and ML modelling.

Throughout the project, PSNC has provided a range of services, encompassing the orchestration of infrastructure resource management as well as optimised automatic usage of external computational and storage resources. Services such as authentication, resource discovery, and task monitoring are offered as standard by the PSNC centre, supported by user services and administrative support. Together, these provide the environment necessary to carry out the software development process based on testing, integration, validation, and benchmarking. Furthermore, PSNC, as a hosting centre and national network operator, offers support from experts in the areas of HPC, cloud, storage and security that fully support all aspects of computing and data infrastructures.

The overarching goal of MARVEL is to deliver an Edge-to-Fog-to-Cloud (E2F2C) framework based on reliable infrastructure that operates in real-world environments, processing large volumes of captured AV data enabled by multi-modal perception and intelligence. More specifically, with regards to the project objectives analysed in the DoA, the delivery of the reliable infrastructure is related to "Objective 3", which states about distributed and secure E2F2C computing framework for processing Big Data and IoT applications as well as to "Objective 5" that concentrates on creating the MARVEL Data Corpus-as-a-service contributing to BDVA standards.

The HPC, cloud and storage infrastructure established in a manner ensuring high-level operability directly contributes to the achievement of the above-mentioned project objectives. This contribution consists of providing and configuring necessary resources for the MARVEL framework development process, including testing, integration, validation, and benchmarking up to deployment of the final solution. The delivered infrastructure also contributes to the fulfilment of the objective related to the MARVEL Data Corpus. Based on the project agreements, PSNC will maintain the infrastructure for the Data Corpus for one year after the end of the project.

1.4 Relation to other WPs and deliverables

An initial version of this document was submitted in June 2022 as 'D5.3 – HPC infrastructure and resource management for audio-visual data analytics – initial version' [6].

This deliverable builds upon foundational work conducted within 'WP1 – Setting the scene: Project setup'. More specifically, the selection of the Use Cases for demonstration draws from the detailed material on Use Case descriptions of deliverable 'D1.2 – MARVEL's Experimental protocol' [1]. Additionally, deliverable 'D1.3 – Architecture definition for MARVEL framework' [2] is an important source for this work, as it contains the refined architecture, which is the blueprint for this release, as well as subsequent releases. D1.3 also provides useful information that D5.8 builds upon, for example, the description of available MARVEL components and their TRL, the grouping of components into building blocks that correspond to architectural layers, and the outline of integration processes that need to be applied.

The work defined within the context of 'WP2 - MARVEL multimodal data Corpus-as-a-Service for smart cities', 'WP3 – AI-based distributed algorithms for multimodal perception and situational awareness', and 'WP4 - MARVEL E2F2C distributed ubiquitous computing framework' serves as a reference for defining the requirements of the infrastructure.

Within WP5, there has been a close collaboration with Task 5.3 and Task 5.4 with regard to the continuous integration of the framework realisation and progress against validated benchmarks, verifying the usefulness of the proposed solutions, respectively. Therefore, this document is related to all releases of the MARVEL framework, starting from MPV (D5.1) [4], through R1 (D5.4) [7] to R2 (D5.6) [9], and also to the benchmarking process (initial version – D5.2 [5] and final version – D5.5 [8]).

1.5 Structure of the document

The structure of this document reflects the infrastructure components delivered to the MARVEL project: computational (HPC and cloud) and storage (data storage and analysis), and is arranged as follows:

- Section 2 discusses the configuration of the HPC system offered to MARVEL from three perspectives: hardware, software, and user services and support. It also includes relevant information regarding granting access to HPC resources and describes infrastructure monitoring tools.
- Section 3 focuses on the cloud computing system, covering overall hardware and system management aspects. Particular emphasis is given to presenting the role of cloud computing system in the MARVEL framework. Additionally, this section outlines implemented solutions for extending the Kubernetes cluster with GPU-equipped nodes at the cloud layer. Finally, it provides information on cloud infrastructure monitoring and contributions to developing a tool to facilitate the monitoring of the entire E2F2C framework's infrastructure.
- Section 4 addresses the storage and data analytics system, focusing on the components from the Data Management and Distribution subsystem of the MARVEL framework and presenting their operations in the context of the Data Corpus.
- Section 5 summarises and concludes the document.

2 High Performance Computing system

This section describes the infrastructure available at the Poznan Supercomputing and Networking Center (PSNC). This HPC centre is equipped with thousands of Central Processing Units (CPUs), numerous graphics cards for processing (GPGPU), storage disks required to provide Massive Parallel Processing (MPP) and, with this the required performance to the applications.

The goal of this elaboration is to better understand capabilities and thus set up the MARVEL infrastructure and provide seamless access to supercomputing resources for the management of the High Performance Computing and High Performance Data Analysis processes.

PSNC provides access to a world-class e-Infrastructure for the scientific community, a specific research and development environment – DIGITAL SCIENCE – for "proof of concept" projects, prototyping or large-scale pilot projects. Another branch of our activity is in DIGITAL INDUSTRY, which is focused on the creation of innovations based on Information and Communication Technologies (ICT), i.e., the execution of specific implementations for various fields of science and industry. Equally important is our work in the context of SOCIAL INNOVATION to spread knowledge and awareness of contemporary technological opportunities among various social groups, as well as actively fighting against digital exclusion. PSNC is a leading HPC centre in Poland with its 7.3 Pflops of computing power, 42 PB of online storage and data management infrastructure, providing direct support for the scientific communities in Poland as well as in Europe (e.g., Nuclear Fusion, Astrophysics, Bioinformatics, Chemistry, Nanotechnology). The PIONIER network is a nationwide broadband optical network that represents a base for research and development in the areas of information technology and telecommunications, computing sciences, applications and services for the Information Society. PIONIER connects 21 Academic Network Centres, Centres of Metropolitan Area Networks (MAN) and 5 of the HPC centres (participating as third parties in PRACE) using their own fibre connections.

Compared to D5.3, this section includes relevant infrastructure updates related to the EAGLE cluster, reflecting the current status at the end of the project. The table of contents has also been expanded to encompass two new sections. The first, titled "Granting Access to HPC Resources," outlines PSNC's internal procedures regarding computational grants. It also provides information on the new methods for submitting and managing HPC jobs introduced in the project's second phase. The other is titled "Monitoring Tools." This part offers detailed insights into monitoring the Eagle cluster and its individual components.

2.1 Hardware

PSNC boasts two dedicated data centres equipped with air and liquid cooling systems, video monitoring, fire protection systems and 24/7/365 monitoring. These include a primary, 2x 820 m², brand new data centre in the Berdychowo district of Poznan, delivered in the first quarter of 2015 (called "BST") and a secondary data centre, in the Poznan central district. The total peak performance of ca. 96K CPU cores is over 7.3 PFlops. PSNC systems have appeared in the Top 500 list several times. PSNC is part of the European HPC infrastructure (PRACE), European and national grid infrastructure (EGI, PL-GRID), national (PLATON e-Science Platform, National Data Storage) and European Data Infrastructure (EUDAT). A view of the currently installed Petascale system is presented in Figure 1.

Figure 1: PSNC Server Room

The data centres are equipped with all the required environmental systems and components including redundant power supplies, own transformer stations, UPS, power generators, fire protection (detection and active extinguishing systems), security and access control systems and finally, 24/7/365 monitoring.

Furthermore, PSNC operates the modern backbone network based on 100 Gbps links between the major HPC sites and 100 Gbps links to bigger academic communities. It participates in the international academic networking consortium GÉANT, which offers multiple 100 Gbps to European countries. PSNC also peers with commercial providers in Poland through links in Poznan as well as with international operators through peering points in Hamburg and Amsterdam, amongst others.

In the following subsections, the resources made available for the implementation of the project's tasks will be discussed. This overview has been divided into three sections: HPC specs (supercomputer called EAGLE), its disk and tape storage, and deployed training infrastructure for rapid prototyping.

2.1.1 EAGLE specification

EAGLE is a PSNC-owned HPC cluster, which delivers conventional computing power and a dedicated GPGPU acceleration environment tailored for AI and big data analytics. Manufactured by Huawei, it consists of 1233 nodes featuring Intel Haswell and Broadwell CPUs. Each node is equipped with 64GB to 256GB of RAM memory and interconnected through a high-speed Infiniband FDR network. These resources are available to the MARVEL users in a shared mode considering the current machine load and demands of other users.

In 2020, EAGLE was expanded by another 1300 CPU servers and 9 servers equipped with GPUs. In Mid-2023, the old servers were removed from the cluster as in Q4 PSNC began another expansion of the machine. The current (end 2023) system configuration is presented in Table 1 - Table 4.

EAGLE System Specification

Table 1 contains a detailed hardware specification of the EAGLE system at PSNC.

Table 1: EAGLE System Specification at PSNC

Nodes	Cores	Memory	Interconnect	Amount of Storage	Storage Performance
1320	63000	300 TB	56 Gb/s Infiniband FDR	4.6 PB Lustre 1 PB /home 10 PB archive	32 GB/s Lustre File system 10 GB/s /home CNFS Cephfs

EAGLE CPU Nodes

Table 2 contains the specification details of CPU nodes of the EAGLE system.

Table 2: EAGLE CPU nodes

Node merchant name	CPU model	No. of nodes	No. of processors and cores	RAM per node	Computing power per node	Node class tag
HUAWEI CH121 V3	Intel Xeon Platinum 8268	1320	2x24	192/384 GB	4.4 TFLOPS	Intel,cascade

EAGLE GPU Nodes

Table 3 contains the specification details of GPU nodes of the EAGLE system.

Table 3: EAGLE GPGPU nodes

CPU model	No. of nodes	No. of processors and cores	RAM per node	GPU model	No. graphic cards per node	Computing power per GPU
Intel Xeon Gold 5115	3	2x10	92 GB	NVIDIA V100	2	7.8 TFLOPS
Intel Xeon 6242	9	2x16	384 GB	NVIDIA V100	8	62.4 TFLOPS

EAGLE Operating System

Table 4 contains the information about the EAGLE operating system, its domain name, and the task management system.

Table 4: EAGLE operating system

Type	Domain Name	Task management system in batch mode
GNU Linux	eagle.man.poznan.pl	SLURM

2.1.2 HPC storage and tape library

This section provides information on the storage systems linked to the HPC. Similarly, as for HPC, presented resources are shared among different users so its efficiency depends on the current load generated by other users. The type and performance of the storage system at PSNC are delivered in Table 5.

Table 5: PSNC Storage

Storage	Storage Performance
LTO 5 3.5 PB	2 GB/s
Jaguar 28 PB	4.2 GB/s

In summary, a detailed view of the schematic infrastructure and their relations at PSNC can be obtained from Figure 2.

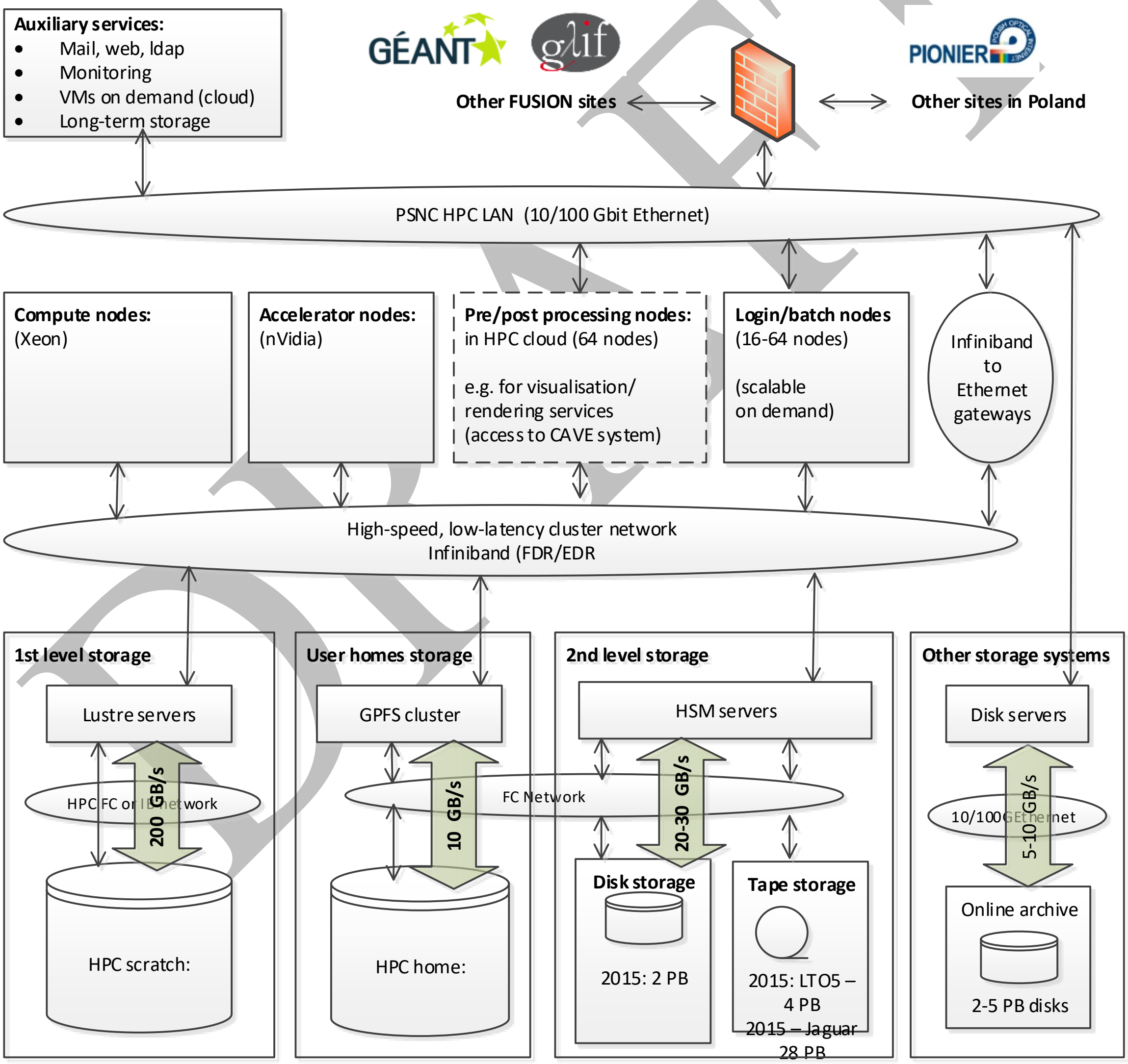

Figure 2: PSNC data infrastructure and correlation with HPC

The storage connected to the HPC is intended for storing short- and medium-term data while performing calculations. The short-term space is called SCRATCH and has a capacity of several

PBs (shared by all users) used to store temporary calculation data. The medium-term space available through GPFS is called PROJECT DATA and is subject to the constraints of the available space. For MARVEL, the limit is currently set to 100 GB but can be increased if needed. The results of the calculations, which are not subject to further processing, are transferred for long-term storage.

