D5.1 HiDALGO System Environment

Document Identification			
Status	Final	**Due Date**	28/02/2019
Version	1.0	**Submission Date**	28/02/2019

Related WP	WP5	**Document Reference**	D5.1
Related Deliverable(s)	D5.2, D5.3, D5.6, D5.7	**Dissemination Level (*)**	Public
Lead Participant	USTUTT	**Lead Author**	Abhishek Abhishek (HLRS)
Contributors	Sergiy Gogolenko (HLRS), Michael Gienger (HLRS), Norbert Meyer (PSNC), Marcin Lawenda (PSNC), Stephan Siemen (ECMWF)	**Reviewers**	Ákos Kovács (SZE)
			Dimitris Tsoumakos (ICCS)

Keywords:
High Performance Computing (HPC), High Performance Data Analytics (HPDA), Cluster Management, Services, Infrastructure, Operation, Services, Getting access

This document is issued within the frame and for the purpose of the HiDALGO project. This project has received funding from the European Union's Horizon2020 Framework Programme under Grant Agreement No. 824115. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission.

(*) Dissemination level: **PU**: Public, fully open, e.g. web; **CO:** Confidential, restricted under conditions set out in Model Grant Agreement; **CI:** Classified, **Int =** Internal Working Document, information as referred to in Commission Decision 2001/844/EC.

1 Document Information

List of Contributors	
Name	Partner
Michael Gienger	HLRS
Sergiy Gogolenko	HLRS
Norbert Meyer	PSNC
Marcin Lawenda	PSNC
Stephan Siemen	ECMWF

Document History			
Version	Date	Change editors	Changes
0.1	11/02/2019	Abhishek Abhishek	Template
0.2	13/02/2019	Abhishek Abhishek	Initial contributions
0.3	15/02/2019	Abhishek Abhishek, Marcin Lawenda, Stephan Siemen	Inclusion of contributions
0.4	20/02/2019	Abhishek Abhishek	Pre-final version for the internal review
0.5	25/02/2019	Abhishek Abhishek	Addressed the comments of the internal review
1.0	27/02/2019	Abhishek Abhishek	FINAL VERSION TO BE SUBMITTED

Quality Control		
Role	Who (Partner short name)	Approval Date
Deliverable leader	Abhishek Abhishek (HLRS)	27/02/2019
Quality manager	Marcin Lawenda (PSNC)	28/02/2019
Project Coordinator	Francisco Javier Nieto de Santos (ATOS)	28/02/2019

Table of Contents

List of Tables

List of Figures

List of Acronyms

Abbreviation / acronym	Description
AI	Artificial Intelligence
CAE	Computer Aided Engineering
CFD	Computational Fluid Dynamics
CPU	Central Processing Unit
D5.1	Deliverable number 1 belonging to WP 5
DHS	Data Handling System
DIAS	Copernicus Data and Information Access Services
DoA	Description of Action
EC	European Commission
ECMWF	European Centre for Medium-range Weather Forecasts
ENS	Members' Ensemble
Gbps	Gigabit per second
GB	Gigabyte
GB/s	Gigabyte per second
GCC	GNU Compiler Collection
GPU	Graphics Processing Unit
GSS	Global Systems Science
HLRS	High Performance Computing Center Stuttgart
HPC	High Performance Computing
HPDA	High Performance Data Analytics
HPSS	High Speed Storage System
HSM	Hierarchical Storage Management System
HW	Hardware
JSC	Juelich Supercomputing Centre
LRZ	Leibniz Supercomputing Centre
KM	Kilometre
MAN	Metropolitan Area Network
MARS	Meteorological Archival and Retrieval System

Abbreviation / acronym	Description
MPI	Message Passing Interface
MPP	Massive Parallel Processing
OTRS	Open Ticket Request System
PAPI	Performance Application Programming Interface
PATC	PRACE Advanced Training Center
PCI	Peripheral Component Interconnect
Pflops/s	Petaflops per second
PRACE	Partnership for Advanced Computing in Europe
PSNC	Poznan Supercomputing and Networking Centre
PB	Petabyte
RAM	Random Access Memory
SSH	Secure Shell
SW	Software
TB	Terabyte
TDP	Thermal Design Power
VM	Virtual Machine
WP	Work Package

Executive Summary

This deliverable highlights the infrastructure available for the HiDALGO project. The document describes the hardware, software and services offered by the three supercomputing centres HLRS, PSNC and ECMWF. The main focus is given on using the already available resources including the process of accessing these resources.

The HPC resources require specialized expertise to use them efficiently and thus, the deliverable provides the mechanisms for accessing and using these resources at HLRS, PSNC and ECMWF.

Furthermore, to ensure seamless access to compute resources within the HiDALGO project, it is highly imperative that we facilitate the cooperation with external initiatives like PRACE to enhance the infrastructure. Consequently, this deliverable also describes the process of accessing the PRACE Research Infrastructure in detail.

1 Introduction

The goal of HiDALGO is to establish a systematic cooperation between the communities from High Performance Computing (HPC), High Performance Data Analytics (HPDA) and experts from a broad range of disciplines to handle Global Challenges in areas such as ecology, technology, sociology and economy. Due to the complexity of Global Challenges, such as air pollution in metropolitan cities or (at present), the refugee crisis in Europe, and their dependency on interdisciplinary collaborations, experts from multiple domains are required to work together for data gathering, modelling, simulation, analysis and visualization. HiDALGO tackles these Global Challenges by encapsulating them in an HPC and HPDA environment and allowing the analysis of these problems by enabling precise simulations, data analytics and visualizations and also by providing the mechanisms of integrating various systems and related data.

The European research project CoeGSS - Centre of Excellence for Global System Science [1] has already started to bring different communities from HPC and Global Systems Science (GSS) together to enable decision-making for highly complex, multidimensional problem spaces. However, the process of solving these multi-dimensional problems requires computation and analysis of huge amounts of data and is highly dependent on the precise representation and systematic evaluation of the problem. This is where HiDALGO comes into picture by creating a facility, offering a suite of services to public and private decision-makers in Europe and around the globe to help them taking better and more efficient decisions.

HiDALGO is an advancement of the CoeGSS project that will build up a platform by seamlessly integrating four key concepts: High Performance Computing, High Performance Data Analytics, Artificial Intelligence and Visualisation into an entirely, integrated workflow. Within the HiDALGO project, special consideration is given to the amalgamation of multiple problem statements for accurate simulations of Global Challenges in order to form a sustainable Centre of Excellence for future stakeholders and their particular problem statements.

Moreover, the HiDALGO project is deeply involved in the design and development of the mechanisms and tools by integrating state of the art technologies with innovative approaches in order to increase the precision of simulation results and at the same time, reduce the time to solution. These approaches will not only address the scientific problems within the Global Challenge area but will also be used for data analytics and deep learning based problems [2].

1.1 Purpose of the document

According to the Description of Action (DoA), the document represents the first deliverable (D5.1) of work package 5 (WP5) that highlights the underlying infrastructure for HiDALGO including hardware resources and access to available software and services. Thus, this deliverable reports on the initial steps involved for setting up the HiDALGO system environment.

1.2 Structure of the document

This document is structured into seven sections. Section 2 and 3 are focussing on the hardware infrastructure and available software at the three supercomputing centres, HLRS, PSNC and ECMWF. Section 4 describes the available services and offerings that can be facilitated by the HiDALGO project. Section 5 highlights the details of the PRACE Partnership for Computing in Europe followed by chapter 6, which describes the process of accessing the computing resources at HLRS, PSNC, ECMWF and PRACE. Finally, the last section concludes the overall document.

2 Hardware

This section describes the hardware infrastructure available at the three supercomputing centres, the High Performance Computing Center Stuttgart (HLRS), the Poznan Supercomputing and Networking Center (PSNC) and the European Centre for Medium-range Weather Forecasts (ECMWF). These HPC centres are equipped with thousands of Central Processing Units (CPUs) and storage disks required to provide Massive Parallel Processing (MPP) and, with this the required performance to the applications.

The goal is to setup the HiDALGO infrastructure and provide seamless access to supercomputing resources to handle the process of High Performance Computing and High Performance Data Analytics.

2.1 HLRS

The High Performance Computing Centre Stuttgart (HLRS) was founded in 1996 under the affiliation of the University of Stuttgart as a research and service institution in order to handle the demand of HPC resources for research institutes and industrial customers. HLRS is the first German National HPC centre and a founding member of the GAUSS Centre for Supercomputing that consists of the three large-scale national supercomputing centres including Juelich Supercomputing Centre (JSC) and Leibniz Supercomputing Centre (LRZ).

For the last 20 years, HLRS has put together broad knowledge and expertise in the field of parallel programming, numerical methods for HPC, visualization, grid and cloud concepts and Big Data processing. HLRS has provided support to over 500 users and operated three Petascale systems in the past. It has a dedicated server room of 800 m^2 that houses the computing and storage resources with a power capacity of 5 Megawatts along with an additional facility in a separate building for failsafe operation.

In HLRS, all the rooms housing the computing and storage resources are connected to the Internet over redundant 100 Gbps links. Additionally, another dedicated link to the PRACE network services is provided that is offering a performance of 20 Gbps.

2.1.1 Hardware information

This section contains the hardware specification of the HLRS systems available for HiDALGO. All the systems are described in detail in order to understand the commonalities, but also the differences between their individual system architectures.

2.1.1.1 Hazelhen

Nodes	Cores	Memory	Interconnect	Amount of Storage	Storage Performance
7,712	185,088	~1 PB	Cray Aries	~10 PB	~ 400 GB/s

Table 1: Hazelhen specifications

CRAY XC-40 Hazelhen is a complete homogeneous system and represents the current flagship system at HLRS based on Intel Haswell processors. It has a peak performance of 7.4 Pflop/s and is ranked at 20th position according to the HPCG benchmark in the Top500 list. The system achieved a performance of 0.138 Pflop/s in the release of November 2018 [3], and a Linpack performance of 5.64 Pflop/s that directly translates to the 30th position all over the world [4].

Figure 1 shows a part of the system, which is made up of 41 cabinets and connected via the proprietary CRAY Aries network. It provides two processors each, containing 24 cores and 128 GB main memory per node. It includes 15 dedicated nodes with large memory sizes, accelerators and local disk support for pre- and post-processing of data. Table 1 contains the detailed hardware specification of Hazelhen.

Figure 1: Hazelhen system at HLRS

2.1.1.2 Cray Urika-GX

Nodes	Cores	Memory	Amount of Storage	Storage Performance
64	2,400	33 TB	100 TB	4 GB/s

Table 2: Cray Urika-GX specification

The Cray Urika-GX system is used for High Performance Data Analytics due to its fast storage performance. In particular, each of the nodes is equipped with a Non Volatile Memory Express storage that provides around 5 GB/s reading and writing bandwidth per node. Its software stack consists of Apache SPARK, Hadoop and the proprietary Cray Graph Engine to develop

High Performance Data Analytics applications in R, Python or Java. Table 2 contains the hardware configuration of the system hosted at HLRS.

Figure 2: Cray Urika-GX at HLRS

2.1.1.3 Kabuki (NEC SX-ACE)

Nodes	Cores	Memory	Interconnect	Amount of Storage	Storage Performance
64	256	4 TB	NEC	250 TB	4 GB/s

Table 3: Kabuki specification

Kabuki is a vector system of NEC's SX series and was installed at HLRS in 2014. It is made up of two racks, each for computing and storage [20]. Its nodes are connected through NEC's proprietary high speed network using a crossbar switch configuration for message passing between nodes with 4 GB/s.

It has a total of 64 nodes with 64 GB main memory per node and a memory bandwidth of 256 GB/s [20]. Due to its smaller size it can only execute small scale vector codes, and is not meant for executing large scale production codes. Table 3 contains detailed hardware specification of Kabuki system at HLRS.

