HiDALGO

D5.5 Innovative HPC Trends and the HiDALGO Benchmarks

Document Identification			
Status	Final	**Due Date**	31/05/2020
Version	1.0	**Submission Date**	19/06/2020

Related WP	WP5	**Document Reference**	D5.5
Related Deliverable(s)	D3.1, D3.2, D5.1, D5.2, D5.3	**Dissemination Level (*)**	PU
Lead Participant	PSNC	**Lead Author**	Marcin Lawenda
Contributors	PSNC USTUTT ECMWF ICCS	**Reviewers**	Konstantinos Nikas (ICCS) Nabil Ben Said (MOON)

Keywords:
New promising technologies, HPC, benchmarks, scalability, efficiency, Exascale, Global Challenges, Global Systems Science

Document Information

List of Contributors	
Name	Partner
Dineshkumar Rajagopal	USTUTT
Sergiy Gogolenko	USTUTT
Dennis Hoppe	USTUTT
Natalie Lewandowski	USTUTT
Krzesimir Samborski	PSNC
John Hanley	ECMWF
Milana Vuckovic	ECMWF
Nikela Papadopoulou	ICCS
Petros Anastasiadis	ICCS

Document History			
Version	Date	Change editors	Changes
0.1	20/05/2020	Marcin Lawenda	TOC, Introduction
0.2	04/05/2020	Krzesimir Samborski, Marcin Lawenda	Chapter 1 and 4
0.3	04/05/2020	Nikela Papadopoulou, Petros Anastasiadis	Chapter 6
0.35	05/05/2020	Dineshkumar Rajagopal, Sergiy Gogolenko, Dennis Hoppe, Natalie Lewandowski	Chapter 2 and 3
0.36	05/05/2020	John Hanley, Milana Vuckovic	Annex 1
0.4	20/05/2020	Marcin Lawenda	Chapter 2, 3 and 4, conclusion, executive summary,

Document History			
Version	Date	Change editors	Changes
0.7	30/05/2020	Konstantinos Nikas, Nabil Ben Said	First review
0.8	05/06/2020	Marcin Lawenda	Addressing project review comments. Final version to be submitted.
0.9	17/06/2020	Marcin Lawenda	Review and approve it.
1.0	19/06/2020	F. Javier Nieto	Final approval

Quality Control		
Role	Who (Partner short name)	Approval Date
Deliverable Leader	Marcin Lawenda (PSNC)	17/06/2020
Quality Manager	Marcin Lawenda (PSNC)	19/06/2020
Project Coordinator	Francisco Javier Nieto de Santos (ATOS)	19/06/2020

Table of Contents

List of Tables

List of Figures

List of Acronyms

Abbreviation / Acronym	Description
AMD	Advanced Micro Devices, Inc.
CI/CD	Continuous Integration / Continuous Deployment
COBIT	Control Objectives for Information and Related Technologies
CRUD	Create Read Update Delete
CSI	Continuous or Continual Service Improvement
CUDA	Compute Unified Device Architecture
DB	Database
DCPMM	Data Center Persistent Memory Module
DRAM	Dynamic random-access memory
EC	European Commission
eTOM	Business Process Framework
FAQ	Frequently Asked Questions
FPGA	Field-programmable Gate Array
GASPI	Global Address Space Programming Interface
GC	Global Challenge
GLPI	French Acronym: Gestionnaire Libre de Parc Informatique
GUI	Graphical User Interface
HBM	High Bandwidth Memory
HIP	Heterogeneous-compute Interface for Portability
HPX	High Performance ParalleX
HTTPS	HyperText Transfer Protocol Secure
IFS	Integrated Foreasting System
ISACA	Information Systems Audit and Control Association
ISO	International Organization for Standardization
IT	Information Technology
ITSM	Information Technology Service Management
IDM	IDentity Manager
L1	Level-1 (basic support)
L2	Level-2 (expert support)
MOF	Microsoft Operations Framework
NVM	Non-volatile memory
NVRAM	Non-volatile random-access memory

Abbreviation / Acronym	Description
OAUTH2	Open Authorization V2
PBS	Portable Batch System
PUE	Power Usage Effectiveness
Q&A	Questions and Answers
RACI	Responsibility Accountability Consulted Informed
REST	Representational State Transfer
SAML	Security Assertion Markup Language
SIAM	Service Integration and Management
SLA	Service Level Agreement
SLURM	Simple Linux Utility for Resource Management
SME	Small- and Medium-scale Enterprise
SSL	Secure Socket Layer
STFC	Science and Technology Facilities Council
URL	Uniformed Resource Locator
VM	Virtual Machine

Executive Summary

Presently, development and optimization activities within HiDALGO project are conducted considering already available hardware and software solutions. However, computer technology constantly develops and it becomes equally important to use the recent achievements offered in this field. That drives us to the main purpose of this document, which is the analysis of information about edge technologies available on the market, which could be of interest for HIDALGO use cases.