2.1.3 Training cluster

MARVEL is using the EAGLE cluster to run its applications, especially those which require access to GPGPUs. This requires the installation and configuration of many services that must work properly with the cluster system, taking into account the requirements of the queue system. This involves performing many tests, which are much easier to carry out in a prototyping environment.

PSNC also offers access to infrastructure for rapid software prototyping. It is important to realise that supercomputers in the HPC domain tend to be heavily loaded, making it compulsory to wait for the computation to run in the job queue. Having a smaller, multi-node computing cluster configured in a manner similar to the original supercomputer is essential to run jobs quickly without having to wait in a system queue. The proposed training infrastructure has a similar Operation System (OS) and queuing system, but only fewer cores and memory are available.

The configuration of the training cluster is detailed in Table 6.

Table 6: Configuration of training infrastructure

Name	Address		Cores	Memory	Disk space
Training0	62.3.171.192	sophora-192.man.poznan.pl	32	32 GB	40 GB HDD + 1000 GB volume
Training1	62.3.171.143	sophora-143.man.poznan.pl	32	32 GB	40 GB HDD
Training2	62.3.171.224	sophora-224.man.poznan.pl	32	32 GB	40 GB HDD
Training3	62.3.171.194	sophora-194.man.poznan.pl	32	32 GB	40 GB HDD
Training4	62.3.171.167	sophora-167.man.poznan.pl	32	32 GB	40 GB HDD

The topology of the training infrastructure setup for MARVEL prototyping is presented in Figure 3.

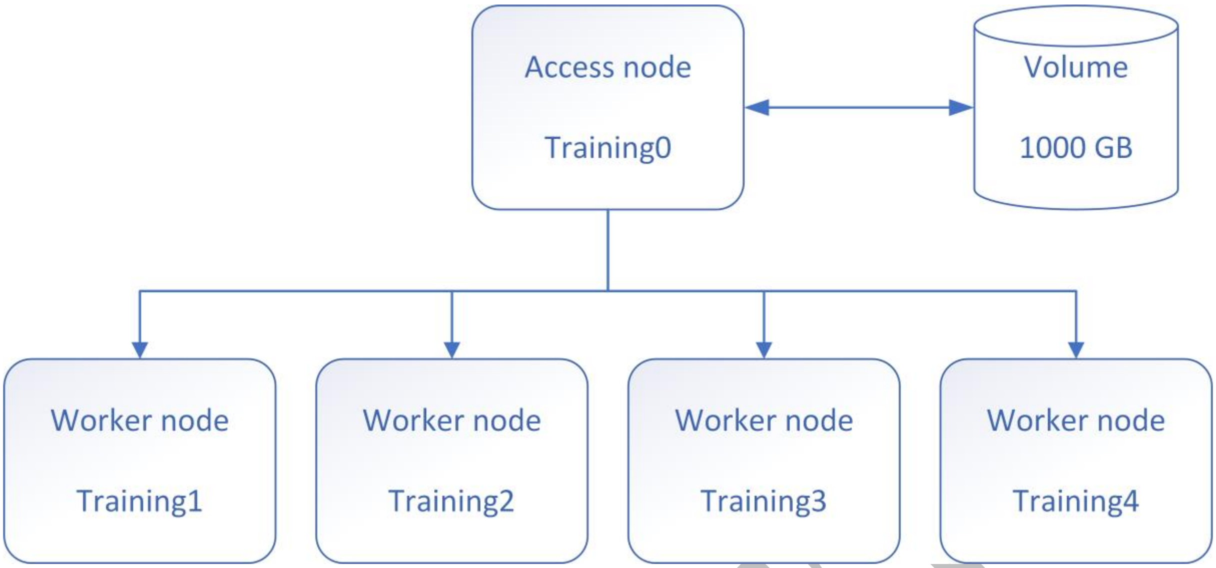

Figure 3: Topology of the training infrastructure

2.2 Software

This section describes the availability of software packages in PSNC for the MARVEL project. Software is an essential aspect in high performance computing as HPC applications differ little from standard computing applications except for the fact that HPC applications are developed to run in parallel and require high performance, physical infrastructure with appropriate libraries and compilers to achieve the highest possible degree of efficiency and effectiveness.

Jobs can be submitted in two modes: batch mode, where application(s) are run in a series without manual interventions and interactive mode, where user interaction (presence) is mandatory to manipulate the launch of programs, deciding on an ongoing basis to take the next processing steps. Batch mode is always preferred to run these applications in clusters, which will be discussed in detail in the following sections (2.2.1 - 2.2.4). However, this section will only discuss the available cluster management software packages and user application execution on a cluster available in PSNC.

2.2.1 Cluster Management Software

In this section, the process related to the batch system, job types, and job management is described. This knowledge is necessary to properly use HPC systems and avoid administration issues. The EAGLE server uses SLURM [10] as resource management system. A batch job can be submitted using the `sbatch` command (implemented by SLURM) along with its configuration files where the user can specify requirements for the application (e.g., memory needed, number of cores, etc.) and load necessary modules or stage files. Finally, the status of the submitted jobs can be accessed by using `sinfo`.

Queue

Running tasks through the SLURM queuing system takes place within queues, which differ in priority, limits and access rights. EAGLE has the queues presented in Table 7.

Table 7: Queue configuration on the EAGLE supercomputer

Queue Name	Job Timeout	Default timeout job	Standard RAM	Comments
standard	7 days	1 day	2 GB	The queue is intended for performing tasks on one node
fast	1 hour	10 minutes	-	High priority; The maximum number of nodes is 1
bigmem	7 days	1 day	2 GB	Queue for tasks requiring more than 128 GB per node. High priority but only 59 nodes belong to this partition
tesla	7 days	1 day	2 GB	Queue with GPU nodes

Submitting jobs

Prior to job submission, the appropriate command launching the application has to be embedded in the script to be correctly read by the queueing system:

`/home/users/user/submit_script.sl`

Example:

```
#!/bin/bash -l
#SBATCH -N 1
#SBATCH --mem 5000
#SBATCH --time=20:00:00
/path/to/binary/file.exe >/path/to/output/file.out
```

To submit the job to the selected queue please use #SBATCH -p parameter:

```
#!/bin/bash -l
#SBATCH -N 1
#SBATCH --mem 5000
#SBATCH --time=20:00:00
#SBATCH -p long
/path/to/binary/file.exe >/path/to/output/file.out
```

Jobs can be submitted using `sbatch` command:

```
sbatch /home/users/user/submit_script.sl
```

Submitting interactive jobs

In this mode, interactive jobs can be submitted by executing the following command:

```
srun --pty /bin/bash or srun -u /bin/bash -i
```

The first command allocates a pseudo terminal that simplifies the work on a remote console. In case of any problems, please use only the second command.

Interactive jobs in graphic mode

From the user's point of view, it is sufficient to log in to the cluster with the -X option

`ssh -X` and execute the following command:

```
srun-interactive -N 1 -n 1
```

Job submission using GPU cards

To submit a job to nodes equipped with GPU cards, it is required to use the tesla partition and add the following section to the submission script:

```
#SBATCH --gres=gpu:<no. of cards for every task>
```

An exemplary job using two cards should contain the following sections:

```
#SBATCH --gres=gpu:2
#SBATCH --partition=tesla
```

A selected number of applications are enabled to use GPUs, either by built-in functionality or using a dedicated module, usually containing "CUDA" in its name.

```
namd/2.10-ibverbs-smp-cuda     <- GPU supported version
namd/2.10-multicore(default)
namd/2.10-multicore-cuda       <- GPU supported version
namd/2.10-multicore-mic
namd/2.12-ibverbs
namd/2.12-ibverbs-smp
namd/2.12-ibverbs-smp-CUDA     <- GPU supported version
```

Checking status of the queue

To check what jobs have been submitted by a given user, please execute the command:

```
squeue -u username
```

Removing jobs

To remove the job, please use the `scancel` command and specify the corresponding job id as a parameter. Both, waiting and running jobs can be removed.

```
scancel job_id
```

2.2.2 Available software

Description of the software available to users at the HPC system. A complete documentation for installed software along with sample scripts to make use of them can be found on the website [11].

Name	Version(s)	Purpose
GCC	4.8, 4.9	Development/Compiler
Intel Parallel Studio XE Cluster Edition: Intel C++ Compiler Intel Fortran compiler Intel Data Analytics Intel MKL (C++, Fortran) Intel TBB (C++) Intel IPP (C++) Intel Advisor (C++, Fortran) Intel Inspector (C++, Fortran) Intel VTune Amplifier Intel MPI (C++, Fortran) Intel ITAC (C++, Fortran) IMSL (Fortran)	15.0.0, 15.0.3, 16.0.0	Development/Compiler Development/Compiler Acceleration Library Math Library Threading Library Media and Data Library Vectorisation/Optimisation Debugging Tools Performance Profiler MPI Library MPI Analyser and Profiler Numeric Library
AMD Core Math Library	5.3.1	Development/Libraries
Abaqus	6.14-2	Application/Engineering
Abinit	7.4.3	Application/Science
Amber	14	Application/Science
Bowtie	1.0	Application/Science
Codeanalyst	3.4.18	Application/Science
Cudatoolkit	7.0.28	Application/Science
Gaussian	09.D.01	Application/Science
Gromacs	5.0.4	Application/Science
Hmmer	3.1b1	Application/Science
Mapdamage	2.0	Application/Science
Matlab	R2013a	Application/Science
Mumax	3.8	Application/Science
Namd	2.10	Application/Science
Orca	3.0.3	Application/Science
Plink	1.0.9	Application/Science
Quantum Espresso	5.2.0	Application/Science
Rna-Seqc	1.1.8	Application/Science
Rsem	1.2.18	Application/Science
Siesta	3.2	Application/Science

Tabix	0.2.6	Application/Science
Tophat	2.0.13	Application/Science
Trinityrnaseq	2.0.6	Application/Science
VcfTools	0.1.14	Application/Science
Velvet	1.2.10	Application/Science
Vowal-wabbit	8.0	Application/Science
Xenome	1.0.1r	Application/Science

2.2.3 Module management

This section discusses the modules' management at the HPC and the standard methods used to set up the user environment for clusters. All available compilers and libraries can be listed by typing the module available command. Every user of the systems at PSNC is allowed to compile and run their own codes. It is also possible to compile and run third-party software using either GCC or Intel compilers [12]. Available compilers, applications and libraries (e.g., MKL) can be listed using the module list command. To make a module active, one has to issue the module load <module name> command. If the compiled application is using any library that needs to be loaded during the compilation, it is required to load these modules before the actual program is executed.

Every application that is storing intermediate data during computation or is using large datasets, needs to copy the data to the /mnt/lustre directory. This directory is shared among all nodes that are used for any application computation. Since this is a temporary space that can be erased at any time, output data should be saved to the user's /home directory afterwards.

All applications must be launched using the queue system. Applications executed on the login node, manually on the allocated nodes or those utilising the /home directory for storing intermediate results will be terminated without warning.

Modules

Modules is a widely used concept that aims to the simplification of different software versions in a precise and controlled way. Modules, use a list of standardised methods to manage the user environment for clusters. It is enough for the user to load a module associated with a given application to set all required environment variables (PATH, LD_LIBRARY_PATH INCLUDE, etc.) accordingly. If the user wants to use a different version of the application, he/she has to unload the current-in-use module and load the module associated with the new version.

Usage

module list	To list loaded modules please use this command
module avail	To list modules available for a given user please use this command
module help <module_name>	Additional info about the module can be obtained by executing this command

module load <module_name>	Loading a module is done using this command
module unload <module_name>	A module can be unloaded this way
module show <module_name>	Checking for environment variables modified by given module is available via this command

Settings customisation

In the modules environment, a user can decide which modules will be loaded when logging into the system. To do this, one has to edit the file `~/.bashrc` or `~/.bash_profile` configuration and add the appropriate lines. Below is an example where a variable ModulePath has been added to the directory, which contains the user module and the module open64 compiler in version 4.2.5.2.

```
[username@hostname ~]$ cat ~/.bash_profile
...
# Environment Modules
# Prepend directory to the MODULEPATH environment variable.
module use --append $HOME/.modules/my_modules
# Modules loaded at login time.
module load open64-4.2.5.2
...
```

2.2.4 User Own software installation

Users are allowed to install their own software in the `$HOME` directory. This section explains how to properly use already delivered modules containing required software such as libraries, compilers, interpreters, and other supportive applications and tools. The installation procedure of the own software is presented in the subsequent section.

Install own software

It is important to note that own software compilation on the head node is forbidden. It is possible to install own software in the /home directory without asking administrators for permission. Users can also ask the administrators to compile and install any kind of software by writing a message to the standard support email. In the consecutive part of the subsection, we discuss how it should be done correctly.

Software Compilation

To compile the software on HPC system, we need to access the interactive console. To do so, a user must submit an interactive job and compile it in an interactive mode to utilise the resources of the remote node. Although the head node provides resources as well, these resources are limited and shared amongst all the users. Consequently, the compilation of an application on the head node impacts other users and shall be avoided in any sense.