Figure 3: Kabuki at HLRS

2.1.1.4 Vulcan (NEC cluster)

Nodes	Cores	Memory	Interconnect	Amount of Storage	Storage Performance
751	10,288	30 TB	Infiniband	500 TB	12 GB/s

Table 4: Vulcan specification

Vulcan is a heterogeneous NEC cluster that contains different processors, memory sizes, compute accelerators and fast local disks. To fulfil the demand of modern HPC applications, the memory sizes per node are up to 256 GB [20]. The main goal of this system is to be used for industrial applications and therefore, the system is setup in two completely independent clusters at two different server rooms to ensure the failsafe operations. Nevertheless, each of the systems provides the individual components and performance detailed in Table 4.

Figure 4: Vulcan NEC cluster at HLRS

2.1.1.5 High Speed Storage System (HPSS)

Storage	Storage Space	Storage Performance	Data Mover
Disk Storage Tape Storage	512 TB 12 PB	20 GB/s 8 GB/s	17

Table 5: HLRS long-term storage

HPSS is a Hierarchical Storage Management System (HSM) and is designed to manage Petabytes of data stored on disks and tape libraries. Currently, HLRS uses a High Speed Storage System (HPSS) that manages 500 TB of disk storage and more than 4 PB of tape storage. The tapes are stored in two spatially separated copies on IBM TS3500 [5] Tape libraries with approximately 4,000 tapes [6].

In the HPSS complex, 19 x86 servers constitute a distributed system that consists of a core server, disk as well as tape movers. Each system is equipped with 32 cores on two sockets using 32 GB (movers) or 128 GB (core server) of main memory [6].

Figure 5: High Speed Storage System at HLRS

2.1.2 Benchmark Systems

The idea of using a Benchmark system is to provide a variety of architectures, different configurations, innovative approaches, or accelerators to support software development and testing. Currently, HLRS hosts three benchmark systems explained in the section below.

2.1.2.1 IBM OpenPower8+

An OpenPower8+ system is part of IBM's power architecture-based server line that was launched in 2016. This system is designed for HPC applications that support Linux and HPDA.

Table 6 contains the configuration details of power8+ system hosted at HLRS.

Cluster (node) configuration:	
Number of Nodes	1
Type/Model of node	IBM Power System S822LC (8335-GTB)
CPU Model name	10-core 2.860 GHz (3.857 GHz turbo) POWER8NVL Processor
Threads per core	8
Cores per socket	10
Sockets	2
Accelerators	2
CPU max MHz	4023
CPU min MHz	2061
L1d cache	64K
L1i cache	32K
L2 cache	512K
L3 cache	8192K
Type of accelerator	GPU (NVIDIA Pascal P100)
Accelerator model	Tesla P100-SXM2-16GB
Accelerators connection	NVLink
RAM	512GB (16x32G RDIMMs) DDR4 1600 MHz
I/O and disks	2*480GB Samsung SSD + NFS
Bus Type	PCIe
Created	10/10/2016
OS Version	Ubuntu 16.04 LTS
Programming Environment	GCC 5.4.0, OpenMPI 1.10.2, Python 2.7.12 + 3.5.2, NetCDF 4.4.0, NetCDF C++ 4.2.1, OpenBLAS 0.2.18, BLAS 3.6.0, LAPACK 3.6.0, ScaLAPACL 1.8.0, Boost 1.58, GDAL 1.11.3, CUDA 8.0
TDP	190W
Power Supply	2*1300W

Table 6: OpenPower8+ configuration at HLRS [21]

2.1.2.2 IBM OpenPower8

This system is a predecessor of the OpenPower8+ system, which has been launched in 2015 based on IBM's power architecture. The difference main difference between both systems is the availability of NVIDIA accelerators, including the NVLink support. Table 7 contains the configuration details of the Power8 system hosted at HLRS.

Cluster (node) configuration:	
Number of Nodes	1
Type/Model of node	IBM Power System S822L (8247-22L)
CPU Model name	10 Core Power 8 CPU
Threads per core	8
Cores per socket	10
Sockets	2
Accelerators	0
CPU max MHz	4023
CPU min MHz	2061
L1d cache	64K
L1i cache	32K
L2 cache	512K
L3 cache	8192K
RAM	256GB (16*16G) DDR3 1600 MHz
I/O and disks	2*600GB Seagate HDD
Bus Type	PCIe
Created	08/10/2015
OS Version	Ubuntu 16.04 LTS
Programming Environment	GCC 5.4.0, OpenMPI 1.10.2, Python 2.7.12 + 3.5.2, NetCDF 4.4.0, NetCDF C++ 4.2.1, NetCDF 4.4.3, OpenBLAS 0.2.18, BLAS 3.6.0, LAPACK 3.6.0, ScaLAPACL 1.8.0, Boost 1.58, GDAL 1.11.3
TDP	190W
Power Supply	2*1400W

Table 7: IBM Power8 configuration at HLRS

2.1.2.3 ARM ThunderX

This system is part of the ThunderX product family of Cavium and is based on 64-bit server-class ARM processors. It was launched in 2016 and is mainly used for data demanding applications and cloud infrastructures.

Table 8 contains the configuration details of the ARM ThunderX system hosted at HLRS.

Cluster (node) configuration:	
Number of Nodes	1
Type/Model of node	CN88XX-CRB-2S
CPU Model name	ThunderX 48 core
Threads per core	1
Cores per socket	48
Sockets	2
Accelerators	0
CPU max MHz	1800
CPU min MHz	2500
L1d cache	32K
L1i cache	78K
L2 cache	16M
RAM	128GB (8*16G RDIMM) DDR4 940 MHz
I/O and disks	1*250GB Seagate HDD
Bus Type	PCIe
Created	22/10/2015
OS Version	Ubuntu 16.04 LTS
Programming Environment	GCC 4.8.5 + 5.4.0, OpenMPI 1.10.2, Python 2.7.12 + 3.4.0 + 3.5.2, BLAS 3.6.0, Boost 1.58

Table 8: ARM ThunderX configuration at HLRS

2.2 PSNC

PSNC is a leading HPC centre in Poland with its 1.7 Pflops/s of computing power, 150 TB memory, a disk space of 7.5 PB, and a mass storage of 47 PB, providing direct support for the scientific communities in Poland as well as in Europe (e.g. Nuclear Fusion, Astrophysics, Bioinformatics, Chemistry, Nanotechnology). PSNC is also the operator of the Polish National Research and Education Network PIONIER, which is connected to the GEANT2 network, and furthermore, is the operator of the Poznań Metropolitan Area Network - POZMAN.

The PIONIER network is a nationwide broadband optical network that represents a base for research and development in the area of information technology and telecommunications, computing sciences, and applications as well as services for the Information Society. PIONIER connects 21 academic network centres, including centres of the Metropolitan Area Networks (MAN) and 5 of the HPC centres (participating as third parties in PRACE) using their own fibre connections. Besides the network facilities, PSNC has other significant facilities, especially with respect to HPC and storage.

PSNC hosts two dedicated data centres (1,600 m²) 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. A view on the currently installed Petascale system is presented in Figure 6.

Figure 6: 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 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.

2.2.1 Hardware Information

Below a detailed description concerning the available hardware at PSNC for HiDALGO is provided.

2.2.1.1 EAGLE System

Table 9 contains a detailed hardware specification of the EAGLE system at PSNC.

Nodes	Cores	Memory	Interconnect	Amount of Storage	Storage Performance
1,233	32,984	120.6 TB	56 Gb/s Infiniband FDR	4.6 PB Lustre 1 PB /home	32 GB/s Lustre File system 10 GB/s /home CNFS

Table 9: EAGLE System Specification at PSNC

a. EAGLE CPU Nodes

Table 10 contains the specification details of CPU nodes of the EAGLE system.

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 E5-2697 v3	589	2x14	64 GB	1.1 TFLOPS	intel,haswell,huawei
HUAWEI CH121 V3	Intel Xeon E5-2697 v3	530	2x14	128 GB	1.1 TFLOPS	intel,haswell,huawei
HUAWEI CH121 V3	Intel Xeon E5-2697 v3	59	2x14	256 GB	1.1 TFLOPS	intel,haswell,huawei
HUAWEI CH121 V4	Intel Xeon E5-2682 v4	55	2x16	128 GB	1.1 TFLOPS	intel,haswell,huawei

Table 10: EAGLE CPU nodes

b. EAGLE GPU Nodes

Table 11 contains the specification details of GPU nodes of the EAGLE system.

CPU model	No. of nodes	No. of processors and cores	RAM per node	GPU model	No. graphic cards per node	Computing power per GPU
Xeon Gold 5115	3	2x10	92 GB	NVIDIA V100	2	7,8 TFLOPS

Table 11: EAGLE GPU nodes

c. EAGLE Operating System

Table 12 contains the information about the EAGLE operating system, its domain name and the task management system.

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

Table 12: EAGLE OS

2.2.1.2 Storage or Tape Library

Table 13 contains information about the type and performance of the storage system at PSNC.

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

Table 13: PSNC Storage

In summary, a detailed view on the schematic infrastructure and their relations at PSNC can be obtained from Figure 7 below.

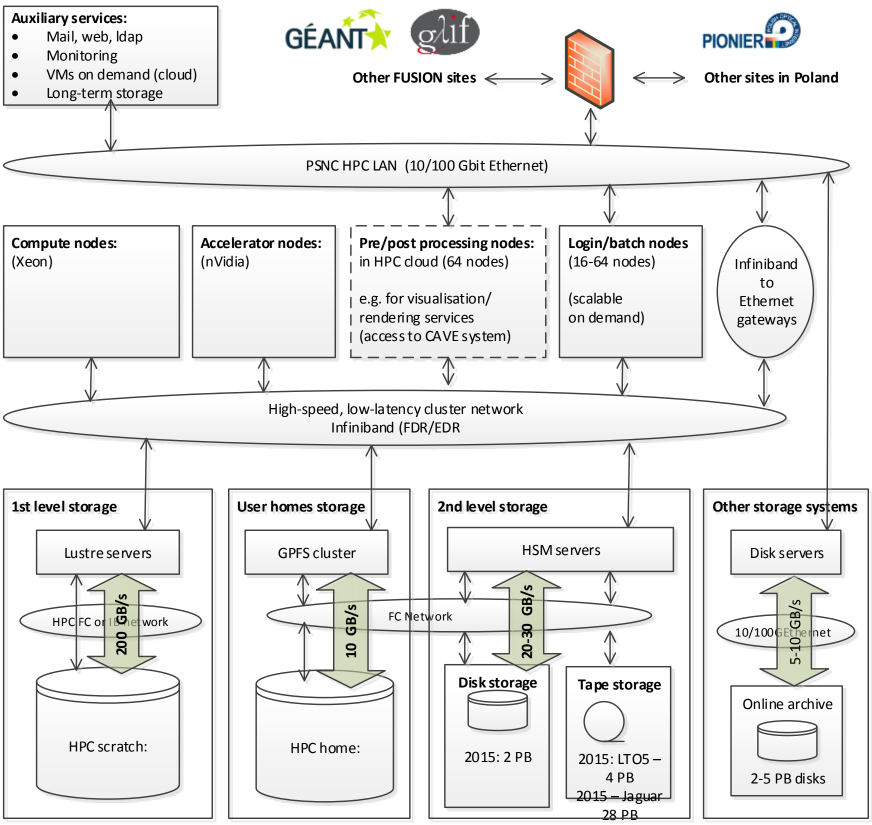


Figure 7: PSNC data infrastructure and correlation with HPC

2.2.1.3 Special Purpose Nodes

In order to facilitate the fast prototyping phase of the foreseen HiDALGO implementations, the following three bold virtual machines (VMs) have been set up:

HiDAP - Urban Pollution (Grant-32543)

Cores	RAM	Disk	IP Address	Operating System
32	32 GB	1040 GB	62.3.170.157	Ubuntu Server 18.04 LTS

Table 14: HiDAP Configuration Details

HiDMiG – Migration (Grant-32542)

Cores	RAM	Disk	IP Address	Operating System
32	32 GB	540 GB	62.3.170.212	Ubuntu Server 18.04 LTS

Table 15: HiDMiG Configuration Details

HiDSN - Social Networks (Grant-32544)

Cores	RAM	Disk	IP Address	Operating System
32	32 GB	540 GB	62.3.170.209	Ubuntu Server 18.04 LTS

Table 16: HiDSN Configuration Details

2.3 ECMWF

ECMWF manages a supercomputing facility for its Member and Co-operating States, providing vital resources for computer modelling of the global atmosphere and ocean and for weather forecasting research. The machine is operated exclusively for time critical work to serve weather and related forecasts. This differentiates it from other HPC centres within the HiDALGO consortium where different works can be scheduled on request. By participating within HIDALGO, ECMWF hopes to show how such single purpose, restricted access HPC's can be integrated into a HPC/HPDA/Cloud workflow.