The HiDALGO system constitutes composition of computation and data flows where number of processing ways and methods are involved (see Figure 1). Covering of all necessary facets requires comprehensive look on achievements from many computational areas. Certainly, pilots will benefit from efficient utilization of applicable hardware and software solutions and gain better possible yields of simulation and analysis tools.

Our analysis starts with system components that make up the HiDALGO infrastructure (Chapter 2). Recent findings on CPU field are discussed considering most significant vendors like Intel, AMD, ARM, IBM and Oracle. To complete the picture of processing units in the next step the focus turns on accelerators, GPGPU units, co-processors and memory technologies.

For some time, we have been observing the limit of development possibilities of HPC systems within the solutions used so far. In the consecutive subchapter, promising architectures and revolutionary approaches to processing are presented. Some of them like Massively Parallel Processor Arrays or ARM- and FPGA-based microservers are already implemented and utilized. Others (e.g. quantum computers) are in prenatal phase but with promising perspectives.

Even the latest solutions will not be effectively used if it is not accompanied by the development of appropriate software. This is considered as a part of co-design process and comprise essential phase of performance improvement. In this domain, the attention is put on tools, which enable insight into the application and evaluation of processing efficiency. Next, recent achievements in mathematical libraries are summarized along with programming paradigms based on standards like MPI or OpenMP and programming interfaces CUDA/OpenCL. This analysis would not be completed without efficient data organization and transfer methods as well as information on best workload managers (SLURM, Torque and PBS) which facilitate the process of multiple jobs management.

In order to give this report a more practical aspect, in Chapter 4 the first benchmarks on new AMD Rome are delivered. Here, simulation applications from two project use cases where investigated and compared with results achieved on already accessible processors.

It is worth to mention that in Annex 1 information about most relevant and substantial projects in the world, which also tackle Exascale challenges, is presented in concise way.

Figure 1. Pave the way to HiDALGO efficient processing.

1 Introduction

1.1 Purpose of the document

The report provides information on new promising technologies, which appear on the market and could have significant influence on the functionality and performance of HiDALGO solutions. Furthermore, the HiDALGO benchmark tests are delivered based on the available systems.

This paper introduces two-dimensional implications. In the first place, it makes an inventory on edge achievements on technology market applicable for HiDALGO workflows. Secondly, delivers practical approach in the form of benchmarking simulation tools on recently acquired computational nodes (AMD Rome).

The HIDALGO project is considered as semi-continuation of its predecessor the CoeGSS project. Therefore in some places, whenever applicable (mostly in Chapter 2), references are implied to previous similar work.

Based on this elaboration, pilot developers would be able select most promising technologies and software solutions to achieve conceivably best performance results.

1.2 Relation to other project work

Performance in computation is essential for all use cases as well as accompanied tools. That is why the knowledge collected in this report should be spread among technical work packages (WP3, W4, and WP6) including this work package (WP5) as well.

WP3, responsible for new tools implementation, optimization and data management, may especially benefit from collected information on new effects introduced by hardware and software solutions. Based on collected knowledge, solutions will be proposed that improve both application performance and data transmission.

WP4, where pilot methods are developed may widely benefit from benchmarking results and suggestions on new mathematical libraries and programming paradigms.

WP5 is responsible for setting up infrastructure for pilots. The selection of appropriate infrastructure components seems to be essential for the project success. WP5 should learn the lesson from information on edge system components, innovations in HPC architectures and finally on workload managers gains.