`srun --pty /bin/bash`	Example of using SLURM
`srun --pty -n 12 /bin/bash`	To access a node with a certain number of cores a user needs to specify -n parameter
`srun --pty -n 12 --mem 16000 /bin/bash`	If a user wants to reserve a certain amount of RAM, the --mem parameter needs to be used
`module load gcc`	When the user accesses the console directly at the node, the modules for the compilation need to be loaded

`./configure --prefix=home_folder`
`make`
`make install`
	The last step is to compile the software in the /home directory, according to the manufacturer's recommendations

`make -j 12`	To utilise compilation processes on several cores, the user can add the -j parameter

Interactive Job

An interactive job allows a user to obtain a shell on a computational node. Because compiling and running any programs on the head node is not allowed, interactive jobs are one way to run graphical user interface software or compile a particular application. The following is an example of how to run a task in interactive mode.

`srun --pty /bin/bash`	Interactive job submission
`srun --pty -n 12 /bin/bash`	A user has to add the -n parameter to access the node with the specified number of cores
`srun --pty -n 12 --mem 16000 /bin/bash`	If a user wants to reserve a certain amount of RAM, the --mem parameter needs to be used

2.3 User Services and Support

Efficient cooperation of MARVEL pilots with infrastructure technical support is essential to fully exploit the potential of available resources. As a hosting centre and national network operator, PSNC has teams of people who can fully support all aspects of computing (especially HPC) and data infrastructures. Below are some areas where cooperation was established using the synergy effect.

Operation of the IT infrastructure

PSNC has been hosting HPC machines for over 25 years and has an experienced team responsible for physical and software maintenance for various types of systems (clusters, GPU machines, and large SMP systems). Currently, the scope of the support covers all issues related to the computing infrastructures, including troubleshooting, compilation, and runtime problems. MARVEL deploys its services on novelty, demanding, and reliable machines managed by an experienced team of professionals following established rules.

Customised User Support

Due to the fact that MARVEL requires specific configurations for its services that fully use the current technical solutions, support and immediate response to emerging problems are expected, which is an indispensable part of contacting the support department. In addition to the infrastructure mentioned in the previous sections, PSNC offers a set of services and support activities intended to assist effective, reliable, and safe usage of the PSNC infrastructure. The services contain in particular user support for daily operations as well as support for the optimisation and efficient usage of the PSNC computing clusters.

Help Desk solution at PSNC

PSNC has supported users with the utility of achieving fast response times on their problems coming from distinct areas of computational infrastructure for years. The Network Operation Center (NOC) is the "Operator on Duty" on a 24/7/365 basis. It reacts in emergency situations and can provide tentative support nearly online. However, all MARVEL user queries and problems must be registered in the PSNC Ticketing System (TTS). Figure 4 demonstrates an example of the PSNC's ticketing mechanism available for the MARVEL users.

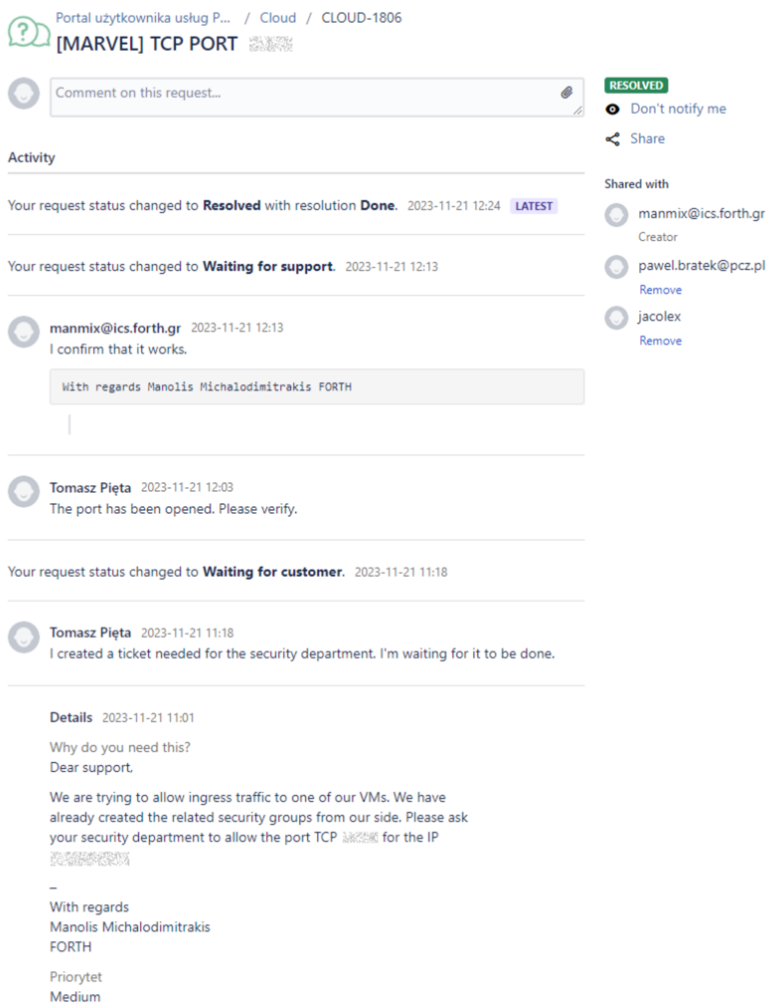

Figure 4: Example of MARVEL request registered in the PSNC Ticketing System

End user support

End User Support corresponds to the service for the user application enhancement in a broader sense. It mainly consists of improving their codes. The permanent support has been addressed to all MARVEL users having HPC accounts in PSNC.

User support on security

During the project, support was extended to MARVEL users engaged in the HPC fusion infrastructure, recognising the pivotal role that the users play as a critical link in the security chain for any system or infrastructure. The emphasis was on enhancing security awareness, acknowledging that even a well-protected infrastructure can be compromised if users exhibit inaccurate behaviours.

Training centre

Since 2005, PSNC has organised training sessions for HPC users coming from different research areas. During approximately 100 trainings, the most important thematic groups have been identified: sequential programming, parallel programming or queuing system usage, just to name a few. The training offer is supported by qualified HPC staff of PSNC and external specialists from the Technical University of Poznan.

As previously described, PSNC offers a wide range of services to support its users. Users were also guided by using the infrastructures, tools, and services provided within MARVEL while additional implementation support was given by technical partners. Also, consortium members and users of the infrastructure that MARVEL provides, can take advantage of the rich portfolio of training provided by PSNC such as parallel programming, scientific visualisation, machine and deep learning, or performance optimisation, to name a few.

2.4 Granting access to HPC resources

Compared to D5.3, this section expands the table of contents describing PSNC's internal policy related to granting access to HPC resources. Furthermore, it also provides information on the new methods for submitting and managing HPC jobs introduced in the project's second phase.

A unified process for requesting HPC resources has been defined and implemented at PSNC. The domestic user represented by Principal Investigator, must submit the official proposal (computational grant) to the Council of Users. The Council of Users is responsible for the review process and takes the decision in terms of granting access. The access rules are regulated and expressed in the "Rules of the usage of PSNC's computational resources". Summing up, the proposal is evaluated, and then the request for account creation is passed to the HPC Operational Department. Each year, a report on the computations and scientific results is expected from the Principal Investigator. The maximal computational grant duration is three years. In summary, the Principal Investigator needs to provide the following information on their grant:

1. Personal data with their scientific portfolio;
2. The title and description of the work foreseen to be processed;
3. Core time requirements;
4. Storage space requirements;

5. The list of the grant's stakeholders (users).

At present, the whole grant management is processed in an electronic manner in the portal developed by PSNC [13]. The portal is developed using an Open Source software, the code of which is still under development and includes continuous updates. The system is responsible for:

1. Gathering the data required to fill up application forms;
2. Management of the groups of users associated with the grant proposal;
3. User accounts and their accounting;
4. Assigned limits guarding;
5. Reporting process and the evaluation of its results by the Council of Users body;
6. Maintaining and presenting user jobs' data in a smart manner;
7. Generating the yearly utilisation reports on demand with the multi-criteria options.

As it was already mentioned, the work on EAGLE is organised around computational grants. To facilitate the project work, one of them has been established: "MARVEL grant" - Grant no. 505. Of course, it illustrates only logical organisation which follows pilots'-oriented approach in the MARVEL project.

To use EAGLE resources, each user must have an account registered at the HPC portal [13]. The portal provides overall information about computational grants (validity, limits, machine allocation), consumed resources and publications presenting the results of computations. Each user account is assigned to the specific computational grant as well as the HPC cluster. Access to the cluster via SSH currently supports only SSH keys, so one has to provide a public key on the user's profile page.

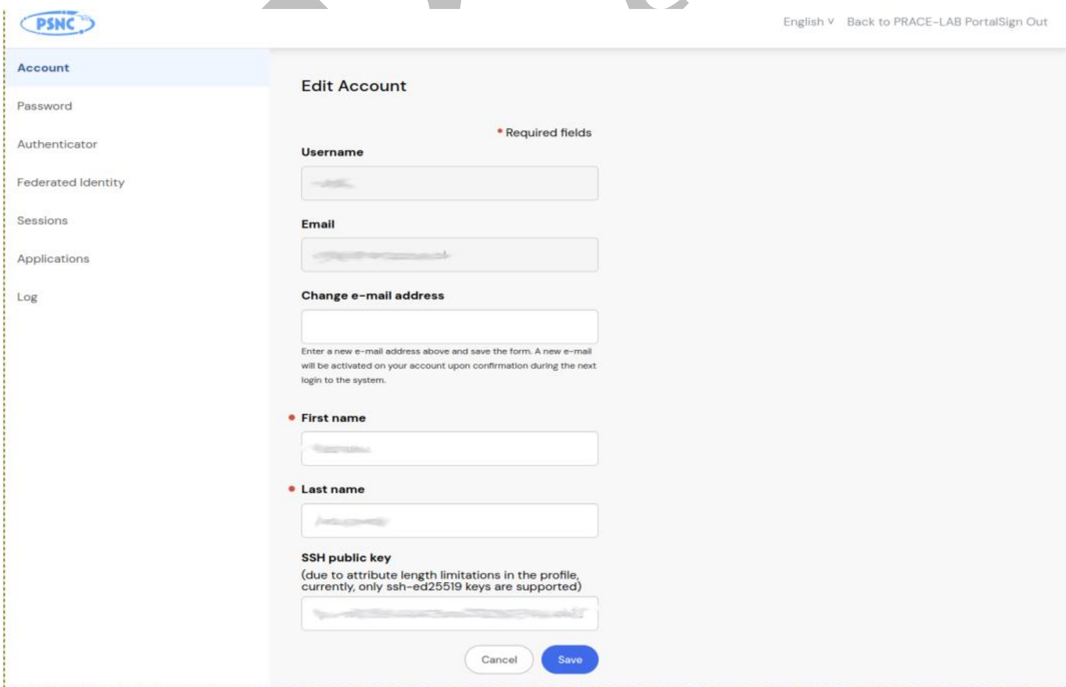

Figure 5: HPC cluster user profile

The EAGLE system can be accessed via SSH:

```
user@host:~$ ssh psncusername@eagle.man.poznan.pl
```

Alternatively, one can use web-based interface for submitting jobs, as presented in Figure 6. One can choose one from pre-defined templates or create its own. The following steps depend on the application template, as different parameters may be required for different applications. For example, using a Python template requires passing files with Python code.

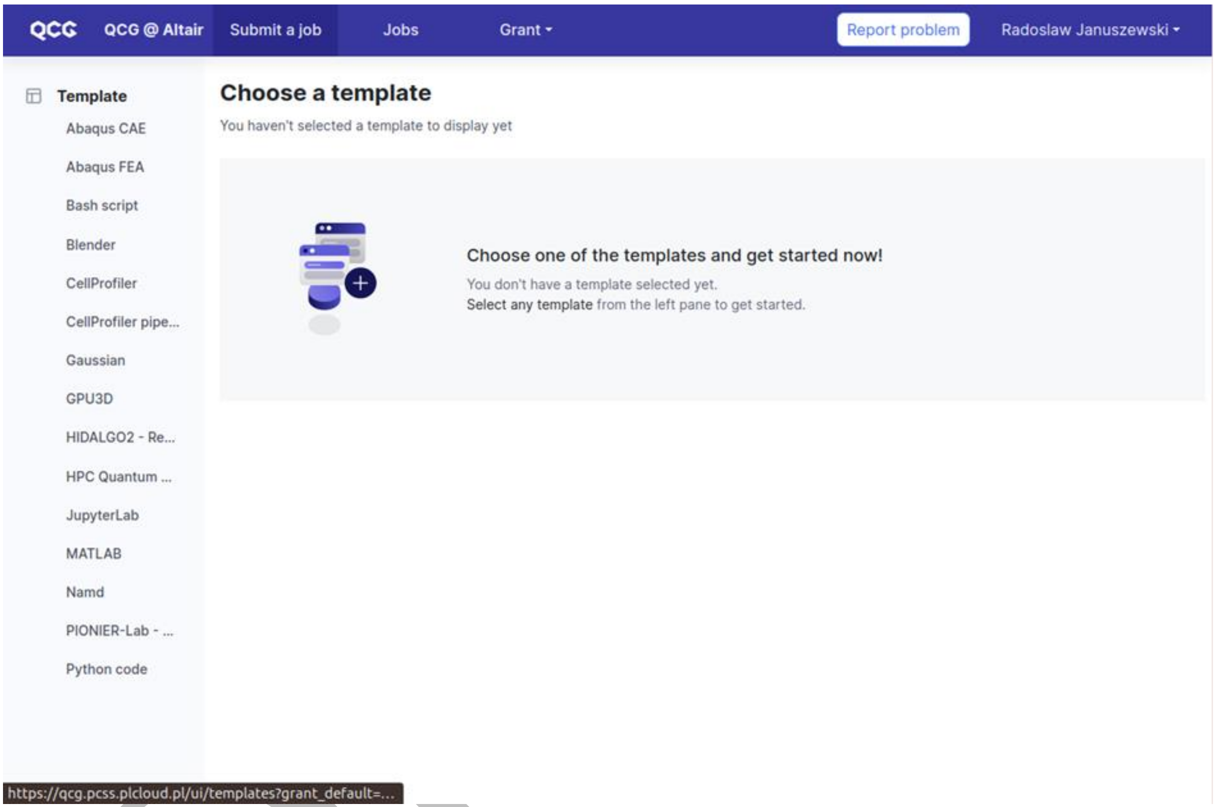

Figure 6: Web-based interface for submitting jobs to the HPC cluster

After selecting or uploading a file with code one only has to click "submit" and the code will be executed on the HPC cluster.