2.3.1 Hardware information

This section contains the hardware specification of the ECMWF systems available for HiDALGO.

2.3.1.1 Cray XC40 system

The main HPC system comprises two identical Cray XC systems designed for having two self-sufficient compute clusters with their own storage, but each with equal access to the high performance storage of the other cluster. This cross-link storage mount allows most of the benefits of having one very large system but dual clusters add significantly to the resiliency of the system, allowing flexibility in performing maintenance and upgrades. When combined with separate resilient power and cooling systems they provide protection against a wide range of possible failures, paramount to a 24/7 operational environment. Table 17 contains detailed hardware specification of Cray XC40 cluster at ECMWF.

Nodes	Cores	Memory	Interconnect	Storage Performance
38,40	138,240	-	Aries Interconnect	16 GB/s

Table 17: Cray XC40 Specification at ECMWF

The system has two identical Cray XC-40 clusters, each has 20 cabinets of compute nodes and 13 of storage. The bulk of the system consists of compute nodes with two Intel Xeon EP E5-2695 V4 “Broadwell” processors each with 18 cores. Four compute nodes sit on one blade, sixteen blades sit in a chassis and there are three chassis in a frame. This gives a maximum of 192 nodes or 6,912 processor cores per cabinet. The number of actual compute nodes in a cabinet will sometimes be less than the maximum since as well as compute nodes, each cluster has a number of “Service Nodes”. These have space for a PCI-Express cards to support a connection to external resources such as storage or networks and are consequently twice the size of a compute node, with only two fitting on a single blade.

Each blade in the system is connected with a single Aries interconnect chip and all the nodes on the blade connect to it via PCI-Express Gen3 links capable of a transfer rate of 16 GB/s in each direction. Each Aries chip then connects via the chassis backplane to every other blade in the chassis. A chip has five other electrical connections, one to each chassis in a group of two cabinets, forming an "electrical group". A further level of network uses optical links to connect every "electrical group".

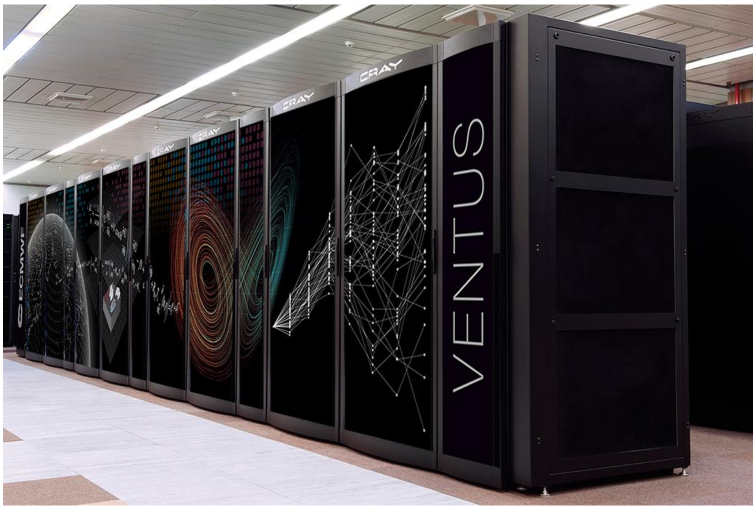


Figure 8: Cray XC40 system at ECMWF

2.3.1.2 High Performance Storage

High performance working storage for the compute clusters is provided by Lustre file systems from integrated Cray Sonexion appliances. Each cluster has two main pools of storage, one for time-critical operational work, the other for research work. Segregating time-critical from research storage helps avoid the I/O contention between workloads and thus limits the variability of run times for time critical work. A similar segregation will be required for the Hidalgo workflows running in the Cloud environment but requiring data produced in the HPC system.

While each cluster has its own high performance working storage and is self-sufficient, it also has access, at equal performance, to the storage resources of the other cluster. This cross mounting allows work to be flexibly run on either cluster, in effect making it work like a single system. Each XC40 system has about 10 Petabytes of storage and offers more than 350 Gigabytes per second of I/O bandwidth.

Storage	Storage Performance	Data Mover	Data Mover Performance
10 PB	350 GB/s	-	-

Table 18: ECMWF Storage

2.3.1.3 Data Handling System

For three decades, ECMWF has operated a large-scale data handling system (DHS), in which all ECMWF users can store and retrieve data that is needed to perform weather modelling, research in weather modelling and mining of weather data.

From the user viewpoint, DHS supports two main applications developed by ECMWF to hide the complexities of the underlying storage management from users:

- *MARS*, the Meteorological Archival and Retrieval System, provides access to a powerful abstraction engine that allows staff and applications to access the meteorological data that has been collected or generated at ECMWF for more than 30 years. MARS stores GRIB and BUFR data, hiding from its users all of the details concerning the physical location and internal organisation of this data. It manages its own set of disk caches for staging data that has been recently acquired, generated or accessed. However, the bulk of its information is stored in HPSS.
- *ECFS* provides users with a logical view of a seemingly very large file system and is used for data that is not suitable for storing in MARS. UNIX-like commands enable users to copy whole files to and from any of ECMWF's computing platforms. ECFS uses the storage hierarchy of disks and tapes within HPSS to store the files and their associated metadata (file ownership, directory structure, etc.).

MARS data represents about 80% of the volume of data stored in the DHS, but only about 6% of the number of files. ECFS data represents the remaining 20% of the data, corresponding to 94% of the files. The DHS provides access to over 260 PB of primary data. An additional 46 PB of backup copies of part of the primary data are stored in the DRS. There are about 260 million files in ECFS and over 18 million in MARS.

Supporting ECFS and most of MARS is an underlying file archiving component, IBM's High Performance Storage System (HPSS), in which data is kept and managed. It keeps track of files that are stored, provides Hierarchical Storage Management (HSM) facilities when needed, and it manages activities related to disks, tapes, tape drives and automated tape libraries.

HPSS is based on version 5 of the IEEE Mass Storage Reference Model. It supports a variety of tape drives and automated tape libraries. Data is transferred from multiple storage devices via multiple data streams over multiple network paths; in this way high aggregate transfer rates are achieved. 'Data movers' (specialised software modules), which can execute on different server machines, send streams of data directly between those servers and the client machines requesting the data transfer. This distributed multi-processing nature of HPSS is one of the keys to its scalability. In turn, HPSS uses IBM's DB2, a high-performance database management system with advanced transaction-processing techniques, to guarantee security, protection and integrity of data.

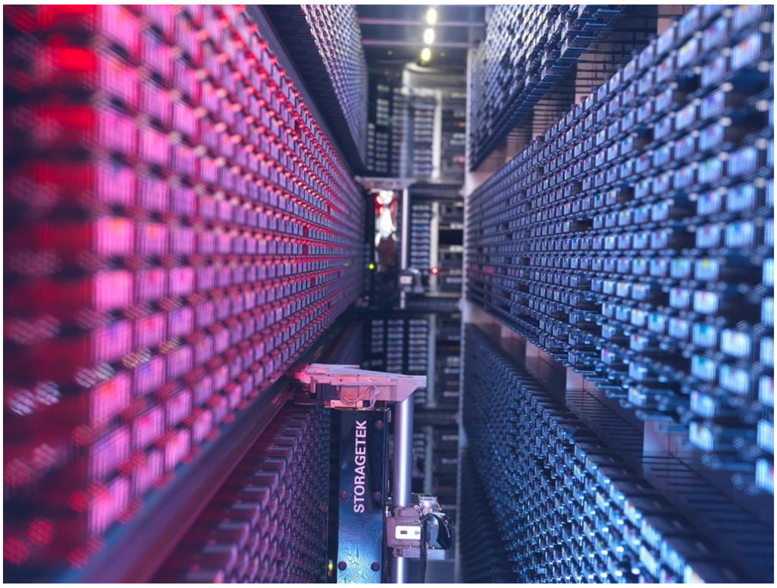


Figure 9: Data Handling System at ECMWF

2.3.1.4 ECMWF Cloud Infrastructure

ECMWF has scaled its computations on supercomputers heavily over the last decades to run ever more complex models of the atmosphere and oceans. What became clear is that users struggle to catch up making use of all the new data being generated. From these experiences ECMWF, for its own and Copernicus related activities, concluded that to make outcomes of Peta-/Exascale computations easily available users need a more flexible and more familiar environment to access and process model outputs. Traditional workflows limit the user uptake and if models go into Exascale with related data amounts this challenge will even grow more. For this reason, ECMWF decided to put considerable efforts into developing cloud environments co-hosted to the HPC. These environments will be very close to the HPC offering very fast and easy access to HPC resources, such as disks.

ECMWF will participate with two cloud infrastructure projects:

- **DIAS WEkEO:** European Commission launched the Copernicus Data and Information Access Services (DIAS) to allow faster access to services of the Copernicus Programme for institutions which lack the necessary resources to handle large data amounts themselves. WEkEO is the DIAS reference service for environmental data, virtual environments for data processing and skilled user support. WEkEO DIAS Service is implemented by EUMETSAT, ECMWF and MERCATOR OCEAN.
- **European Weather Cloud:** - A private cloud initiative by ECMWF and EUMETSAT to make it easier for users to access data products from both organisations. It will focus on the requirements of the MS of both organisations.

While the two clouds will have different purposes and business models, technology wise they will be quite similar. The user layer to interact with the cloud will be Morpheus with OpenStack as software platform offering the infrastructure-as-a-service on which applications and services are build.

The success of ECMWF's archive of operating more than 30 years without users having to change how they work with it has been thanks to the design of an API. A similar approach has been successfully demonstrated for two decades with APIs allowing users to define services from ECMWF (e.g., real time dissemination of data). The API protects users from effects of internal changes and make it stable and a good investment over time. The API, thanks to its declarative nature, has survived changing environments in IT and various changes in programming languages. During HIDALGO, ECMWF wants to explore more the use of APIs to access data & services from the HPC & archive. This should demonstrate how users on the cloud environment and in the HiDALGO workflows can integrate results from ECMWF in a sustainable way.