1.3 Structure of the document

This document is structured in five major chapters:

- **Chapter 2** describes most essential components for system efficiency like CPUs, accelerators, graphical processing units, co-processors and new memory architectures. The focal point is placed on latest achievements offered by hardware vendors. Next part of this chapter delivers information about architectures that are the hope of efficient computations in the future. The section details what is quantum computer. Next, promising solutions, available already today, are presented here, like: Massively Parallel Processor Arrays or ARM and FPGA based microservers.
- **Chapter 3** analyses various software solutions, which facilitate the process of improving application yield. It starts with tools for analysing performance and identifying vulnerabilities in their operation. Then, mathematical libraries, which are so important in HPC computing, are investigated against their capabilities of parallelization and vectorization. Next subsections focus on programming standards. The crowning of this part is the discussion on workload managers.
- **Chapter 4** presents first achievements on new architectures where a new processor AMD Rome is involved. The benchmarking is done under two different environments and for two simulation applications coming from Migration and Urban Air Pollution pilots.
- **Chapter 5** concludes this deliverable and provides information about next steps.
- In order to collect state of the art knowledge in another Exascale-related projects **Annex 1** was established. Eleven most prominent projects and their endeavour towards better performing applications are discussed.

2 System components

Global System Science (GSS) applications have to attain the best performance by leveraging the features of recent and future promising HPC hardware. The market evaluation of various HPC components is conducted against the needs of GSS applications and the affordability of HPC hardware within the project duration by HPC centres (PSNC and HLRS). This is the initial analysis to identify and define the trends of HPC components based on the current situation, and it can be updated further in the final deliverable D5.8 to provide a complete list from the different vendors for establishing a baseline to conduct GSS benchmark experiments and co-design activities.

2.1 General-purpose CPUs

GSS applications are currently developed using general-purpose CPUs and MPI based distributed programming to achieve the expected results and scalability, so selecting the best latest CPU components will impact directly the current GSS applications without requiring many changes in the codebase. Different CPU vendors (AMD, ARM, Fujitsu SPARC) are entered into the HPC server markets for improving the current architectural limitation so that comparing all of them along with the traditional HPC CPU vendors (Intel Xeon, IBM Power) will provide a complete analysis to select the best CPUs from the market for satisfying the needs of GSS applications. The comparison was conducted against the major trends like the number of cores, the number of memory channels, length of vector operations and cache memory to select the best CPUs from each vendor based on the informed decision-making.

2.1.1 Intel x86

Intel releases HPC Server processors under the brand name of Xeon, which has the capabilities to provide a higher number of core counts, memory channels and cache size to attract the HPC applications. AVX (Advanced Vector Extension) is the special instruction set supported in those processors to improve HPC applications performance by enabling vector operations in the system. Current HiDALGO HPC systems, i.e. PSNC Eagle (Xeon E5-2697 v3 and Intel Xeon E5-2682 v4) and HLRS Hazelhen (Xeon CPU E5-2680 v3), are featured by the Xeon Haswell and Broadwell microarchitecture, which is based on the old processor packaging technology (22nm and 14nm) and microarchitectural designs. Therefore, our analysis has to be conducted based on the more recent Intel processor families like Cascade Lake, Cooper Lake and Ice Lake, in order to provide a concrete idea about the trends, substantiate the best microarchitecture for benchmarking, and further analysis. Cascade Lake [1] is the latest microarchitecture available in the market during this deliverable, so it is compared against

the predecessors Skylake and summarized in Table 1 to give an overview of the comparison at the micro-architecture level.

Features	Cascade Lake	Sky Lake
Processor number	Intel Xeon platinum 9282	Intel Xeon Gold 6140
Fabrication Technology	Enhanced 14nm and multi-chip package in a die	Enhanced 14nm++
Release Date	April 2019	July 2017
Number of cores	56 cores per socket.	18 cores per socket.
Number of threads	2 threads per core	2 threads per core
Core frequency	From 2.6 GHz (base) to 3.8 GHz (Turbo)	From 2.3 GHz (base) to 3.7 GHz (Turbo)
Number of memory channels	12 DDR4 channels per socket	6 DDR4 channels per socket
Memory support	DDR4-2933MHz and 3D XPoint	DDR4-2666MHz
Memory Bandwidth	21.33 GB/s per channel for DDR4	21.33 GB/s per channel for DDR4
L1 instruction cache	32 KB/core 8-way set associative	32 KB/core 8-way set associative
L1 data cache	32 KB/core 8-way set associative	32 KB/core 8-way set associative
L2 cache	1 MB/core 16-way set associative	1 MB/core 16-way set associative
L3 cache	77 MB 11-way set associative	24.75 MB/core 11-way set associative
Advanced Vector Extension	SSE 4.2 and AVX2.0 with AVX 512 bits and VNNI (Vector Neural Network Instructions) logic on Port 0 and Port 1 as part of the	SSE 4.2 and AVX2.0 with AVX 512 bits

Features	Cascade Lake	Sky Lake
	FMAs (Fused Multiplication and Addition)	
Processor Interconnect	4 UPI	3 UPI
TDP (Thermal Design Power)	400W	140W
Input and Output	(3x16) lanes of PCIe 3.0 and additional x4 lanes PCIe 3.0 reserved exclusively for DMI (Direct Media Interface).	(3x16) lanes of PCIe 3.0

Table 1: Intel Sky Lake and Cascade Lake microarchitecture comparison

Cascade Lake is differentiating from Skylake by the following key features and it is beneficial to the GSS applications accordingly.