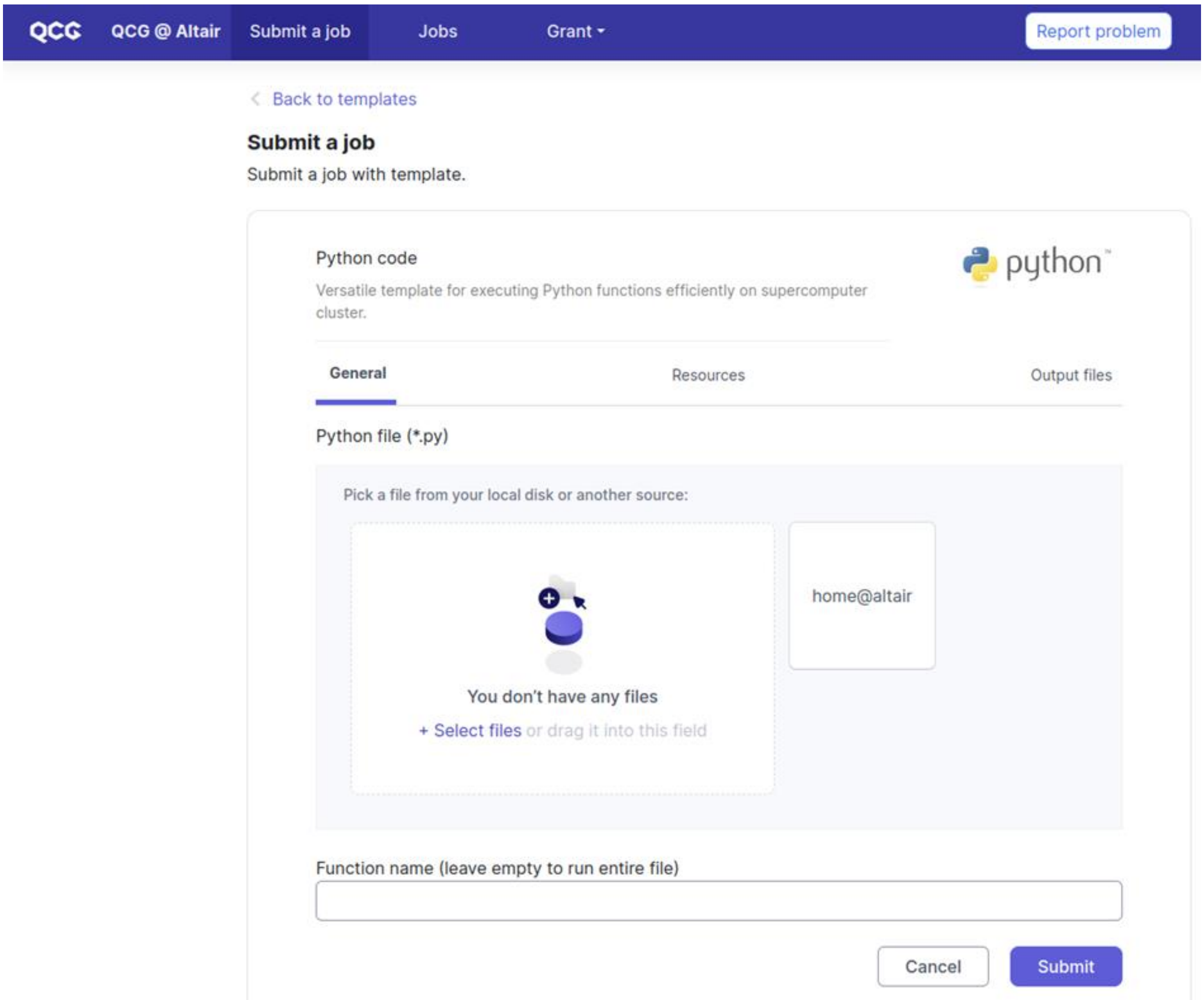

Figure 7: Submitting a job to the HPC cluster using web-based interface

It is also possible to see the details of the submitted job as presented in Figure 8.

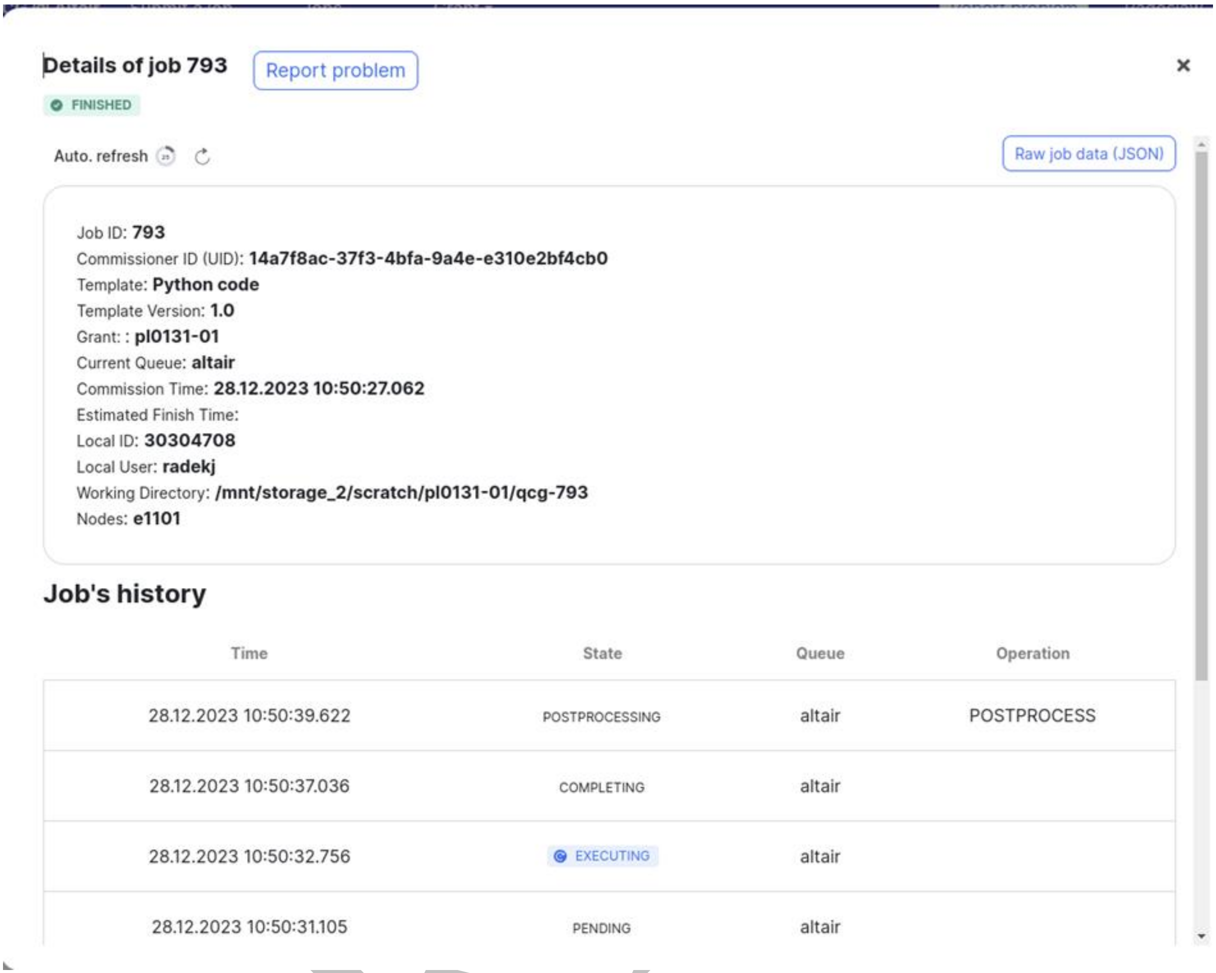

Figure 8: The details of a job submitted to the HPC cluster using web-based interface

Another option is to use a web-based SSH client that is accessible from the portal. This method is similar to using a traditional SSH client. The only difference is that one does not have to install additional applications or set up keys.

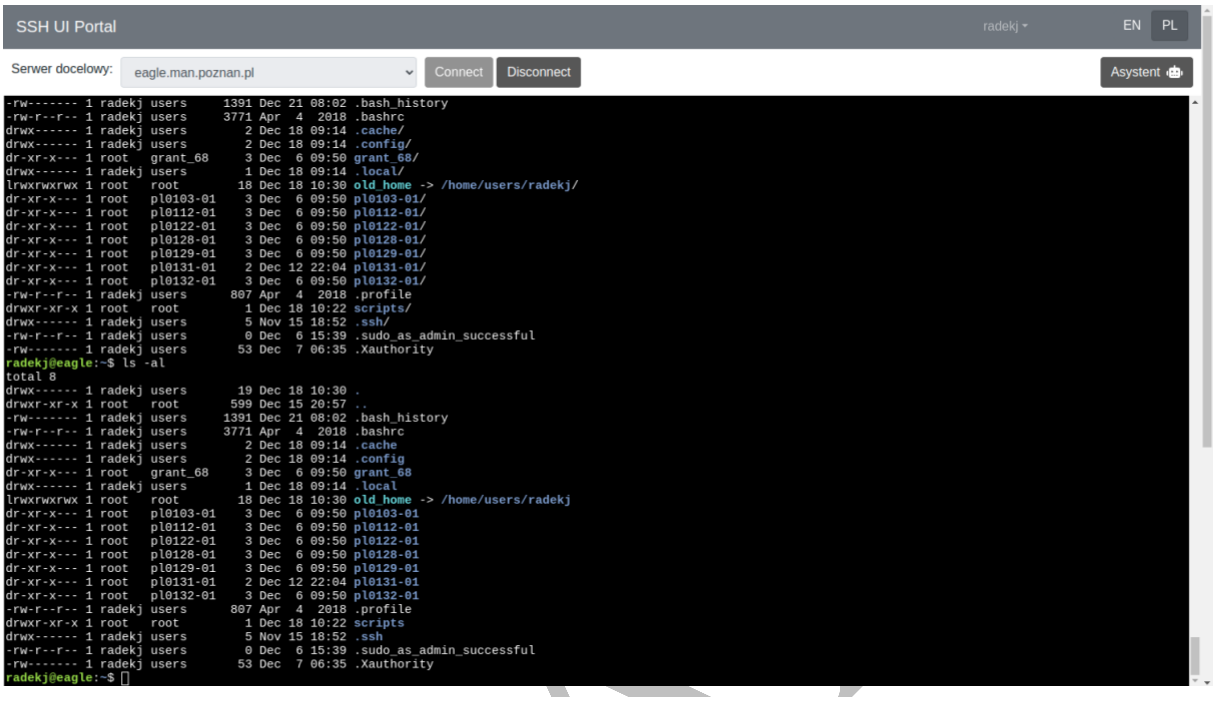

Figure 9: Web-based SSH client for submitting jobs to the HPC cluster

Both new ways of interaction with the HPC cluster can be accessed directly from the browser using [14] for web interface or [15] for web SSH client or via the main portal selecting the appropriate tool from the service detail page as shown in Figure 10.

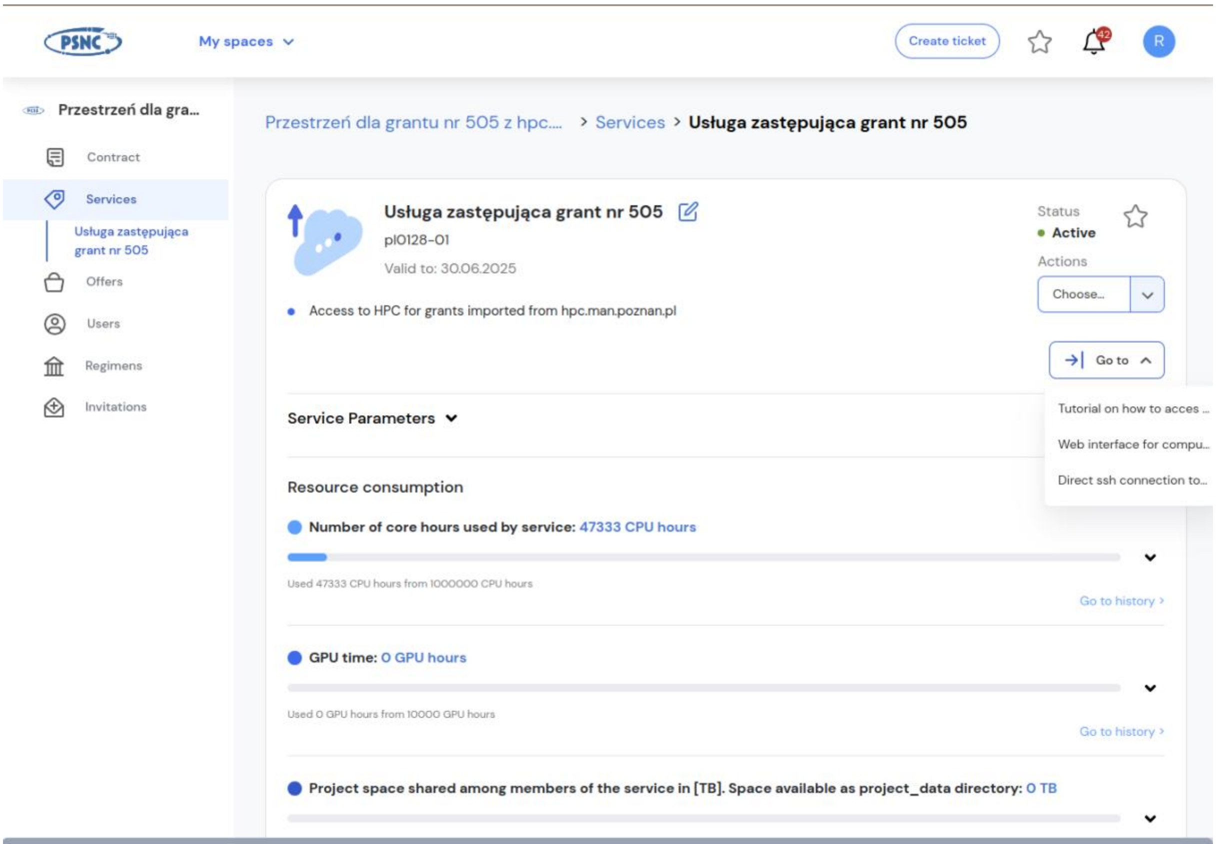

Figure 10: Services page of the HPC cluster main portal

Access to the Virtual Machines is granted based on the RSA public keys of the user. Operations of the user account creation and importing of the public keys are handled by the Virtual Machine administrator. The administrator is also responsible for server maintenance and the installation of all necessary software.