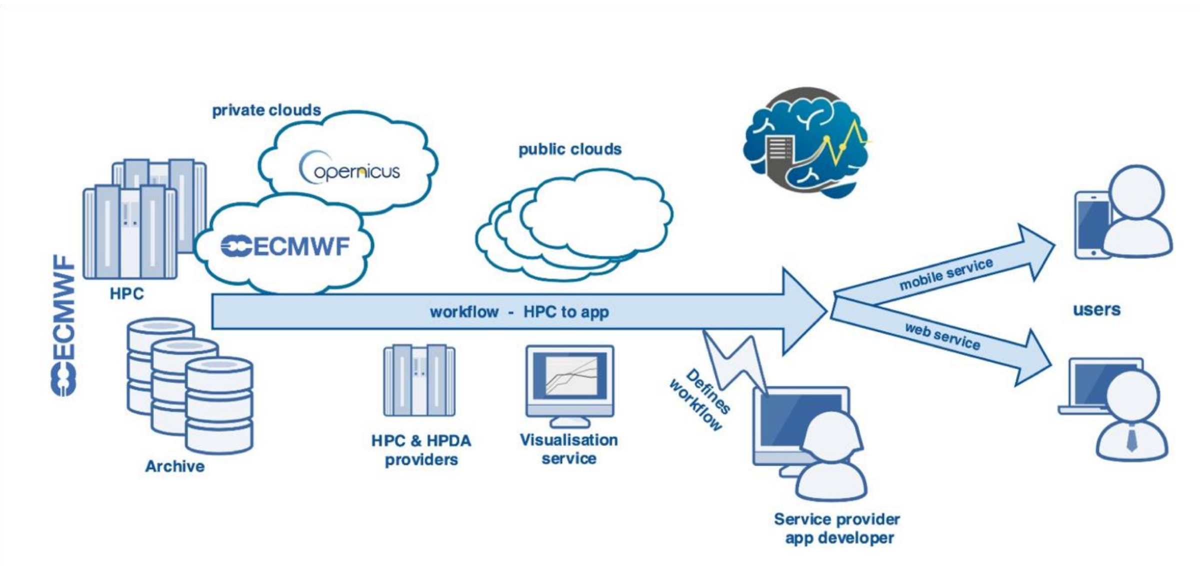


Figure 10: ECMWF Cloud Infrastructure

3 Software

This section describes the availability of software packages at HLRS, PSNC and ECMWF for the HiDALGO project. Software is an essential aspect in High Performance Computing, since HPC applications are not much different from standard computing applications except the fact that HPC applications are developed to be executed in parallel and requires a high-performance, physical infrastructure with appropriate libraries and compilers to achieve the highest possible degree of performance and efficiency.

Batch mode is always preferred for executing these applications on the clusters, which will be discussed in detail in the following sections. However, this section will only cover the available software packages for cluster management and executing the user's application on the clusters available at HLRS, PSNC and ECMWF.

3.1 HLRS

HLRS provides a broad variety of software on its systems. This software can be grouped in the following categories:

- **Development** tools including version control systems, configuration and building tools, compilers, interpreters, as well as parallel programming libraries (MPI, TBB, etc.).
- **Code analysis** tools including debuggers, profilers, and profiling libraries (PAPI, etc.).
- **Scientific libraries** including numerical libraries, network analytics libraries, etc.
- **Scientific applications**: besides visualization, computer-aided engineering (CAE).
- **Computational Fluid Dynamics** (CFD) such as for computational chemistry (GAMESS-US, Gaussian, Molpro, MOPAC, TURBOMOLE, and VASP) or computational structural mechanics (ABAQUS, LS-Dyna, Permas, and MD FEA bundle).
- **Miscellaneous** software (workload management, containerization, etc.).

Table 19 reflects the state of the software stack installed on the Hazelhen system as of January 2019. Due to lack of space, the table collects information only on the software packages, which are relevant for the pilots of the HiDALGO project. Default versions are marked in bold. Note even though available versions of some software differ from cluster to cluster, the core packages listed in Table 19 are uniformly present on all HLRS systems.

Name	Version(s)	Purpose
gnu-tools	201805	Development tools
git	2.9.5, 2.12.3	Version control

svn	1.8.19	Version control
GCC	6.1.0, 7.3.0, 4.9.3, 5.3.0	Compilers
ICC	14.0.3, 16.0.3, 17.0.4, 18.0.1, 19.0.0	Compilers
LLVM	5.0.1	Compilers
JDK	1.7.0, 1.8.0	Java Development Kit
Cray Python	2.7.15.1, 2.7.15.3, 3.6.1.1, 3.6.5.3	Python interpreter (with libraries)
Cray R	3.4.2, 3.3.3	R interpreter (with libraries)
Cray mpich	7.7.1, 7.3.2, 7.6.3, 7.7.0, 7.7.3	Optimized MPI library
Intel TBB	2019.0	Multithreading library
Boost	1.66.0, 1.59.0	C++ non-standard libraries
Cray CCDB	3.0.4, 3.0.3, 2.0.0	Debugging
DDT	18.0.2.1	Debugging
GDB4HPC	3.0.10	Debugging
Valgrind4HPC	1.0.0	Debugging (memory leaks, etc.)
PerfTools	7.0.4, 6.5.2,	Profiling
PAPI	5.4.3.1, 5.5.1.4, 5.6.0.4	Profiling
Advisor	2018.1, 2018.2	Profiling
Vampir	9.4.0	Profiling
Inspector	2018.1, 2018.2	Profiling
CrayPAT	2.5.15, 2.5.14	Profiling
Intel Vtune	2018.2, 2019.0	Profiling
Extrae	3.5.2	Profiling
Cube	4.4	Profiling
Paraver	4.7.1	Profiling
Cray Trilinos	12.12.1.1, 12.10.1.2	Numerical libraries
Cray TPSL (ParMeTiS, etc.)	16.03.1, 17.06.1, 18.06.1	Numerical libraries bundle
Inter MKL	2018.1, 2019.0, 2018.2	Numerical libraries bundle
Cray Petsc	3.7.6.2, 3.7.6.0, 3.8.4.0, 3.6.3.0	Numerical linear algebra
SLEPc	3.8.3, 3.7.4	Numerical linear algebra
Cray LibSci	16.03.1, 17.11.1, 18.07.1	Numerical linear algebra bundle
snap	4.0	Network analytics library

Cray PE ML plugin	1.0.1	Machine learning (Python 2/3)
Cray HDF5	1.8.16, 1.10.2.0	Scientific data format
Cray NetCDF	4.4.1.1.6, 4.6.1.2, 4.6.1.3	Scientific data format
Dakota	6.7	Uncertainty quantification
ANSYS	15.0, 16.2, 17.2, 18.1, 19.2	Computational fluid dynamics
OpenFOAM	4.0, 5	Computational fluid dynamics
STAR-CCM+	8.06, 9.06, 10.04, 11.06, 12.02, 13.06	Computational fluid mechanics
Ansa BETA	14.2.5, 15.3.1, 17.1.1, 16.1.0	Computer-aided engineering
Pointwise	18.0, 18.1, 17.3	Computer-aided engineering
VTK	8.1.0	Visualization
ParaView	5.5, 5.3.0	Visualization
coviseOSMesa	1.0	Visualization
Environment Modules	3.2.6.7, 3.2.10.6, 3.2.11.1, 4.1.3.1	PE control
MOAB	9.1.1	Workload management
TORQUE	6.1.1.1, 6.1.2	Resource management
Shifter	16.08.4	Containerization

Table 19: List of software available at HLRS

Most of the software is available for both academic and commercial use. The notable exceptions are:

- `ANSYS` is for non-commercial use only;
- Current CD-Adapco license allows to use `STAR-CCM+` for University of Stuttgart members and non-profit partners only.

We refer the interested reader to the HLRS wiki [7] for a more detailed information on the availability of software. Note though the software information is nicely maintained, sometimes it misses data on the latest software updates. This data can be always checked by calling the command `module avail` on the systems.

3.1.1 Cluster Management Software

In order to manage applications launched on the HPC systems, HLRS uses the TORQUE [8] distributed resource manager in conjunction with the Moab [9] scheduler. TORQUE supports three types of user jobs:

- **Interactive** jobs that allow to run commands on compute nodes interactively. They are intended for small tests and development tasks like debugging. Resources, like number of nodes, memory, and processor time are very limited for this type of jobs.
- **Batch** jobs used for production runs;
- **Heterogeneous** jobs – a special type of batch jobs utilizing multiple types of execution nodes simultaneously (e.g., graphic nodes and cluster nodes).

TORQUE does not provide a comfortable GUI for job submissions. Nevertheless, all kinds of functionality are accessible via the command line interface. The command `qsub` allows to submit jobs with TORQUE. In order to submit an interactive job, the user must specify the option `-I`.

E.g., the following command allocates one node with 24 cores for interactive jobs.

```
qsub -l nodes=1:ppn=24 -I
```

As soon as resources are allocated, the user can launch commands on the allocated compute nodes via the command line interface. Batch jobs – simple and heterogeneous – require the preparation of the batch scripts for their submission. These scripts serve as `qsub` input including a specification of commands that should be executed on the compute nodes.

```
qsub -l nodes=1:ppn=24 ./my_hidalgo_batchjob_script.pbs
```

As soon as the batch job is submitted, it receives unique identifier. This identifier enables to trace a job state with the `qstat` command and delete jobs from the execution queue via the `qdel` command. For more details on these commands, one can run `man qsub/qstat/qdel`.

TORQUE batch scripts are shell scripts containing flags and commands to be interpreted by a shell and are used to run a set of commands in sequence. In contrast to the common shell scripts, TORQUE batch scripts allow to set details of resource allocation with `#PBS`-directives and provides some additional environment variables (e.g., `$PBS_O_WORKDIR`).

The typical single node TORQUE batch scripts looks like this:

```
#!/bin/bash
#PBS -N hidalgo_job_name
#PBS -l nodes=1:ppn=24
#PBS -l walltime=00:20:00

# Change to the directory that the job was submitted from
cd $PBS_O_WORKDIR

# Launch the application
./my_hidalgo_executable arg1 arg2
```

In case of MPI applications, the user must prepend calls of executables with the specification of a Message Passing Interface launcher (e.g., `aprun -n 24 ./my_hidalgo_executable` runs `my_hidalgo_executable` in 24 MPI processes on Cray systems). A full list of all TORQUE environment variables and `#PBS`-parameters can be found in the official documentation [8].

3.1.2 Get your code running

In order to be launched on a cluster, the codes must be compiled on the actual system before the execution. The user must provide correct environment settings, not only during the configuration and compilation of the software, but also when this software is executed (namely, in interactive `qsub` sessions and in TORQUE batch scripts). In order to simplify the environment control, HLRS clusters use environment modules. Environment modules allow to support multiple versions of software, to switch easily between them, to disable unused packages, and to create integrated software packages. The command `module` implements all these operations. In particular, one can load, for instance the default GNU compilers [10], with:

```
module load gcc
```

The option `switch` allows to switch from the default GNU compilers to some specific version.

```
module switch gcc gcc/4.9.3
```

The option `list` helps to list all active modules.

```
module list
```

For more information, the command `man module` shows detailed information on the `module` command.

Nevertheless, if some special packages are required, but not installed on the clusters, there is still the opportunity to install missing components into the `$HOME` folder.

3.1.3 Install own software

The user is permitted to install personalized software packages into the `$HOME` directory without contacting a system administrator. HLRS clusters support the standard UNIX toolchains for configuring, compiling, and installing new software packages from sources. In particular, they have the `CMake` and GNU build systems (`autoconf`, `automake`, `libtool`, `gnulib`, GNU `make`). Moreover, `SCons` can be easily installed as Python package with `pip`. In addition, HLRS clusters have modules for fast and safe switching between programming environments for four popular C/C++/Fortran compiler sets:

- `PrgEnv-cray` for Cray compilers;

- `PrgEnv-gnu` for GNU compilers;
- `PrgEnv-intel` for Intel compilers;
- `PrgEnv-pgi` for PGI compilers.

Usually the installation of C/C++/Fortran packages consists of the following steps:

- Prepare environment (choose compilers, load dependencies, etc.);
- Configure sources with `cmake` or a specific `configure` script;
- Build, test, and install the package with GNU `make`.

Installation of Python and R packages is also standard for UNIX systems. Python packages can be installed with `setup.py` or `pip`. R packages are installed by calling `install.packages` from `Rscript`.

In cases that require assistance, the users can request for support via the trouble ticket system: http://www.hlrs.de/support/trouble-ticket-submission-form/.

3.2 PSNC

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 Vectorization/Optimization 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

Name	Version(s)	Purpose
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

Table 20: PSNC software list

A complete documentation for installed software along with sample scripts to make use of them can be found here: https://wiki.man.poznan.pl/hpc/index.php?title=Oprogramowanie.