- Multi-chip package with UPI link (2- to 8-way connection) to support a higher number of cores in the server, so GSS applications can leverage those to improve performance and scalability.
- Twice the number of memory channels supported, so memory bandwidth is approximately twice to improve performance of memory-bound GSS applications.
- Intel Optane Pmem (3D XPoint) memory is supported, so the memory-bound GSS application can gain benefit with 3D XPoint aware programming. 3D XPoint memory bandwidth is higher than DDR4, so it will improve performance naturally for in-memory computing and memory-bound applications.
- VNNI instruction is supported in AVX 512 bits to improve convolutional neural network algorithms by accelerating inner convolution neural network loops.
- Intel Cascade Lake platinum 9282 TDP (Thermal Design Power) is 2.8 times more than the Intel Sky Lake Gold 6140, so the performance-power and performance-price ratio comparison are needed to ensure the best microarchitecture between those processors.

GSS benchmark experiments were already conducted on the Skylake and Haswell processors on the CoeGSS project [2], so we have to focus on the next-generation of Intel processors (Cascade Lake) in HiDALGO project to conclude the best processors for GSS applications. Intel Xeon Skylake Gold 6140 processor and its cluster nodes specification details are provided in the CoeGSS D5.8 deliverable [3]. Intel Xeon platinum 9282 is the high-end HPC processor from the Cascade Lake family, which is best suited for the GSS benchmark evaluation with 3D

Xpoint or DDR4 memory. Cooper Lake and Ice Lake are the future series planned to be released by mid of 2020 in the Xeon processor family based on the "enhanced 14nm++ or 10nm++" fabrication process, so they may be considered as viable candidates for further analysis in the next deliverable.

2.1.2 AMD x86

AMD has released in the last few years HPC server processors with two times performance improvement from its predecessor, which is reflected from Bulldozer to current AMD Epyc processors. Bulldozer was released by 2014 and based on the old microarchitecture and 32nm fabrication, so those processors can be ignored for further analysis. AMD Epyc design is based on the Zen microarchitecture for supporting AVX instruction set, larger cache memory and higher bandwidth to meet the needs of HPC applications. AMD Epyc Naples (7551 code) is based on the Zen microarchitecture (14nm fabrication process), which is further improved with the second-generation AMD Epyc Rome (7742 code) processors based on the Zen2 microarchitecture (7nm fabrication process with the multi-chip package) to support different HPC workloads, so both AMD Epyc processors are compared at the microarchitecture level in Table 2.

Features	AMD Epyc Rome (Zen2)	AMD Epyc Naples (Zen)
Processor number	AMD Epyc Rome 7742	AMD Epyc Naples 7551
Fabrication Technology	7nm multi-chip module package	14nm
Release Date	August 2019	June 2017
Number of cores	64	32
Number of threads	2 threads per core	2 threads per core
Core frequency	2.25 GHz to 3.4 GHz	2.0 GHz to 3.0 GHz
Number of memory channels	8	8
Memory support	DDR4-3200	DDR4-2666
Memory Bandwidth	204 GB/s	170 GB/s

Features	AMD Epyc Rome (Zen2)	AMD Epyc Naples (Zen)
L1 instruction cache	32KB/core 8-way set associative	64KB/core 4-way set associative
L1 data cache	32KB/core 8-way set associative	32KB/core 8-way set associative
L2 cache	512KB/core 8-way set associative	512KB/core 8-way set associative
L3 cache	256MB 16-way set associative	64MB 16-way set associative
Advanced Vector Extension	SSE 4.2 and AVX2.0	SSE 4.2 and AVX2.0
Processor Interconnect	Infinity fabric	Infinity fabric
TDP (Thermal Design Power)	225 W	180 W
Input and Output	Infinity fabric with x16 lanes of PCIe 4.0	Infinity fabric with x16 lanes of PCIe 3.0