2.5 Monitoring Tools

This section expands the table of contents describing monitoring tools related to the EAGLE cluster. This information was not presented in D5.3 but served as valuable information from the beginning of the project.

EAGLE cluster is being monitored on physical (servers, switches, state of links, etc.) and logical levels. From MARVEL's perspective, more important is logical level monitoring on which all basic components are tested and checked for proper behaviour. The monitoring and SLA levels are related to multiple tests that evaluate both physical aspects of the infrastructure and available functionality. All metrics are gathered and calculated using Zabbix software.

▼ Eagle	OK		0.7836	99.2164 / 99.9000
NIS server - hpc-nis is unavailable by ICMP	OK		0.0000	100.0000 / 99.9000
▶ Slurm service	OK		0.0000	100.0000 / 99.9000
▶ Storage	OK		0.7806	99.2194 / 99.9000
▶ Subnet manager	OK		0.0000	100.0000 / 99.9000
▼ UI	OK		0.0260	99.9740 / 99.9000
Eagle UI is responding - eagle-ui is unavailable by ICMP	OK		0.0231	99.9769 / 99.9000
Slurm is responding on UI - Slum client - cannot query queue info!	OK		0.0000	100.0000 / 99.9000
SSH service on Eagle UI - SSH service is down on eagle-ui	OK		0.0252	99.9748 / 99.9000

Figure 11: Example of measurement results from the EAGLE cluster

Figure 11 presents an example of measurement results from the EAGLE cluster. One can see that there are several columns listed. The first column holds the name of the component followed by the current state. The next column shows unavailability levels both in graphical and numerical format. The last column presents the current and threshold values of the SLA for a given module. Some of the modules are aggregates of multiple lower-level metrics, but in most cases, these reflect the redundant nature of each service.

The tested modules of EAGLE are:

- EAGLE: aggregate value that reflects the overall availability level of the HPC cluster. This aggregate is calculated assuming that if any of the underlying services are not working at a given moment, the entire service is considered as not available. From a technical perspective, it is not always true, but it reflects the “worst case” scenario.

- NIS server: reflects the state of the IDP provider. The unavailability of this service is unlikely to affect ongoing processing on the cluster, but it will prevent interactive work and the start of new calculations.

- Slurm: reflects the state of the resource manager. The unavailability of this service prevents new jobs from starting and may cause disruption in currently running calculations.

- Subnet manager: the component responsible for managing a high-speed cluster network. The unavailability of this service will prevent applications from accessing fast storage and, in the case of MPI-based parallel applications, will cause fatal errors in processing jobs. Currently, the cluster consists of several Infiniband fabrics, so there are failure zones: not all processing is affected by this kind of error.

- UI: the component responsible for monitoring the traditional user interface available via SSH service. In this component stare, several processes are monitored both from basic functionality (work or does not work) and from a performance perspective.

3 Cloud computing system

PSNC cloud computing system is located in two locations:

- BST which is composed of 170 servers, 35TB memory and 2.7 PB of storage space
- DCW which is composed of 25 servers, 10TB memory and 0.7PB of storage space

These locations are interconnected with a fast backbone network with a bandwidth of 400Gb. The topology of the cloud is presented in Figure 12.

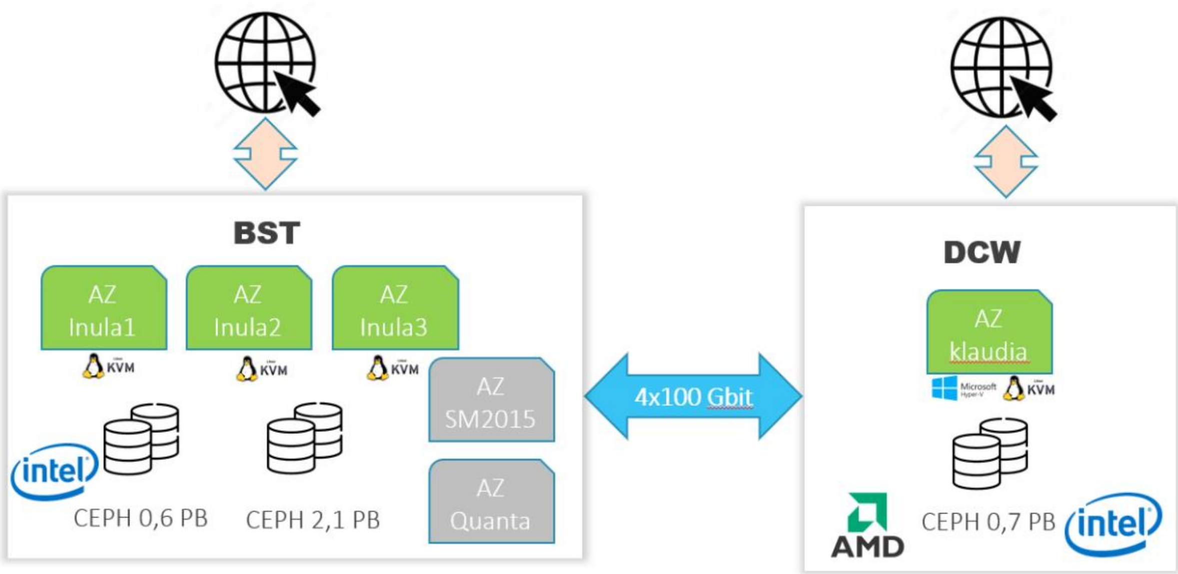

Figure 12: Topology of the cloud computing system

Compared to D5.3, this section includes relevant updates reflecting the current status at the end of the project. Additionally, the content has been expanded to clearly explain the role of the cloud computing system from the project's perspective and highlight achievements from the second part of the project. Specifically, this section describes implemented solutions for extending the Kubernetes cluster with GPU-equipped nodes at the cloud layer. Furthermore, it provides information related to cloud infrastructure monitoring and outlines contributions to developing a tool that facilitates monitoring the entire E2F2C framework's infrastructure.

3.1 Cloud hardware

This subsection details the specifications of the equipment included in the cloud system.

Table 8: BST cloud area specification

Availability zone	Processor	Cores	Memory [GB]	Backbone network [Gbit]	No. of servers
Inula-1	Xeon(R) CPU E5-2697	56	192	Infiniband FDR (56)	48
Inula-2	Xeon(R) CPU E5-2697	56	192	Infiniband FDR (56)	48
Inula-3	Xeon(R) CPU E5-2697	56	256	Infiniband FDR (56)	48
Supermicro-2015		40	256	Ethernet 10 Gbit	14
		32	256	Ethernet 10 Gbit	4
quanta		32	96	Ethernet 10 Gbit	5
Total		**8912**	**35793**		

Table 9: DCW cloud area specification

Availability zone	Processor	Cores	Memory [GB]	Backbone network [Gbit]	No. of servers
Klaudia	Xeon Gold 6138	80	384	Ethernet 10 Gbit	6
	AMD EPYC 7451	96	512	Ethernet 10 Gbit	14
Total		**1824**	**9472**		

3.2 Cloud management system

The popularity of cloud computing is rapidly increasing both in the industrial and academic domains. While computing power-intensive tasks are still mainly performed on supercomputers, analytics, data visualisation, database engines, and web page services require a runtime environment that is primarily flexible and reliable. For this type of purpose, PSNC has a cloud environment that allows you to run virtual machines with the Linux Operating System (i.e., Redhat, Centos, Ubuntu, Debian, etc.) and Windows. The system, operating under the control of the OpenStack system, connects computing and application servers with data storage solutions and the network. With the utilisation of cloud technologies, services and applications running on virtual machines become resistant to hardware failures - in the event of problems, the machines are automatically started on redundant servers. An additional advantage is the ability to adjust the amount of available resources based on the needs of the application.

PSNC provides:

- **Flexible virtual machines running for network services under the control of the selected operating system**. If the machine handles network services such as e-mail, websites, etc., the selection of such a machine is the best solution.
- **Computing machines in a cloud environment**. Despite the fact that the most processing intensive tasks are counted on supercomputers, the requirements of some applications (e.g., Windows application) make the use of a computer cluster impossible. However, in such cases, one can still benefit from servers to run virtual machines, which have 10-20 times greater performance than even the fastest laptops. Thanks to the

appropriate configuration of the cloud environment, the virtual machine will have a performance similar to the physical server with a given configuration - the physical resources of a given server will be dedicated to this virtual machine.

- **Private clouds**. Instead of a single machine, it is possible to create a private environment in which, within the available resources (processors, memory, disk), it is possible to independently create many virtual machines with configurations corresponding to a specific application. For this type of application, it is also possible to define network configuration between machines, firewall rules both for the entire environment and for individual machines, as well as the use of advanced mechanisms such as load balancers.
- **Cloud data storage and management systems** - In this case, classic matrix systems and cluster file systems integrated with the computing and service cloud environment as well as object-oriented data storage systems (S3, Ceph and Swift) can be used.

PSNC cloud computing system offers access to the following services:

- virtual machines
- private cloud (VPC)
- virtual disks
- object Storage
- shared storage
- API
- orchestration
- load balancers
- VPN
- firewall
- control panel
- Linux OS
- Microsoft SPLA

3.3 Cloud computing system from project perspective

The MARVEL project encompasses ten use cases distributed across three pilot locations: Malta (GRN pilot), Trento (MT pilot), and Novi Sad (UNS pilot). The use cases span across various categories, addressing activities related to traffic monitoring, anomaly analysis, detection of criminal/anti-social behaviours, area-specific scrutiny, audio event localisation in crowds, or overall improvement of road safety. The following ten use cases collectively demonstrate the versatility of the MARVEL framework in addressing various real-world scenarios relevant to smart cities.

- GRN1 – Safer Roads
- GRN2 – Road user behaviour
- GRN3 – Traffic Conditions and Anomalous Events
- GRN4 – Junction Traffic Trajectory Collection
- MT1 – Monitoring of crowded areas
- MT2 – Detecting criminal/anti-social behaviours
- MT3 – Monitoring of parking places
- MT4 – Analysis of a specific area
- UNS1 – Drone experiment
- UNS2 – Localising audio events in crowds

A comprehensive overview of all the above use cases is available, in deliverable D5.6 – "MARVEL Integrated Framework – Final Version" [9], which provides a detailed description of all the integrated components coming from the seven different subsystems of the MARVEL framework:

- Sensing and perception Subsystem
- Security, Privacy and data protection Subsystem
- Data Management and distribution Subsystem
- Audio, visual and multimodal AI Subsystem
- Optimised E2F2C processing and deployment Subsystem
- E2F2C Infrastructure
- User interactions and decision-making toolkit.

The PSNC cloud infrastructure is integral to the MARVEL project, serving as the foundation for deploying the software stack and storing project data. The provided resources facilitate the efficient deployment of components running in the cloud layer of the E2F2C framework.

Table 10 presents the components of MARVEL deployed on the cloud infrastructure, depending on the specified use case. These components are categorised within the subsystems of the MARVEL framework to which they belong. The last ten columns refer to the ten mentioned use cases. To better understand the role of the cloud infrastructure, let us briefly discuss the individual components deployed within it.

Table 10: Components of the MARVEL framework deployed on the cloud infrastructure depending on the use case

MARVEL subsystem	Component name	GRN				MT				UNS	
		1	2	3	4	1	2	3	4	1	2
Security, privacy and data protection subsystem	EdgeSec VPN	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Data management and distribution subsystem	DatAna	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	DFB	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	HDD	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Audio, visual and multimodal AI	AVAD			✓			✓	✓	✓		
	SED				✓		✓	✓	✓		
	AVCC				✓						
	ViAD					✓					
	VCC					✓					
	AT						✓	✓			

Optimised E2F2C processing and deployment	MARVdash	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
User interactions and decision-making toolkit	SmartViz	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	MARVEL Data Corpus	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

EdgeSec VPN, a vital component of the security, privacy, and data protection subsystem, ensures a secure peer-to-peer VPN network within the E2F2C architecture. Its primary function is to encrypt network communication, safeguard against unauthorised access, and integrate hosts into the Kubernetes cluster. This facilitates seamless accessibility through MARVdash for automated deployment within the MARVEL E2F2C framework.

The Data Management and Distribution Subsystem components deployed on the cloud infrastructure encompass DatAna, DFB, and HDD. Their primary goal is to proficiently handle large volumes of data sourced from diverse origins and ensure its appropriate distribution. DatAna facilitates seamless data handling and transformation across various data sources, ensuring user-friendly ingestion and movement. DFB plays a crucial role in securing data transfers between components and managing permanent storage, addressing challenges related to data volume and heterogeneity. HDD specialises in optimising Apache Kafka data topic partitioning within MARVEL's distributed data delivery, significantly contributing to enhanced Kafka cluster performance. Like EdgeSec VPN, these components are applicable in all use cases. Further details about this component group will be provided in Section 4.