3.2.1 Cluster Management Software

In this chapter, 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 [11] as resource management system. A batch job can be submitted using the `sbatch` command 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 following queues:

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 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

Table 21: Types of Queues on EAGLE

- **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 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 point of view, it is sufficient to log in to the cluster with the `-X` option `ssh -X bellis` 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
```

3.2.2 Get your own code running

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

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 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 nodes or applications that use the `/home` directory for storing intermediate results will be terminated without warning.

- **Modules**

Modules are the standard methods used to manage the user environment for clusters. It is enough for the user to `load` modulea 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, one has to unload the current-in-use module and load the module associated with the new version.

- **Usage**

Listing loaded modules

To list loaded modules please use the following command:

```
module list
```

To list modules available for a given user please use the following command:

```
module avail
```

Additional info about the module can be obtained by executing the command:

```
module help <module_name>
```

Loading a module is done as follows:

```
module load <module_name>
```

A module can be unloaded this way:

```
module unload <module_name>
```

Checking for environment variables modified by given module is available via:

```
module show <module_name>
```

- **Settings customization**

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 in which 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
...
```

3.2.3 Install own software

It is important 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 an email to support-hpc@man.poznan.pl. In the consecutive part of the subchapter, we discuss how it should be done correctly.

- **Software Compilation**

To compile the software on a HPC system we need to access the interactive console. To do so, a user must submit an interactive job and compile in interactive mode in order 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, which shall be avoided in any sense.

Example for using SLURM:

```
srun --pty /bin/bash
```

To access a node with a certain number of cores a user needs to specify `-n` parameter:

```
srun --pty -n 12 /bin/bash
```

If a user wants to reserve a certain amount of RAM, the `--mem` parameter needs to be used:

```
srun --pty -n 12 --mem 16000 /bin/bash
```

When the user accesses the console directly at the node, the modules for the compilation need to be loaded, such as:

```
module load gcc
```

The last step is to compile the software in the `/home` directory, according to the manufacturer's recommendations, e.g.:

```
./configure --prefix=home_folder
make
make install
```

In order to utilise compilation processes on several cores, the user can add the `-j` parameter:

```
make -j 12
```

- **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 (e.g., Matlab, Maple, etc.) or compile a particular application. The following is an example on how to run a task in interactive mode.

Interactive job submission:

```
srun --pty /bin/bash
```

A user has to add the `-n` parameter to access the node with the specified number of cores:

```
srun --pty -n 12 /bin/bash
```

If a user wants to reserve a certain amount of RAM, the `--mem` parameter needs to be used:

```
srun --pty -n 12 --mem 16000 /bin/bash
```

3.3 ECMWF

As described in section 3.3 above, ECMWF's compute facilities are strictly managed and access within itself is limited. This is a consequence of the operational stability the systems have to provide.

The normal workflow of ECMWF is to setup suites which users are able to interact and manage through the ecFlow suite manager. This suite manager sits on top of the job scheduler and abstracts system or scheduler specific behaviour. Suites are often developed on Linux workstations and then move to production Linux clusters or the HPCs. Suites run on one machine/cluster but can also be spread over various systems at ECMWF.

The suites themselves can make use of third-party software, compiled code and/or scripts. Often software developed by ECMWF is used, since it is specialized for fast efficient data access and to understand the specific data formats which are used in weather and climate sciences.

During the HiDALGO project, workflows controlled through the orchestrator should trigger work on the cloud machines. The ecFlow scheduler has a Python interface to describe suites and this can be used to package work for ECMWF environments. Using this mechanism has the advantage that ecFlow allows triggers from outside source like models running on the HPC.

Name	Version(s)	Purpose
MARS client	6.24	Retrieval & Regridding of data from MARS archive
Metview	5.5	Batch & desktop tool which build on top of the MARS client
ecPDS	2019/01	Real-time dissemination system to
Magics	4.0	Meteorological plotting package
ecFlow	5.0	Suite management tool on top of job schedulers
FDB	5.0	Field database for fast access to forecast data on disk

Table 22: ECMWF software

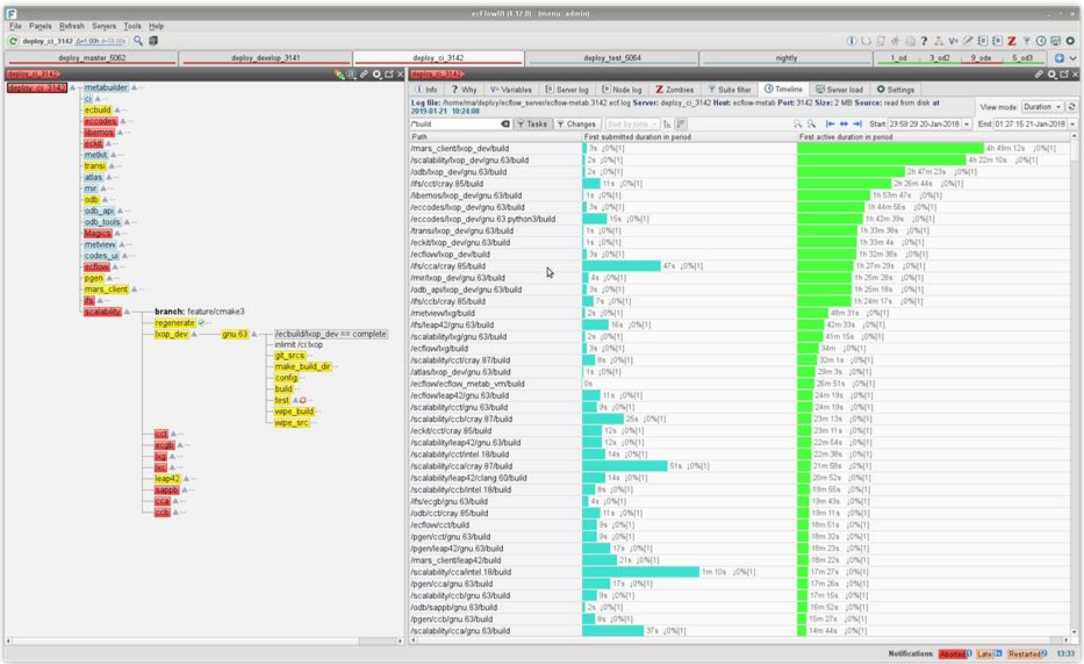


Figure 11: Snapshot of the ecFlow suite manager

3.3.1 Cluster Management Software

Like any HPC centre, ECMWF's HPC systems are set up with software for the management of the clusters. Since these will not be exposed to users in the project they are not further described here.

Users will be instead exposed to the ECMWF cloud infrastructures described in section 3.3. These cloud environments will be managed through OpenStack. This software presents users with an interface to define & create their own resources in forms of VMs, disks and networks to build their applications. During the life time of the HiDALGO project ECMWF plans to implement an additional layer on top of OpenStack using the Morpheus software. This package will allow users to move VMs from one cloud environment to another. This could include a future HPC environment if this is supported by vendors.

3.3.2 Get your own code running

ECMWF expects their users to use the cloud environments to bring their own codes and perform the compilation themselves. ECMWF normally gets involved in optimizing codes for platforms, especially disk access since most processing involves the handling of large data amounts.

3.3.3 Install own software

Deployment of software packages at ECMWF premises is managed by the User Support team. If users want to install their own software normally have to do this in their own environments by themselves. One of the motivations of offering cloud environments is to allow users build easier environments with their own preferred tools. The expectation is that users are using standard deployments tools & package managers to do so.

4 Services and Offerings

This section describes the services that are offered by HLRS, PSNC and ECMWF. The services such as user support that are already available will be used as a baseline for the further development within the HiDALGO project.

4.1 HLRS

HLRS has already established various service offerings that include the provisioning of computing resources for small and large-scale organizations, training and education services for academic and industrial users as well as particular end user support. The following subsection describes a variety of available services at HLRS that are been offered all around Europe.

4.1.1 Operation of the IT infrastructure

Over the past 20 years since the foundation of HLRS in 1996, it has proved its expertise in building the process of providing efficient and reliable services to the academic and industrial users. It has over 100 people working on system operations, code optimization as well as in research and development of innovative technologies. This fact is underpinned by the various Petascale systems that HLRS has setup and operated during the past decade.

For handling daily operations of computing and storage resources along with network operations and power supply, HLRS has experienced teams consisting of experts from different domains. These teams are also involved in European and German research projects along with the operations of the local infrastructure at HLRS.

4.1.2 User services and support

HLRS not only supports system operations as described in the previous section but also offers services in order to support users to guarantee a reliable and effective utilisation of the HLRS infrastructure. These services include daily user support, training, code optimization and efficient usage of HLRS computing clusters. These service offerings can be reused or enhanced to match the requirements of the HiDALGO project. However, if this is not possible, the services including the experience to set them up can be used as baseline for further developments.

- **Resource offering**

HLRS offers a variety of supercomputing systems reflecting the different needs of its scientific and industrial customers. It provides the opportunity for using the available High Performance Computing to enterprises. To provide the compute services to the users, HLRS has extended the job scheduling system, which sets priorities to relevant jobs in order to provide precise and swift results. HLRS delivers extensive expertise for supporting scientific applications in various disciplines, also as a result through the association with the PRACE service offerings.

- **User support**

For issues related to user support, HLRS offers an Open Ticket Request System (OTRS), personal email contact and a dedicated phone line. Users facing issues are handled on personal basis in a fast and effective manner. To ensure effective user support, experts at HLRS closely collaborate with the experts at the vendors' sites in order to deliver the best possible solution to its customers.

- **Consulting**

HLRS offers consulting services to the users in the area of code optimization, benchmarking of HPC systems, profiling and debugging tools, amongst others. HLRS employs experts from various domains that analyse and solve the problems by providing a customized solutions or optimize the performance of existing codes.

- **Training**

HLRS plays a major role in providing training courses to academic and industrial users. It has a dedicated training department that organizes around 50 training programs each year in the areas of High Performance Computing that includes parallel programming, programming for scientific computing, scientific visualizations and many more [8]. HLRS offers both classroom and online training courses, an always up-to-date list of courses can be found under the following link: https://www.hlrs.de/training/.

Furthermore, as a member of the PRACE consortium, HLRS also organises courses through the PRACE Advanced Training Centre (PATC) for the users all around Europe.

- **Co-design**

As mentioned in the earlier section, HLRS works in collaboration with world-class HPC vendors to analyse and upgrade the existing systems. For this purpose, an innovation management team has been setup, which takes care about novel architectures for any kind of High Performance Computing [13].

4.2 PSNC

PSNC as a hosting centre and national network operator has teams of people that can fully support all aspects of computing (especially HPC) and data infrastructures.

4.2.1 Operation of the IT infrastructure

PSNC is hosting HPC machines for more than 25 years and has an experienced team responsible for both, 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.

4.2.2 User services and support

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 optimization 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 basis. It reacts in emergency situations and can provide the tentative support nearly on-line. However, all user queries and problems must be registered in the PSNC Ticketing System (TTS).

- **Training centre**

Since 2005, PSNC has organized training sessions for the HPC users coming from different research areas. During approximately 100 trainings, 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.

- **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 users having HPC accounts in PSNC.

- **User support on security**

The user becomes a critical link in the security chain of every system or infrastructure. Even an optimally protected infrastructure may be endangered if its users tend to behave in an inaccurate way. Thus, it is proposed that activities be undertaken focused on supporting different groups of users involved in the HPC fusion infrastructure in the security area. The majority of the activities will be performed during the whole project lifetime. They will be focused also on users of the infrastructure who may not have Information Technology security skills, or even lack on general domain knowledge.

4.3 ECMWF

ECMWF provides global weather forecasts, climate re-analyses and specific datasets, designed to meet different user requirements. These are disseminated via the web, point-to-point dissemination, data servers and broadcasting.

4.3.1 Operation of the IT infrastructure

The current operational environment of ECMWF produces numerical weather forecasts on a daily cycle of four forecasts per day, identified by the valid hour of the run: 00Z, 06Z, 12Z and 18Z.

The main runs are 00Z and 12Z, which provide a high-resolution (HRES) 10-day forecast, with a global grid of 9 KM and 137 atmospheric levels, supplemented with a 51 members' ensemble (ENS) of 15-day forecasts on a global grid of 18 KM and 91 levels.

The 06Z and 18Z are smaller runs intended to supply our clients with more boundary-conditions for local area models, in the intermediate hours between the larger runs. As a consequence, these are shorter, extending only to 5-days at the same resolution as the ENS.

All these runs are executed on a tight time-critical schedule according to a strict Service Level Agreement with our clients and Member-States. The schedule is roughly one hour from the moment we stop receiving weather observations to the delivery of the weather products to the user. This makes our HPC and ancillary systems highly sensitive to external interaction, and thus limits the way users of ECMWF data will interact with ECMWF's systems in the HiDALGO project.

At a high level, these time critical workflows form a pipeline of data flow. It starts with the assimilation of weather observations from all over the world, which are received in ECMWF's data centre and processed by the data acquisition systems. These observations are then filtered and assimilated into the model, where they are used to correct the previous forecast in order to generate the initial conditions for the next forecast. The initial conditions are then used to start the model forward time stepping that leads to the next weather forecast. The model writes its raw model output in global fields. These still need further post-processing to create user-specific weather products. ECMWF has a dedicated workflow for the product generation, which gathers daily the users' requests and computes the products, handling them to the dissemination system that ensures their timely delivery to the users chosen locations around the world.

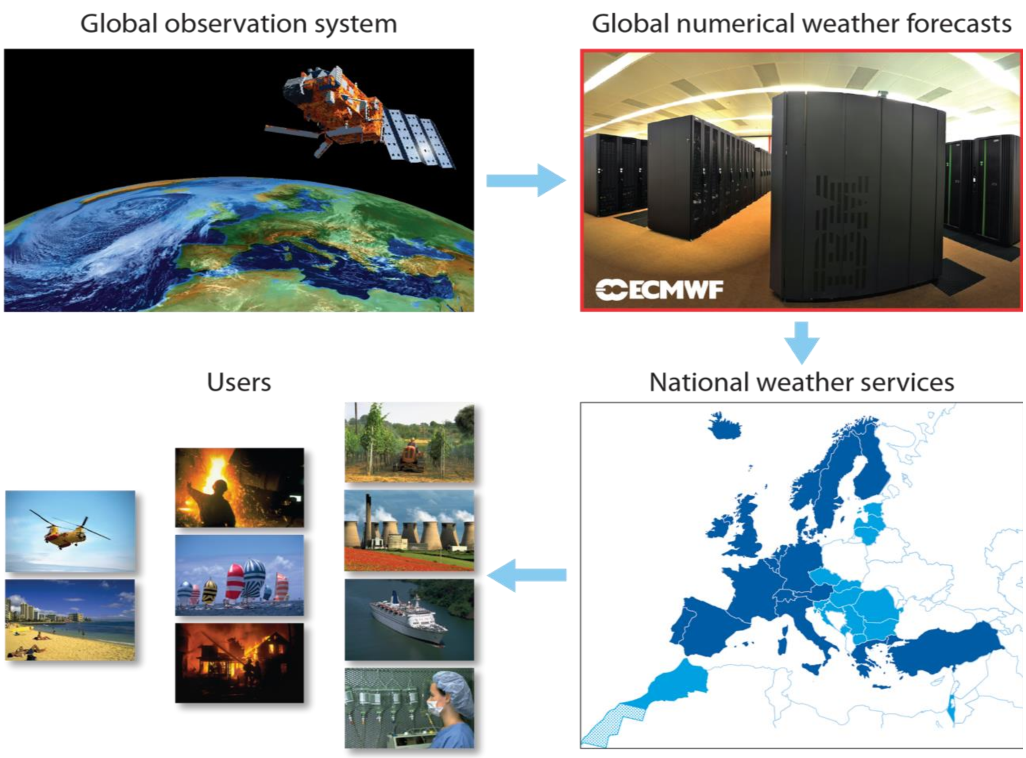


Figure 12: ECMWF Workflow

Figure 12 specifies the workflow of ECMWF. The ECMWF model outputs global weather forecast fields. Member states and clients very often require specific tailored products, e.g. temperature in the whole country of Germany or precipitation on a coarse latitude-longitude grid. This means that from the original model fields, ECMWF are required to transform, interpolate, cut-out or rotate the data to suite the users. This is achieved by a subsequent system called Product Generation, which also operates within the 1-hour time-critical operational window. This system reads roughly 70% of the original model output and generates today roughly 20 TB of products daily.

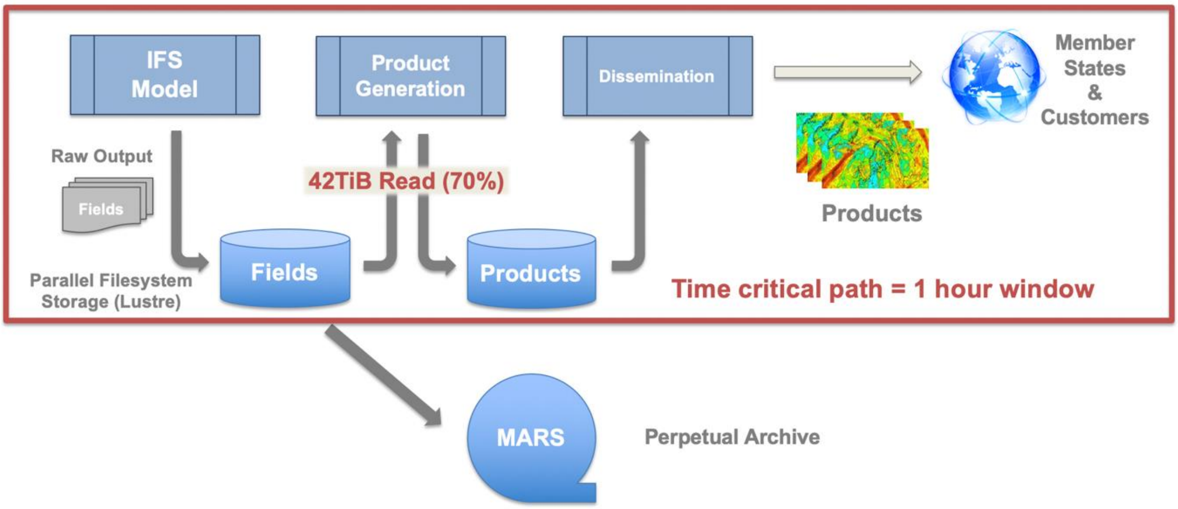


Figure 13: ECMWF Weather Forecast Model

4.3.2 User services and support

ECMWF has its own User Support team looking after all users registered onto ECMWF's system and also providing support for users outside ECMWF which use ECMWF products, services and software. Users have with the Service Desk one contact point for advice, enquiries, problem reporting. The Services Desk can be reached using the email address

servicedesk@ecmwf.int

and every report will be added to our JIRA tracking system to be followed up by support staff or relevant analysts. ECMWF is a 24/7 operational centre and operators will monitor the reports to the Service Desk during the night and on weekends.

Users are encouraged to consult the online user support portal at

https://confluence.ecmwf.int/display/UDOC/User+Documentation

when using the systems and data from ECMWF.

Support for ECMWF developed software packages can be found on the Software Support portal at

https://confluence.ecmwf.int/display/SUP/Home

The Copernicus services operated on the behalf of the European Commission at ECMWF have their own user support team. This user support team specialises in the support of users wanting to use Copernicus data and services. The support team provides also a knowledge base on Copernicus data and services at

https://confluence.ecmwf.int/display/CKB/Copernicus+Knowledge+Base

Both support teams will be involved in supporting partners in the HiDALGO projects. The ECMWF User Support will help users to register and work with ECMWF services, while the Copernicus User Support will help with the use of Copernicus data in the use cases of the project.

5 PRACE – Partnership for Advanced Computing in Europe

All previous sections have described the details about hardware, software and services of the HiDALGO infrastructure at HLRS, PSNC and ECMWF. This section will focus on PRACE – Partnership for Advanced Computing in Europe, which highlights the cooperation between several supercomputing centres in order to provide a pan-European supercomputing infrastructure for strengthening the European users of HPC resources [19].

The Partnership for Advanced Computing in Europe was started with a mission to provide access to best in class supercomputing and data management resources and services for scientific as well as engineering research and development across Europe. It was established as an international non-profit organization in Brussels with representatives from 26 member countries.

PRACE includes five main members: BSC in Spain [14], CINECA in Italy [15], GENCI in France [16], ETH Zurich/CSCS representing Switzerland [17] and the Gauss Centre for Supercomputing in Germany [18] that provide access to their systems through PRACE. HLRS is one of the hosting member of PRACE and thus offers direct access to PRACE Research Infrastructure. Many of the associated members of PRACE do not only provide access to their HPC infrastructures, but also provide training and education service via seasonal workshops and seminars for effective use of the PRACE Research Infrastructure.

Access to PRACE resources is only available for academic or industrial research and development under certain conditions that are specific to the PRACE HPC centres. To access the PRACE HPC resources, an application has to be submitted to a dedicated PRACE call, which are announced on the PRACE website [19]. Section 7.4 contains a detailed explanation of the PRACE resource access process.

Following is the list of resources granted by PRACE for the Centres of Excellence. This is equal to 0.5 % of the operational capacity of all the systems. HiDALGO plans to make use of those resources in order to extend the local capacity available at HLRS, PSNC and ECMWF, but also to ensure code portability, profiling and benchmarking for various architectures and systems.

Cluster Name	Resource Available
MareNostrum 4	825.000 core hours
Marconi KNL	4.660.000 core hours
Marconi Broadwell	270.000 core hours
Juwels	495.000 core hours
HazelHen	475.000 core hours
SuperMUC	845.000 core hours

Cluster Name	Resource Available
Joliot-Curie KNL	470.000 core hours
Joliot-Curie SKL	660.000 core hours
Piz Daint	2.176.000 core hours and 32.000 node hours