Another subsystem of the MARVEL framework is audio, visual, and multimodal AI, which includes six different components (AVAD, SED, AVCC, ViAD, VCC, AT) deployed on cloud infrastructure in at least one use case. The Audio-Visual Anomaly Detection (AVAD) component detects situations that are considered as novel in visual scenes based on sequential frames and audio. It produces scores indicating the likelihood of each frame containing a novelty and can provide anomaly heatmaps. The Sound Event Detection (SED) component is responsible for identifying sound events and their temporal location in the audio signal, which is crucial for understanding smart city environments and detecting actions based on audio signals in various MARVEL use cases. The Audio-Visual Crowd Counting (AVCC) component estimates the number of people in a scene using an image and a 1-second audio snippet. AVCC, trained in a supervised manner, requires fully labelled data and can run on systems with or without a GPU. The Visual Anomaly Detection (ViAD) component identifies novel situations in video frames, producing a label declaring the sequence as normal or anomalous. Trained for a specific camera, ViAD recognises various events as anomalies based on provided data. The Visual Crowd Counting (VCC) component estimates the number of people in a scene from a single image, producing a heatmap. Trained in a supervised manner, VCC requires fully labelled data and can run on systems with or without a GPU. The Audio Tagging (AT) component in MARVEL recognises sound source activity in predefined audio segments. The information can be used independently or complementarily to gain a deeper understanding of the scene.

MARVdash, a vital component of the Optimised E2F2C processing and deployment subsystem is deployed on the cloud infrastructure across all use cases. Operating as a Kubernetes dashboard, MARVdash streamlines user interaction and service deployment by integrating seamlessly into the MARVEL Kubernetes cluster. With an intuitive interface and efficient deployment mechanisms, it leverages Kubernetes to simplify the configuration and initiation of services. Serving as a central orchestrator, MARVdash facilitates the deployment of data management platforms, AI components, and software across different layers of the E2F2C framework.

The last group of components (SmartViz, MARVEL Data Corpus) deployed on the cloud infrastructure is the user interactions and decision-making toolkit. SmartViz, serving as the User Interface (UI) for the Decision-Making Toolkit (DMT) within the MARVEL framework, plays a pivotal role in facilitating exploratory data analysis. Through interactive visualisations, features, and widgets, SmartViz provides crucial insights, supporting end-users in areas such as urban planning and safety, aligning with the broader objectives of the MARVEL pilot initiatives. MARVEL Data Corpus is designed to supply Machine Learning datasets to scientific and industrial communities, sourcing anonymised and annotated video and audio data from MARVEL pilots. Data Corpus serves a triple purpose: storing, facilitating internal AI training, and making datasets accessible to external users through programmatic and graphical interfaces, with an expected growth in data volume as the project advances.

To ensure the optimal operation of the MARVEL system, a meticulous analysis of individual component requirements across various layers was essential. This involved estimating computational power, memory, disk, and network performance for each component within the R2 deployment context. MARVEL partners focused on estimating component requirements, while the Consortium documented potential infrastructure specifications, aligning both aspects during the initial system design to meet end-user needs and allocate available resources effectively.

In the cloud infrastructure, where multiple components and instances were intended to be deployed, the requirements were aggregated per use case. This strategic approach ensured the allocation of necessary resources for Virtual Machines (VMs) hosting the MARVEL Kubernetes cluster and deployed services. The initial plan was to assign specific MARVEL services to specific cloud VMs, ensuring an optimal resource access. However, it was later determined that the built-in automated load balancing mechanisms of Kubernetes adeptly handle service allocation to available VMs without encountering any issues. Consequently, the manually devised allocation plans were retained as a backup plan.

Table 11 provides an overview of the individual VMs from the cloud layer. In the initial release of the MARVEL framework (M18), all VMs were situated in the DCW region. However, in the final version of the framework, resources from both DCW and BST regions are used. The reason behind this change is the fact that the BST region has better storage capabilities and therefore is intended for the MARVEL Data Corpus. The five first listed machines in the DCW region (marvel, worker1, worker2, worker3, worker4) are Kubernetes cluster nodes managed by MARVdash. The other machines serve diverse project requirements, such as performance monitoring of the MARVEL infrastructure based on the Zabbix software (marvel-zabbix), hosting GitLab repository server (gitlab), or facilitating Data Corpus augmentation (CorpusAugmentation). Summarising both regions, the MARVEL project consumes 342 VCPUs and 450GB of RAM.

Table 11: Configuration of VMs delivered to the project

Region	Instance Name	VCPUs	RAM
DCW	marvel	32	64
	worker1	32	64
	worker2	40	40
	worker3	40	40
	worker4	40	40
	marvel-zabbix	4	8
	gitlab	4	8
	edgesec_vpn_supernode	2	4
	edge_sec_vpn_gw	2	4
	acme-dns	2	4
	CorpusAugmentation	32	32
	DataCorpusNew	16	32
	DataCorpusCluster1	8	16
	DataCorpusCluster2	8	16
BST	DataCorpusClusterBST1	16	16
	DataCorpusClusterBST2	16	16
	DataCorpusClusterBST3	16	16
	DataCorpusClusterBST4	16	16
	DataCorpusClusterBST5	16	16
Summary		**342**	**452**

3.4 Access to GPU from cloud computing system

The MARVEL project uses two types of computing resources in PSNC. These are cloud and HPC computing resources. CPUs are available within the cloud environment, while within HPC, the MARVEL project has access to CPUs and GPUs. At the initial stage of the MARVEL project, GPU resources were available via the queuing system (standard interface for access to HPC resources – Figure 13). As the project was evolving, the initial solution emerged to be insufficient as the integration of typical HPC batch workflow proved to be difficult and of little use. Using HPC service imposes that all compute jobs are assigned to specific hardware only for strictly defined periods (in our case, max 7 days), and the applications must run under a specific version or distribution of Linux. In addition, while applications running on an HPC server may access the Internet, there is no way of exposing such applications to be accessed from outside the HPC cluster. Since the majority of the MARVEL software stack is running on the PSNC cloud, an attempt to connect the GPU server to the cloud was made to enable more effective use of resources in a shared environment. The MARVEL project uses Kubernetes software, and it was desired that both types of CPU and GPU computing resources to be available in this environment. Therefore, one of the GPU servers was delegated specifically for the MARVEL project. Since this server is part of the HPC cluster, it was not possible to give access to the machine directly. It was decided to run a virtual machine on a dedicated server and configure the network in such a way that communication between cloud resources (virtualisation) and HPC resources (virtual machine with exposed GPU cards) is possible. The server that hosts the VM with GPUs is marked as "reserved" from an HPC management point of view, so no HPC jobs will disturb MARVEL computations. The biggest challenge in this setup was providing seamless network communication between MARVEL VMs running on OpenStack cloud and VMs running on the HPC cluster, as these two environments are independent from a network perspective. The connectivity was implemented by exposing internal HPC VLAN to cloud network nodes, which enabled us to create a special router in the cloud that routes traffic from and to the MARVEL project in the cloud to the machine in the HPC network. Currently, this is a static setup, however, a solution enabling dynamic resource allocation from the cluster to the cloud environment via a queuing system has been tested. An illustrative operating diagram is shown in Figure 13.

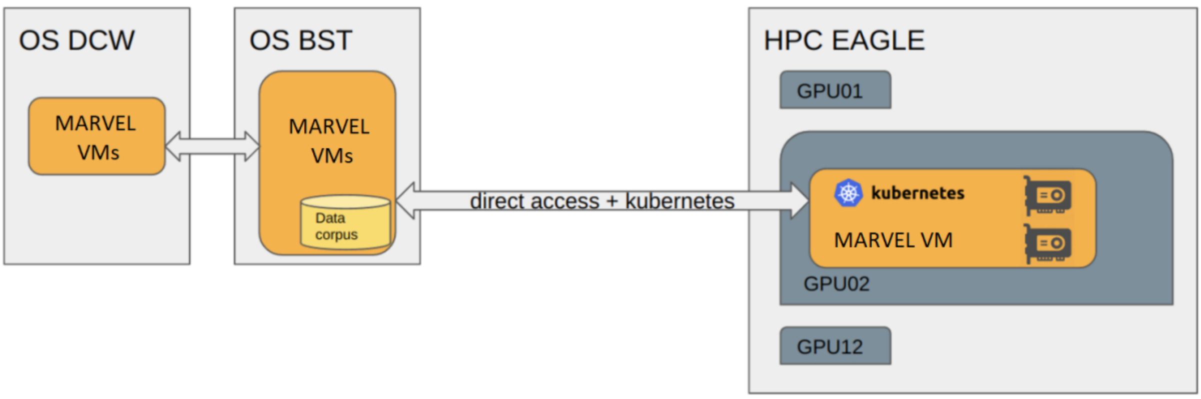

Figure 13: Direct connection with selected GPU node + Kubernetes extended to GPU node

Table 12 presents the specifications of the provided VM equipped with GPUs.

Table 12: Specifications of VM equipped with GPUs

GPU	VCPUs	RAM
2x NVIDIA V100	36	64 GB

The final solution of the MARVEL has the following advantages:

- All VMs that build up the MARVEL platform can have a homogeneous software stack and can be configured in the same way,
- Ability to directly access data to and from any MARVEL container for GPU-based applications,
- MARVEL Kubernetes connects all compute resources, including GPU, so that tasks can be submitted from MARVdash directly or from any other MARVEL container via REST or CLI,
- Despite semi-static configuration, it is easy to add new GPU resources.

3.5 Monitoring tools

Cloud infrastructure (called "LabITaaS") is being monitored on both physical (servers, switches, state of links, etc.) and logical levels. From MARVEL's perspective, more important is logical level monitoring on which all basic components are tested and checked for proper behaviour. The monitoring and SLA levels are related to multiple tests that evaluate both physical aspects of the infrastructure and available functionality. All metrics are gathered and calculated using Zabbix software.

Labitaas	OK	0.5974	99.4026 / 99.9900
cinder	OK	0.0000	100.0000 / 99.9000
horizon	OK	0.0069	99.9931 / 99.9000
keystone	OK	0.0000	100.0000 / 99.9000
neutron	OK	0.5412	99.4588 / 99.9000
nova	OK	0.0023	99.9977 / 99.9000
S3 RADOS GW	OK		
Subnet manager	OK	0.0000	100.0000 / 99.9900

Figure 14: Example of measurement results from the LabITaaS

Figure 14 presents an example of measurement results from the cloud infrastructure. One can see that there are several columns listed. The first column holds the name of the component followed by the current state. The next column shows unavailability levels both in graphical and numerical format. The last column presents the current and threshold values of the SLA for a given module. Some of the modules are aggregates of multiple lower-level metrics, but in most cases, these reflect the redundant nature of each service.

The tested modules of LabITaaS are:

- LabITaaS: Aggregates values for the cloud infrastructure calculated based on all components. This aggregate considers the whole cloud as non-100 % functional if any components are unavailable.
- cinder: Blocks storage service for all virtual machines and volumes used in the cloud environment. Unavailability of this service may affect running VMs (virtual machines) as it may (but does not have to) indicate that VM lost access to the underlying storage.
- horizon: Web interface used for human interaction, such as creation or manipulation of virtualised resources. The unavailability of this service does not affect the running VMs but prevents the possibility of manipulating the resources using the web interface and does not affect any already configured services.
- keystone: Basic identity service used by OpenStack components. Unavailability mostly affects the ability of components to communicate with each other but does not imply malfunction of VMs.
- neutron: Network virtualisation service. Unavailability affects the possibility of changing or creating networking components in the cloud and may also mean some problems with accessing some or all virtual machines from the external networks.
- nova: Service responsible for the creation and manipulation of the VMs. Unavailability prevents the creation of new VMs.

- S3 RADOS GW: Access to object storage via an S3 gateway. This represents the availability of the S3 storage. It does not directly affect VMs but may mean a lack of data access for services that use this storage.

- subnet manager – Management service for Infiniband network. PSNC cloud relies on high speed, low-latency Infiniband network for storage and networking. Malfunction of this service means severe problems with all other services and VMs running in the environment.

From the MARVEL perspective, the above statistics are quite general in nature, as they concern the entire PSNC cloud infrastructure and not only the MARVEL project. For this reason, PSNC and FORTH have developed an additional monitoring system based on Zabbix software. It is fully dedicated to the project and allows monitoring the performance of the individual components deployed across different layers of the E2F2C framework. While the Zabbix central server is located in the DCW region of the cloud infrastructure, Zabbix agents are deployed on diverse devices, such as Raspberry Pi, VMs, and physical machines distributed across cloud, fog, and edge environments. These agents collect crucial performance data securely transmitted to the central Zabbix server through EdgeSec VPN connections. The system includes comprehensive dashboards organised into multiple pages, providing a holistic view of the infrastructure's metrics such as Uptime, Storage utilisation, Network performance, CPU usage, RAM statistics, GPU insights, and detailed Disk utilisation data. This architecture enables efficient monitoring, analysis, and optimisation of various devices across cloud, fog, and edge environments. Figure 15 presents an example panel of the developed tool. For more information on this tool, please refer to the deliverables D5.5 and D5.6.

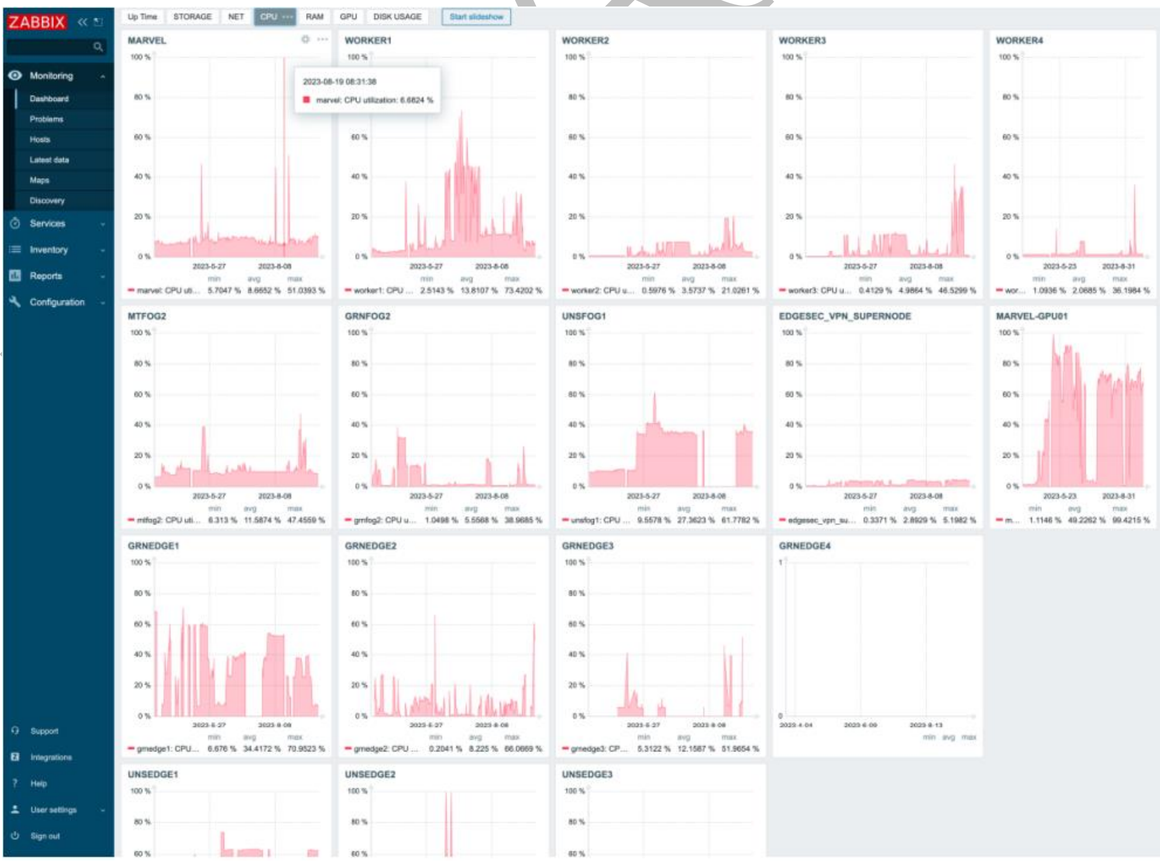

Figure 15: CPU page of MARVEL infrastructure monitoring tool

4 Storage and Data analytics system

The escalating demand for processing vast volumes of data has propelled the increasing popularity of High Performance Data Analytics (HPDA). HPDA seamlessly integrates High Performance Computing (HPC) or other dedicated infrastructures with Big Data, enabling the efficient analysis of extensive operational and static data alongside multiple streaming data sources. This synergistic approach proves highly advantageous when the requirements for volume, velocity, and variety exceed the capabilities of individual domain-specific state-of-the-art solutions. The execution of computations is facilitated through the utilisation of frameworks designed to scale the implementation both in terms of data and the underlying infrastructure. A myriad of solutions tailored to the specific problem at hand and the preferred programming language are readily available in the market.

Currently, the PSNC mass storage infrastructure consists of several types of systems, including mid-size disk arrays, high performance disk arrays, software-defined disk servers, as well as specialised systems, including an efficient file server and SSD arrays. The disk system mainly consists of NetApp E5600 series storage arrays, typically with 120 x 4 TB each. They are connected to Brocade 16Gbit/s switches (96 ports/switches). HPC arrays are DDN SFA4k12x with 420 x 4 TB drives per array. Attached to them there are 4 OSS Luster servers, as well as a cluster of Luster MDS servers. File servers include EMC Isilon 5x x410 with 10Gbit Ethernet interfaces. SSD arrays include EMC XtremeIO and NetApp EF500 arrays with FC 8Gbit/s and FC 16Gbit/s/IB QDR interfaces. PSNC also supports a cluster of disk servers that run OpenStack Swift with Swift and S3 interfaces and offer RADOS and S3 as well as RBD standards. PSNC offers 42PB of online storage and data management infrastructure, providing direct support for the scientific communities in Poland as well as in Europe. Moreover, PSNC is a leader of a Polish initiative named National Data Storage - KMD (Krajowy Magazyn Danych). By the end of 2024, the infrastructure created by the KMD members will increase the capacity of the data storage infrastructure by 200 petabytes and the capacity of the tape space by 180 petabytes, and the DataLake model used in its construction will ensure its high flexibility. In addition, as a result of the MARVEL project, by 2024, cost-effective services for basic data storage and management processes, as well as their processing and analytics will be provided.

Compared to D5.3, this section includes relevant updates reflecting the current status at the end of the project. It provides information on the data analytics solutions implemented in the MARVEL framework. Specifically, it describes the Data Management and distribution subsystem of the MARVEL framework, briefly explaining its role in data aggregation in the Data Corpus. Furthermore, it discusses the implementation of Data Corpus and the provided computing and storage infrastructure.

4.1 Data management and distribution subsystem of MARVEL framework

In the MARVEL project, the subsystem responsible for data management and distribution comprises the DFB, StreamHandler, DatAna, and HDD components. Its primary objective is to effectively manage extensive volumes of data originating from diverse sources and ensure their proper distribution. All the components of this subsystem are deployed in the cloud layer, except for StreamHandler, which is deployed in the Fog layer and interacts with the others.

This section presents the most relevant information of the listed components. Further information can also be found in D2.4 [3], which is directly devoted to the Management and Distribution toolkit.

DatAna functions as a versatile platform, seamlessly handling data across diverse computing layers. With its user-friendly graphical interface and minimal coding requirements, DatAna simplifies the definition of data flows, accommodating static data such as files and databases, as well as dynamic data streams like MQTT and Apache Kafka topics. Deployed within the Apache NiFi ecosystem [16], DatAna ensures scalability, high throughput, and runtime adaptability. It utilises NiFi Registry for version control and communicates with ML inference models through MQTT brokers. This pivotal component centrally processes and transforms real-time data outputs from inference pipelines, ensuring traceability through attributes like absolute timestamps and contributing to MARVEL's data models. In the R2 release, DatAna's functionality extends beyond data processing, enabling future enhancements such as data enrichment, threshold examination, and seamless integration of control signals from different components. The DatAna infrastructure, illustrated in Figure 16, spans cloud, fog, and edge layers, presenting a comprehensive architecture that facilitates efficient communication and streamlines data flow within MARVEL's evolving data ecosystem.

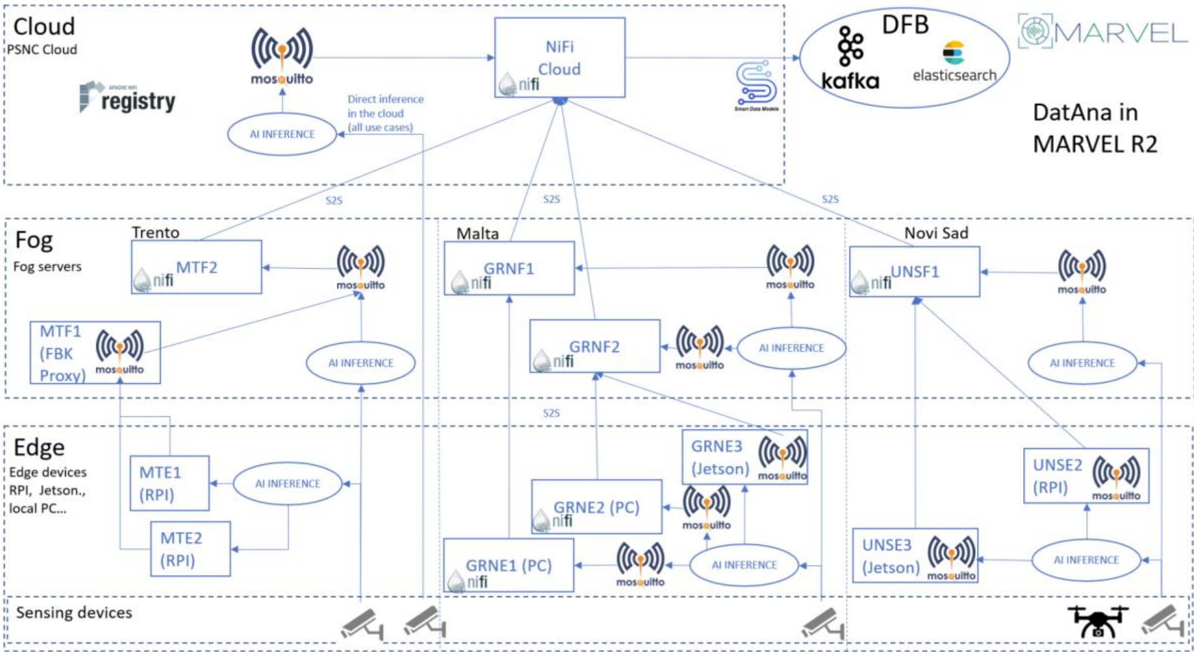

Figure 16: DatAna topologies

Data Fusion Bus (DFB) serves as a customisable solution facilitating secure, efficient transfer of heterogeneous data across connected components and permanent storage within MARVEL's framework. Comprising dockerised, open-source elements, the DFB's robust architectural design adeptly addresses challenges related to diverse data structures and organisational threats, providing a secure, common interface for seamless data access. Key modules, including Apache Kafka and Elasticsearch, empower the DFB with efficient data handling, scalability, and cloud-native adaptability. In the MARVEL context, the DFB plays a crucial role in both the Data Management Platform (DMP) and the inference pipeline. It efficiently aggregates and relays streamed inference results from MARVEL AI components to SmartViz and other tools, securely storing them in Elasticsearch. The DFB also offers a user-friendly graphical interface for visualisation through Kibana and Grafana, significantly enhancing monitoring capabilities. Deployed in the cloud, the DFB's specific implementation, as detailed in Figure 17, caters to diverse R2 use cases. Iterative enhancements continue to address SDM-compliant configuration, real-time fusion of inference results, and seamless integration with SmartViz for verification and visualisation. The central role of the DFB in aggregating and relaying inference results underscores its critical significance in shaping MARVEL's resilient and efficient architecture.

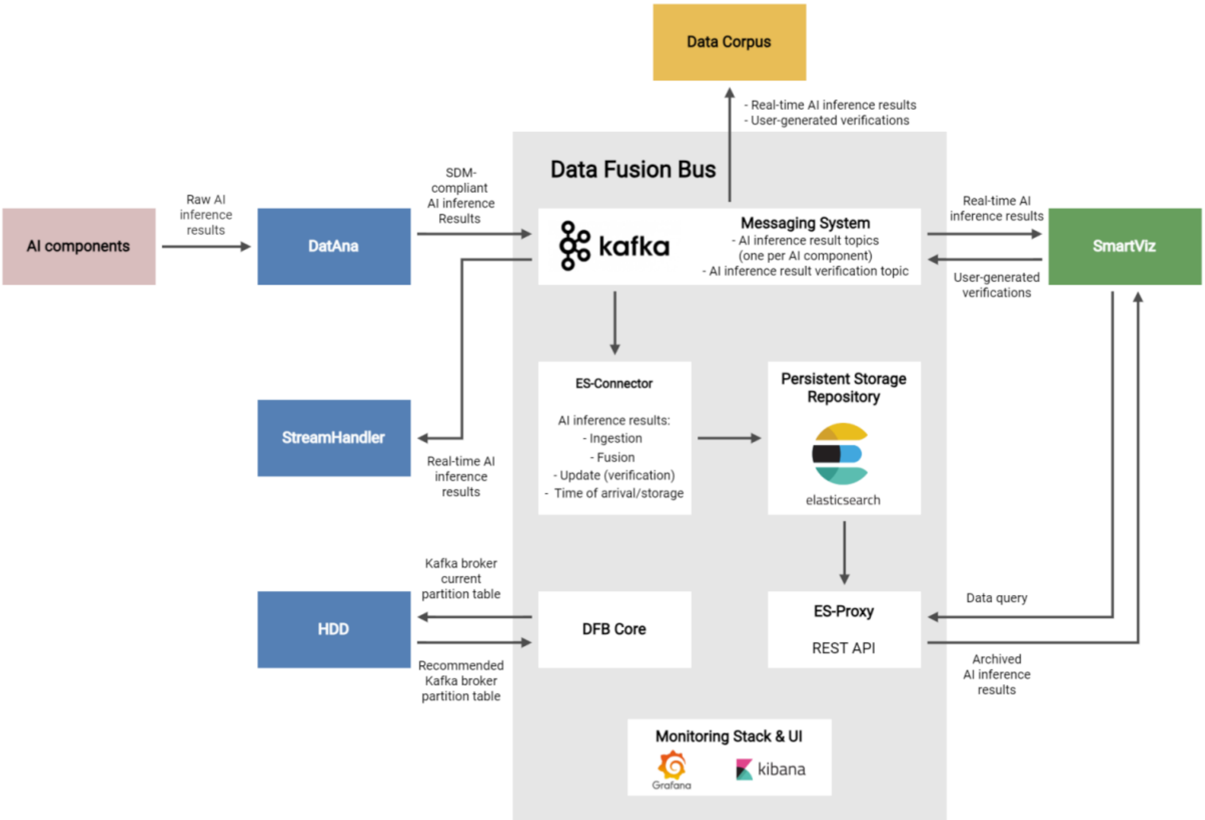

Figure 17: DFB internal architecture and interactions

INTRA's StreamHandler Platform within MARVEL's R2 exemplifies a Microservice Architecture, featuring four distinct services showcased in Figure 18. This modular approach facilitates scalability, allowing seamless replication of instances to address potential bottlenecks. Notably, StreamHandler integrates with the Data Fusion Bus (DFB) to introduce event-based audio-video file production. The platform's internal operations leverage Java, SpringBoot, MinIO, Docker, REST API, and Kafka. Figure 18 outlines each service's role: Service 1 initiates segmentation, Service 2 provides videos to SmartViz through REST API calls, Service 3 constructs audio-video files based on events, and Service 4 functions as the storage component. This decoupled architecture enhances flexibility and stability, with storage serving as the primary interaction point. StreamHandler's pivotal role remains focused on ensuring data privacy by deploying in the Fog layer, close to audio-video sources. Operating individually in each pilot server, it supports the seamless processing of sensitive data, underscoring its critical contribution to efficient communication, data flow, and real-time processing within MARVEL's evolving framework.