Table 23: PRACE resources available for the Centres of Excellence

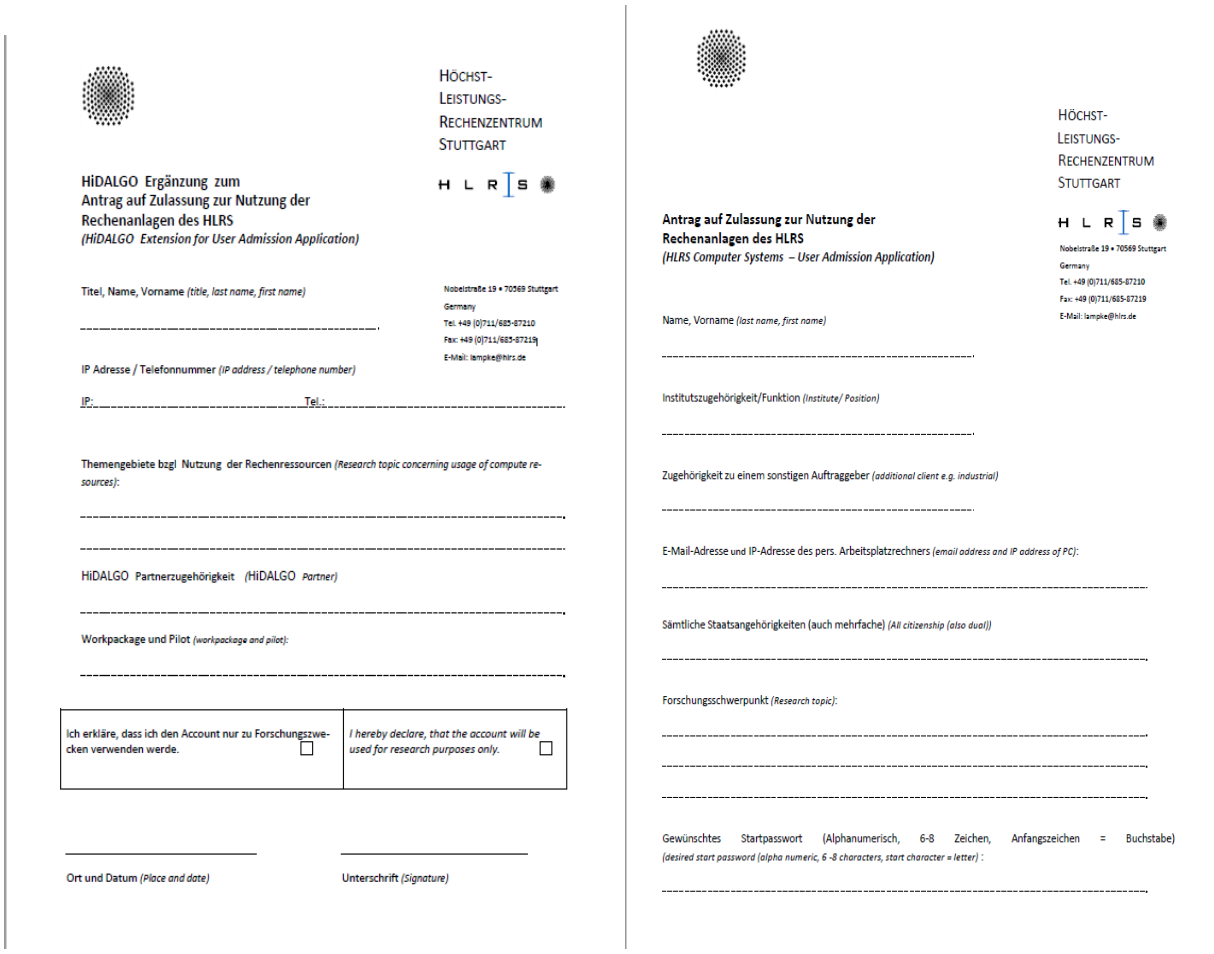
6 Getting Access

This section describes the process of getting access to the available systems at HLRS, PSNC and ECMWF, which are directly available to the HiDALGO project. All three centres have distinct processes for accessing the resources that is explained thoroughly in the sections below. Moreover, the details about accessing the PRACE Research Infrastructure is also given in this section.

6.1 HLRS

For getting the access to the systems at HLRS, a user has to go through a complete manual process. All users who want to access the systems have to provide the two filled forms as shown in Figure 14 along with their signatures and company stamps to HLRS. However, to simplify the process for the consortium partners, these documents have been uploaded to the following location

https://repository.atosresearch.eu/repository/index.php/apps/files?dir=%2FHiDALGO%2FWP5%2FDeliverables%2FD5.1.



HÖCHST-
LEISTUNGS-
RECHENZENTRUM
STUTTGART
H L R | S

Nobelstraße 19 • 70569 Stuttgart
Germany
Tel. +49 (0)711/685-87210
Fax: +49 (0)711/685-87219
E-Mail: lampke@hlrs.de

HiDALGO Ergänzung zum Antrag auf Zulassung zur Nutzung der Rechenanlagen des HLRS
(HiDALGO Extension for User Admission Application)

Titel, Name, Vorname *(title, last name, first name)*

IP Adresse / Telefonnummer *(IP address / telephone number)*

IP: Tel.:

Themengebiete bzgl Nutzung der Rechenressourcen *(Research topic concerning usage of compute resources)*:

HiDALGO Partnerzugehörigkeit *(HiDALGO Partner)*

Workpackage und Pilot *(workpackage and pilot)*:

Ich erkläre, dass ich den Account nur zu Forschungszwecken verwenden werde. ☐	*I hereby declare, that the account will be used for research purposes only.* ☐

Ort und Datum *(Place and date)*

Unterschrift *(Signature)*

HÖCHST-
LEISTUNGS-
RECHENZENTRUM
STUTTGART
H L R | S

Nobelstraße 19 • 70569 Stuttgart
Germany
Tel. +49 (0)711/685-87210
Fax: +49 (0)711/685-87219
E-Mail: lampke@hlrs.de

Antrag auf Zulassung zur Nutzung der Rechenanlagen des HLRS
(HLRS Computer Systems – User Admission Application)

Name, Vorname *(last name, first name)*

Institutszugehörigkeit/Funktion *(Institute/ Position)*

Zugehörigkeit zu einem sonstigen Auftraggeber *(additional client e.g. industrial)*

E-Mail-Adresse und IP-Adresse des pers. Arbeitsplatzrechners *(email address and IP address of PC)*:

Sämtliche Staatsangehörigkeiten (auch mehrfache) *(All citizenship (also dual))*

Forschungsschwerpunkt *(Research topic)*:

Gewünschtes Startpasswort (Alphanumerisch, 6-8 Zeichen, Anfangszeichen = Buchstabe)
(desired start password (alpha numeric, 6 -8 characters, start character = letter) :

Figure 14: HLRS user forms

The first form shown in Figure 14 needs to be filled and signed by the applicant along with the stamp of the institute to which the applicant belongs. The second form shown in Figure 14 requires some additional information regarding the application that will be executed.

The documents will go through an official approval process after its submission at HLRS. After the approval, a user account following the HLRS naming convention is created for the applicant along with the initial password for access. Finally, the systems at HLRS can be accessed using SSH connections via system frontend/login nodes as shown in table 24.

Systems	Frontend/login nodes	Special Information
Hazelhen	hazelhen.hww.de	It uses a DNS round robin for load balancing.
Kabuki	kabuki.hww.de	
Vulcan	vulcan.hww.de	It uses a DNS round robin for load balancing.
Urika	gilgamesh-login1.hlrs.de enkidu-login1.hlrs.de	

Table 24: HLRS systems and access

No additional graphical user interface or application interface have been provided, the user has to use a secure shell connection to access the systems. Consequently, all HLRS systems can be accessed only via SSH connections as shown in the example below.

```
user@host:~$ ssh hlrsusername@hazelhen.hww.de
user@host:~$ ssh hlrsusername@vulcan.hww.de
user@host:~$ ssh hlrsusername@kabuki.hww.de
user@host:~$ ssh hlrsusername@gilgamesh-login1.hlrs.de
user@host:~$ ssh hlrsusername@enkidu-login1.hlrs.de
```

6.2 PSNC

A unified process for requesting HPC resources has been defined and implemented at PSNC. The domestic user, in person of the 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 the 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, 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: https://hpc.man.poznan.pl/ (see Figure 15). The system is responsible for:

1. Gathering the application forms data.
2. Management of the groups of users associated within 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.

The portal is developed using Open Source software, the codes are still under development and include continuous updates. Finally, the systems can be accessed via SSH:

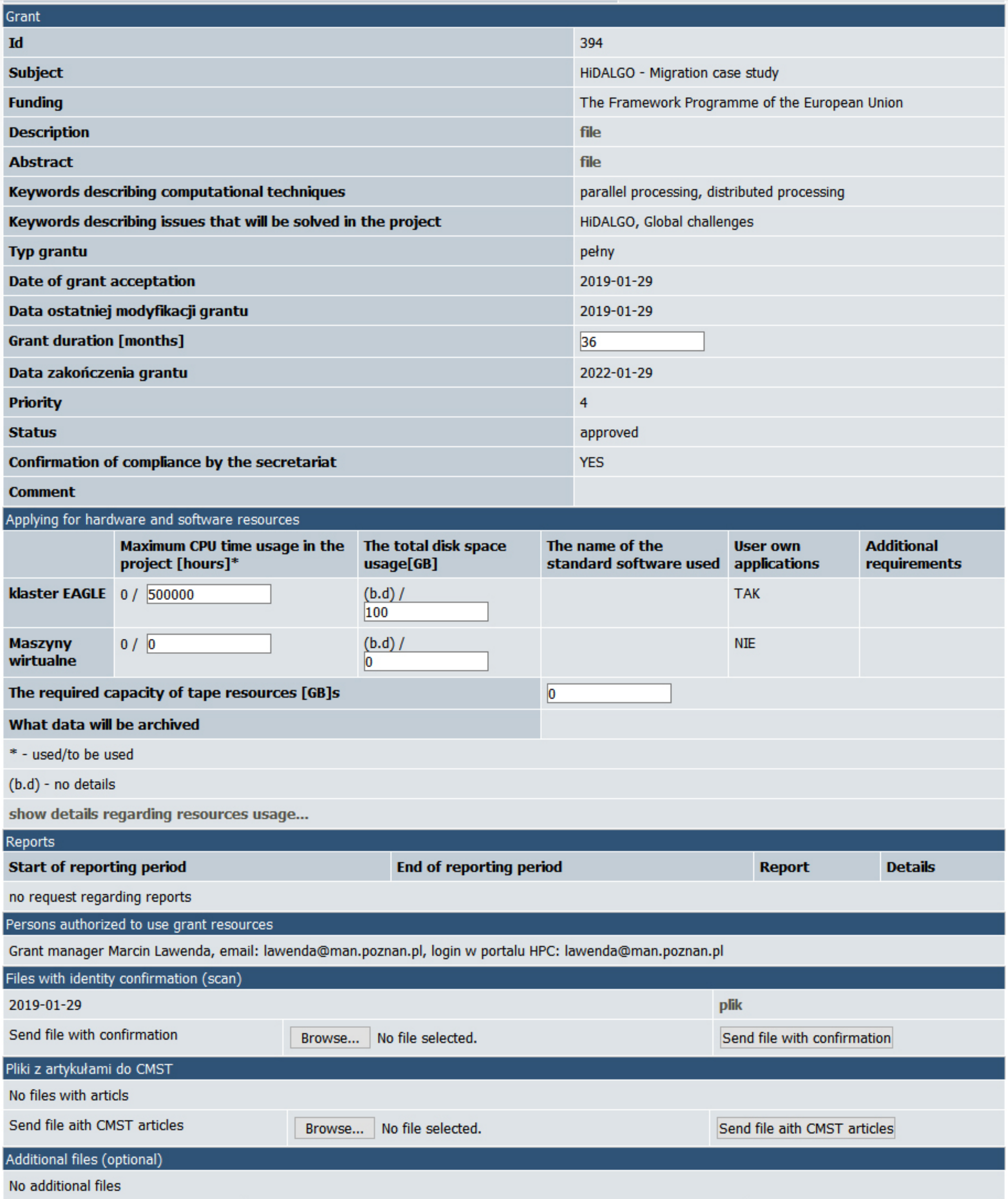
```
user@host:~$ ssh psncusername@eagle.man.poznan.pl
```

As it was already mentioned, the work on EAGLE is organized around computational grants. To facilitate the project work, four of them have been established:

- HiDMiG "HiDALGO - Migration case study" - Grant 32542
- HiDAP "HiDALGO - Urban Pollution case study" - Grant 32543
- HiDSN "HiDALGO - Social Networks case study" - Grant 32544
- HiDPrep "HiDALGO – Test and preparation case study" - Grant 32545

Of course, it illustrates only logical organization which follows pilots' oriented approach in the HiDALGO project. Other applications and cases, which are under investigation (e.g. Artificial Intelligence, data analytics, pre- or post-processing, other supporting tools) will be assigned based on either technical coupling or balancing reasons.

In order to use EAGLE resources, each user must have an account registered at the HPC portal: https://hpc.man.poznan.pl. The portal provides overall information about computational grants (validity, limits, machine allocation), consumed resources and publications presented results of computations. Each user account is assigned to the specific computational grant as well as HPC cluster. Moreover, portal allows to set up a password for machine.



Grant	
Id	394
Subject	HiDALGO - Migration case study
Funding	The Framework Programme of the European Union
Description	file
Abstract	file
Keywords describing computational techniques	parallel processing, distributed processing
Keywords describing issues that will be solved in the project	HiDALGO, Global challenges
Typ grantu	pełny
Date of grant acceptation	2019-01-29
Data ostatniej modyfikacji grantu	2019-01-29
Grant duration [months]	36
Data zakończenia grantu	2022-01-29
Priority	4
Status	approved
Confirmation of compliance by the secretariat	YES
Comment	

Applying for hardware and software resources

	Maximum CPU time usage in the project [hours]*	The total disk space usage[GB]	The name of the standard software used	User own applications	Additional requirements
klaster EAGLE	0 / 500000	(b.d) / 100		TAK	
Maszyny wirtualne	0 / 0	(b.d) / 0		NIE	

The required capacity of tape resources [GB]s	0
What data will be archived	

* - used/to be used

(b.d) - no details

show details regarding resources usage...

Reports

Start of reporting period	End of reporting period	Report	Details
no request regarding reports			

Persons authorized to use grant resources

Grant manager Marcin Lawenda, email: lawenda@man.poznan.pl, login w portalu HPC: lawenda@man.poznan.pl

Files with identity confirmation (scan)

2019-01-29		plik
Send file with confirmation	Browse... No file selected.	Send file with confirmation

Pliki z artykułami do CMST

No files with articls		
Send file aith CMST articles	Browse... No file selected.	Send file aith CMST articles

Additional files (optional)

No additional files

Figure 15: Information about HiDALGO computational grant at PSNC HPC portal

Access to the Virtual Machines is granted based on RSA public keys of the user. Operations of the user account creation and importing 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.

6.3 ECMWF

To get access to any ECMWF system an application has to be filed with ECMWF Service Desk or any Computer Representative of ECMWF's Member & Co-operative States. The access is normally not for HPC resources since this would require a special project to be set up.

As from the beginning of 2019, a new system will be put into place to allow access to the cloud environments. The new way should be more automated and less bureaucratic. Partners in the HiDALGO project will be one of the first users able to use this new system. The access will be very similar to what is offered already for the WEkEO system on

https://www.wekeo.eu/documentation/using_virtual_machines

6.4 PRACE

In this subsection, a detailed explanation for accessing the PRACE Research Infrastructure is provided. PRACE systems are available to scientists and researchers from academia and industry from around the world through two forms of access:

- **Preparatory Access**: This is intended for short term access to resources for code-enabling and porting which is required to prepare proposals for Project Access and to demonstrate the scalability of codes. Applications for Preparatory Access are accepted at any time, with a cut-off date every three months and they only undergo technical review.
- **Project Access**: This is multi-year access (up to three years) intended for individual researchers or research groups for production runs. After the submission, the application for system access will go through a scientific and technical review, which is the so called PRACE Peer Review Process that is carried out by technical experts and leading scientists. They evaluate the proposals submitted in response to the bi-annual calls that are announced officially. More information with respect to the PRACE call procedure can found here http://www.prace-ri.eu/call-announcements/.

As described in the previous section, an application has to be submitted to access the PRACE HPC resources using an online application form that can be accessed here: https://prace-peer-review.cines.fr/. The applicants are required to fill the form, which includes details about the application environment, required computing resources, but also services [19].

7 Conclusions

This document represents the first official deliverable of work package 5 of the HiDALGO project. The work that have been performed in the first three months of the project life cycle is listed and special consideration is given to the hardware and software infrastructure including the services that are available and can be used within the project.

The document contains a brief introduction about the HiDALGO project and highlights different aspects of the available infrastructure. In the first section, the hardware infrastructure of the three supercomputing centres HLRS, PSNC and ECMWF is described. The second section highlights the available software infrastructure including software packages, compilers and libraries installed on the available clusters. Also a description on how to install additional software packages on the computing systems is provided. Moreover, the services such as training and user support that are already present at HLRS, PSNC and ECMWF are described in detail as well. This document also covers the research infrastructure provided by the Partnership for Advanced Computing in Europe in order to offer support in terms of physical resources to the HiDALGO project.

Finally, for all kind of users, the process for accessing the systems at all three supercomputing centres is described by providing access forms for HLRS, a description of the PSNC web portal including the process of accessing the PRACE Research Infrastructure. The objective of providing these access mechanism is to create a baseline for all the users in the HiDALGO project.

References

[1] CoeGSS. http://coegss.eu/. Last visited in February 2019.

[2] Grant Agreement No. 824115: HiDALGO. Report. European Commission 2018.

[3] HPCG Benchmark. http://www.hpcgbenchmark.org/custom/index.html?lid=155&slid=297. Last visited in January 2019.

[4] LINPACK Benchmark. https://www.top500.org/list/2018/11/?page=1. Last visited in January 2019.

[5] IBM TS3500 Tape Library. https://www.ibm.com/de-en/marketplace/ts3500. Last visited in February 2019.

[6] "HPSS Introduction – HLRS Platforms". https://wickie.hlrs.de/platforms/index.php/HPSS_Introduction. Last visited in February 2019.

[7] HLRS Platforms. https://wickie.hlrs.de/platforms/index.php/Platforms. Last visited in January 2019.

[8] Adaptive Computing LTD, TORQUE documentation. http://docs.adaptivecomputing.com/torque/4-0-2/Content/topics/12-appendices/commandsOverview.htm. Last visited in January 2019.

[9] Adaptive Computing LTD, MAOB HPC Suite Documentation. http://www.adaptivecomputing.com/support/documentation-index/moab-hpc-suite-documentation/. Last visited in February 2019.

[10] GCC, the GNU Compiler Collection. https://gcc.gnu.org/. Last visited in February 2019.

[11] SLURM Workload Manager. https://slurm.schedmd.com/. Last visited in February 2019.

[12] Intel® C++ Compiler 19.0 Developer Guide and Reference. https://software.intel.com/en-us/cpp-compiler-developer-guide-and-reference. Last visited in February 2019.

[13] HLRS Solutions & Services. https://www.hlrs.de/solutions-services/. Last visited in February 2019.

[14] Barcelona Supercomputing Centre. https://www.bsc.es/. Last visited in February 2019.

[15] CINECA. https://www.cineca.it/en. Last visited in February 2019.

[16] GENCI. http://www.genci.fr/en. Last visited in February 2019.

[17] CSCS. https://www.cscs.ch/. Last visited in February 2019.

[18] GCS - Gauss Centre for Supercomputing. http://www.gauss-centre.eu/gauss-centre/EN/Home/home_node.html. Last visited in February 2019.

[19] PRACE Research Infrastructure. http://www.prace-ri.eu/. Last visited in February 2019.

[20] M. Gienger, N. Meyer, S. Petruczynik, R. Januszewski, A. Cheptsov, B. Koller. D5.1 – Definition of the CoeGSS Operation Environment. CoeGSS. 2015.

[21] M. Gienger, S. Gogolenko, A. Geiges, D. Kaliszan, S. Petruczynik, R. Januszewski, P. Wolniewicz. D5.8 – Second report on provided testbed components for running services and pilots. CoeGSS. 2018.

Annexes

(Annexure–I)

PSNC User Service and Support

A. *Help Desk Solution*

PSNC has supported users with the utility of achieving fast response on their problems coming from distinct areas of computational infrastructure for years. The Network Operation Centre (NOC) is the “Operator on Duty” on a 24/7 basis. It reacts in emergency situations and can provide the tentative support nearly on-line. However, all user queries and problems must be registered in the PSNC Ticketing System (TTS).

Every ticket generated by the system is persistent or has a "history" showing what happened to the ticket within its life cycle. TTS has the ability to merge multiple requests about the same incident, thus making it possible to work on an incident rather than on singular requests. TTS is a multiuser system which means that multiple agents may work simultaneously on the tickets in the system, reading the incoming messages, bringing them in order, and answering them.

The PSNC Security Department also uses TTS for the internal PSNC purposes concerning configuration of own security systems (firewall policies etc.), internal security audits and deployment of new network segments. The procedure is simplified in that case (as it usually involves a smaller number of administrators and security experts within PSNC), but in principle is very similar to what has been described above.

B. *Training Centre*

Since 2005, PSNC has organized training sessions for the HPC users coming from different research areas. During approximately 100 trainings, 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.

The ordinary 1-day training session consists of two parts: a theoretical introduction and a hands-on part. During practical sessions, the participants can run the exercises prepared by the tutor team to get deeper insights. This efficient process is enabled by training accounts with full online access to the PSNC infrastructure.

PSNC also provides multi-day schools devoted to certain HPC paradigms or applications. Specialists of well-known vendors are invited to conduct the thorough lectures and exercises on Intel technologies that are widely uses at PSNC.

C. 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 users having HPC accounts in PSNC.

End User Support corresponds to the service for the user application enhancement in a broader sense. Mainly it consists of improving their codes in the following aspects:

- Scalability,
- Optimisation,
- IO management,
- Optimizing the utilization of the precompiled math libraries and
- Numerical algorithms.

The permanent support has been addressed to all users having HPC accounts in PSNC who:

- Write their own code (C, C++, Fortran languages),
- Port codes for different computer architectures,
- Shall compile / recompile the provided software,
- Test the performance of compiled codes,
- Use tools supporting the programmers in the creation and optimization and
- Have problems with running third-party applications.

D. User support on security

The user becomes a critical link in the security chain of every system or infrastructure. Even an optimally protected infrastructure may be endangered if its users tend to behave in an inaccurate way.

Thus, it is proposed that activities be undertaken focused on supporting different groups of users involved in the HPC Fusion infrastructure in the security area. The majority of the activities will be performed during the whole project lifetime. They will be focused also on users of the infrastructure who may not have Information Technology (IT) security skills, or even lack on general domain knowledge.

1. Security trainings for the common users

A series of security trainings will be prepared and conducted. The trainings will be prepared for the common users of the project infrastructure and, therefore, the contents will be much less technically oriented and provided in a way that is easy to be understood by a person who is not an IT specialist (and especially an IT security expert).

Generic topics of such trainings could include, among others:

- Guide to secure use of the project infrastructure
- Hints on how to create and remember complex passwords
- Secure configuration of the client host connected to the project infrastructure

- Warnings how to combat social attacks (including phishing)
- What to do after the project user suspects to be a victim of a cyber-attack

In this particular case it seems infeasible to conduct a survey amongst the common users. It should be security experts who decide the relevant topics on a user request basis. If necessary, individual users can always ask the Security Consultancy Service (see below) for the further knowledge.

This approach should provide every HPC project user sufficient knowledge on how to prepare or use the project infrastructure. The presentations will also contain advices, what to do in the case of a particular problem that may be security oriented (e.g. ask for help the Consultancy Service or the CERT service, described below).

2. Security Consultancy Service

It will be assured that a group of experts, experienced in IT security in general and in assuring security of HPC research infrastructures in particular, will be able to provide reactive and proactive activities to the HiDALGO project during the whole project lifetime.

Reactive activities will include:

- Responding to security related questions from any user of the project infrastructure (e.g. common users, administrators of the infrastructure, developers producing software components in the project).
- Providing consultancy and expert knowledge for the purposes of security trainings and providing training materials.
- Other on demand activities like cooperating with the project CERT in cases where e.g. forensics are necessary after a security incident.

Proactive activities will include:

- The service will additionally identify (and update if required) the list of the technologies involved in the project.
- Issuing (possibly with the cooperation of the project CERT) notifications and newsletters regarding particular security issues. While the newsletters are intended to be regular (e.g. monthly or quarterly), the notifications may be issued after an unexpected, important event (e.g. detecting dangerous phishing email sent to the project community – in which case the notification would be sent to the whole community, or issuing a critical patch to the used software component, in which case the notification would be directed only to the administrators of the project infrastructure).

The CERT team offers support for end users of the infrastructure. The users will be able to submit information about potential security incidents that occurred in the project infrastructure.