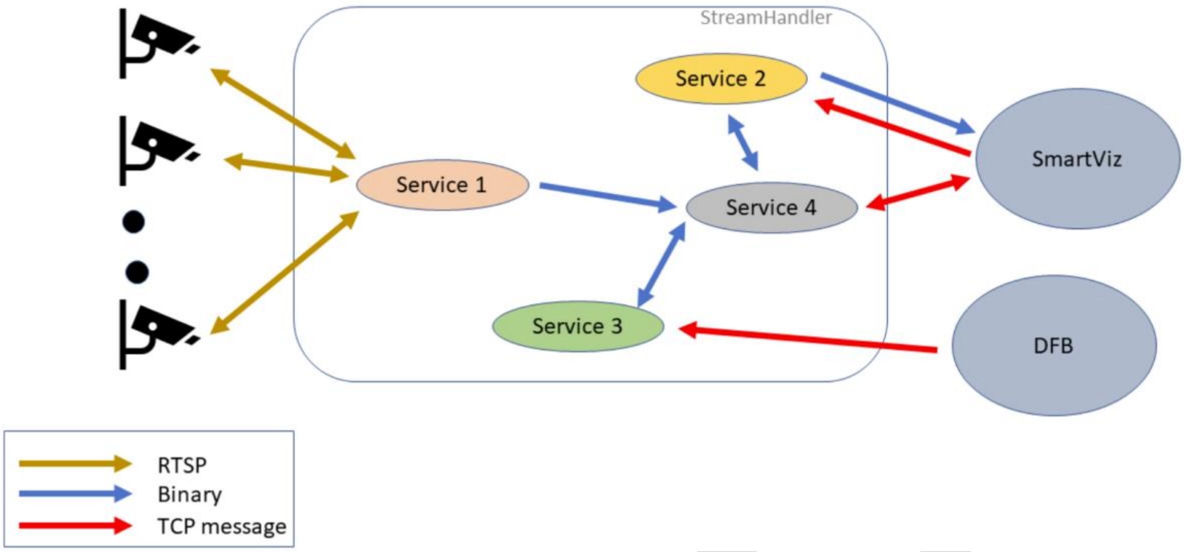

Figure 18: StreamHandler implementation

HDD is a component of the MARVEL distributed data delivery and access algorithmic schemes, contributing significantly to the optimisation of data topic partitioning within the Apache Kafka framework. Tailored to complement MARVEL's data distribution technologies, HDD addresses the challenge of fine-tuning Kafka cluster performance, which remains an open research problem. Balancing contrasting forces that influence partition decisions, HDD offers recommendations by considering factors such as partition density and its impact on metadata operations and replication latency. Despite Apache Kafka's inherent optimisations, HDD's main contribution lies in determining the optimal number of partitions per topic, enhancing parallelism in message consumption while considering resource overhead. The component's internal operations involve modelling Kafka topic partitioning, formulating an optimisation problem, and employing two algorithms respecting application constraints. HDD is implemented in the Octave SW suite32, chosen for its open-source nature in contrast to commercial alternatives and its computational power for mathematical programming applications.

4.2 MARVEL Data Corpus

The MARVEL Data Corpus serves a pivotal role in the project by offering anonymised and annotated video and audio datasets from MARVEL pilots to both scientific and industrial communities. This section presents the general overview of the Data Corpus, and details provided computing and storage infrastructure. For more information on the Data Corpus, please refer to D2.5, titled “Corpus-as-a-Service specifications and business continuity” (due M36.

The MARVEL Data Corpus utilises various technologies, including Hadoop Distributed File System (HDFS) for data storage across low-cost machines, ensuring resilience to hardware failures. HBase, an open-source, non-relational, distributed database that manages data relationships in a non-schema less approach, while Zookeeper serves as the administrative application for HBase. Ambari's web interface facilitates easy management of Hadoop services through webpage and RESTful APIs.

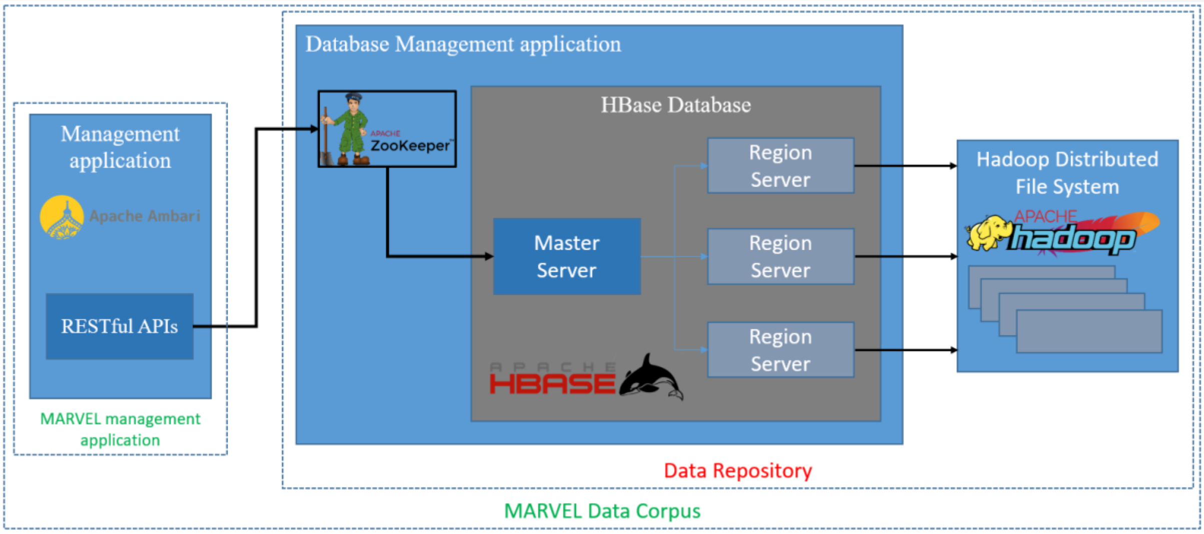

Figure 19: MARVEL multimodal Data Corpus-as-a-Service for smart cities

The Data Corpus plays a central role in aggregating diverse data sources, such as the MinIO [17] repository accessed through StreamHandler instances at Fog nodes across pilot locations. These instances gather anonymised AV data from AudioAnony and VideoAnony at Edge and Fog nodes, storing segmented AV data streams in their MinIO repositories. AV files are then directly retrieved from these repositories by the Data Corpus. Furthermore, the Data Corpus enhances its repository by collecting inference results from various MARVEL use cases. By subscribing to DFB Kafka topics, it receives real-time updates from DatAna. Additionally, SmartViz provides crucial information on human-verified inference results, serving as a valuable "ground truth" for labelled datasets aligned with MARVEL AI components' training requirements.

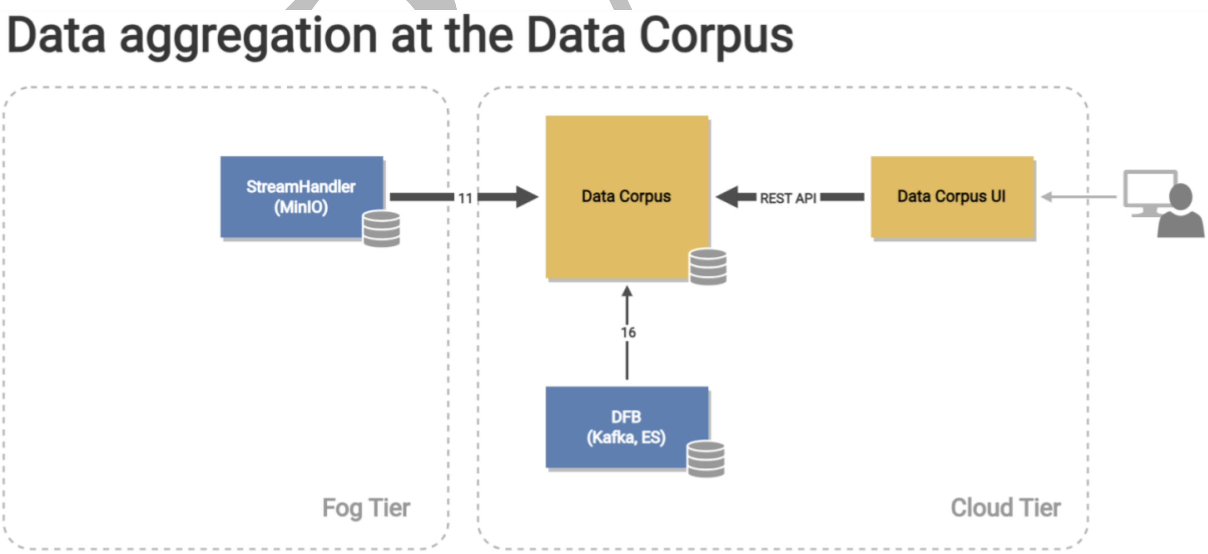

Figure 20: Data aggregation at the MARVEL Data Corpus

The initial goal of MARVEL was to accumulate over 3.3PBs of data in the Data Corpus. Throughout the project, extensive discussions with partners took place, exploring various options to ensure efficient storage infrastructure. One considered option was creating a dedicated environment linked to the Hadoop HBase cluster. However, due to data accumulation challenges faced by pilots, achieving the goal of gathering over 3.3PBs proved unattainable. Consequently, a decision was made to adhere to an approach based on virtualised infrastructure with an attached CEPH storage backend. This approach facilitated a smooth increase in available resources, aligning with the growing needs of the MARVEL project.

Table 13 illustrates the current configuration of VMs associated with the Data Corpus. In summary, the machines connected to the Data Corpus currently have the capability to store nearly 1.8PBs of data. The PSNC has enabled further expansion of the virtual environment with additional storage in the BST region in case the pilots accumulate more data. As per the project agreements outlined in KPI-O5-E4-1, PSNC will maintain the Data Corpus until the end of 2024.

Table 13: Configuration of VMs related to the MARVEL Data Corpus

Region	Instance Name	VCPUs	RAM	Storage for OS [GiB]	Storage for data [GiB]
DCW	CorpusAugmentation	32	32	100	10 000
	DataCorpusNew	16	32	100	21 480
	DataCorpusCluster1	8	16	100	10 000
	DataCorpusCluster2	8	16	100	250 000
BST	DataCorpusClusterBST1	16	16	100	300 000
	DataCorpusClusterBST2	16	16	100	300 000
	DataCorpusClusterBST3	16	16	100	300 000
	DataCorpusClusterBST4	16	16	100	300 000
	DataCorpusClusterBST5	16	16	100	290 000
Summary		**144**	**176**	**1 781 480**	

5 Conclusions

This document is the official output of the tasks T5.1 and T5.2 within the WP5 of the MARVEL project. The work that is listed had been performed during the entire project life cycle, with particular attention paid to the hardware and software infrastructure, including available services that can be used in favour of the project's purposes.

The document provides a brief outline of the MARVEL project in respect of resource demands and highlights various aspects of the available infrastructure. The first part described the High Performance Computing System from three perspectives: hardware, software, and user services and support. It also included relevant information regarding granting access to HPC resources and described infrastructure monitoring tools. The next part focused on the cloud computing system, covering overall hardware and system management aspects. Particular emphasis was given to presenting the role of cloud computing system which served as the foundation for deploying the software stack and storing project data of the MARVEL project. The provided resources facilitated the efficient deployment of components running in the cloud layer of the E2F2C framework. Finally, this document also reported on the infrastructure delivered for the Data Corpus, designed to provide a substantial volume of anonymised and annotated audio-visual datasets from MARVEL pilots to scientific and industrial communities. Based on the project agreements, PSNC will maintain the infrastructure for the Data Corpus for one year after the end of the project.

Compared to the initial version, which was D5.3, this document included relevant updates related to hardware and software reflecting the current status at the end of the project. Additionally, the content had been expanded to include sections about achievements from the second part of the project. Specifically, the document described implemented solutions for extending the Kubernetes cluster with GPU-equipped nodes at the cloud layer. Furthermore, it provided information related to HPC and cloud infrastructure monitoring and outlined contributions to developing a tool that facilitates monitoring the entire E2F2C framework's infrastructure. Finally, this document reported all the information related to the cloud infrastructure and its technical decisions that led to the realisation of the MARVEL. As the final setup used by MARVEL is reported, this deliverable may be used as a starting point for other scientific or industrial frameworks.

6 Bibliography

[1] "D1.2: MARVEL's experimental protocol," MARVEL Project, 2021. Confidential.
[2] "D1.3: Architecture definition for MARVEL framework," MARVEL Project, 2021. https://doi.org/10.5281/zenodo.5463897
[3] "D2.4 - Management and distribution Toolkit – final version," MARVEL Project, 2023. https://doi.org/10.5281/zenodo.8147109
[4] "D5.1: MARVEL Minimum Viable Product," MARVEL Project, 2021. https://doi.org/10.5281/zenodo.5833310
[5] "D5.2: Technical evaluation and progress against benchmarks – initial version," MARVEL Project, 2022. https://doi.org/10.5281/zenodo.6322699
[6] "D5.3: HPC infrastructure and resource management for audio-visual data analytics – initial version," MARVEL Project, 2022. https://doi.org/10.5281/zenodo.6821304
[7] "D5.4: MARVEL Integrated framework – initial version," MARVEL Project, 2022. Confidential.
[8] "D5.5: Technical evaluation and progress against benchmarks – final version," MARVEL Project, 2023. https://doi.org/10.5281/zenodo.10438311
[9] "D5.6: MARVEL Integrated framework – final version," MARVEL Project, 2023. https://doi.org/10.5281/zenodo.8315386
[10] "SLURM workload Manager," 2023. [Online]. Available: https://slurm.schedmd.com/
[11] "PSNC Eagle software," 2023. [Online]. Available: https://wiki.man.poznan.pl/hpc/index.php?title=Oprogramowanie.
[12] "Intel oneAPI DPC++/C++ Compiler," [Online]. Available: https://www.intel.com/content/www/us/en/developer/tools/oneapi/dpc-compiler.html#gs.3n9rbk
[13] "PSNC HPC portal," 2023. [Online]. Available: https://pcss.plcloud.pl
[14] "Web-based interface for submitting jobs to the HPC cluster," 2023. [Online]. Available: https://qcg.pcss.plcloud.pl/
[15] "Web-based SSH client for submitting jobs to the HPC cluster," 2023. [Online]. Available: https://ssh-ui.pcss.plcloud.pl/
[16] "Apache NiFi Ecosystem", 2023. [Online]. Available: https://nifi.apache.org/
[17] "MinIO", 2023. [Online]. Available: https://min.io/