Table 2: Microarchitecture level comparison between AMD Epyc Naples and Rome processors

AMD Epyc Rome has introduced various new and innovative technologies to makes it unique and better than the contemporary processors (AMD Epyc Naples, Intel Cascade Lake and Skylake processors) as mentioned below:

- Overall AMD Epyc Rome is two times better than the AMD Epyc Naples, Intel Cascade Lake and Skylake processors [4].
- FLOPS performance is 1.79 times better than the Cascade Lake and 2.12 time better than the Sky Lake [5]. FLOPS performance improvement is based on the improvements of Zen2 microarchitecture, the number of core counts, 7nm multi-die Chiplets and clock speeds frequency [4].
- I/O die is introduced as a separate 14nm Chiplet in the sockets, which contains memory controller, PCIe controllers and infinity fabric connection for remote socket access. This resolves NUMA quirk to improve locality to outperform the caveats of AMD Epyc Naples processor [4] [5].
- DDR4-3200 DIMMs are supported, so they are clocked 20% faster than DDR4-2666 and 9% faster than DDR4-2933 to improve bandwidth and latency compared to AMD

Naples, Intel Cascade and Sky lake processors. GSS memory-bound applications can directly benefit with these high bandwidth memories [4] [5].

- PCI Express fourth-generation x16 lane (PCIe x16 v4.0) is introduced to provide two times more performance than AMD Naples, Intel Cascade and Sky Lake. They also support higher bandwidth connection to InfiniBand, other fabric and storage adapters, NVMe SSDs, and in the future GPU Accelerators and FPGAs with I/O die and PCIe x16 v4.0. I/O bandwidth is overall will be increased two times to support I/O-bound applications in the HPC and GSS domain [4].
- AMD Rome achieves higher performance per watt than other Intel family of processors to manage power efficiently as shown in Figure 2 [4].
- GSS and HPC applications were written and performance-tuned for the Intel x86 Xeon based processors, so they have to be ported properly in the AMD x86 Epyc to leverage the processor's capability; this is the only caveat of new architecture [4].

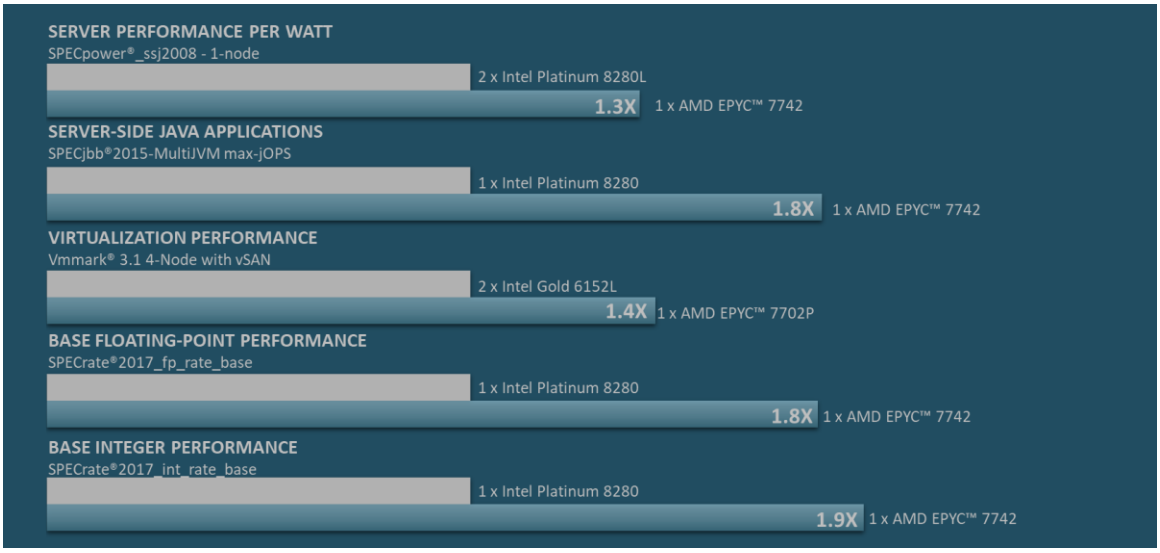

Figure 2: SPEC rate to compare the performance per watt for AMD Epyc Rome 7742, 7702P and Intel Cascade Lake processors Platinum 8280 and Gold 6152L

PSNC offered HPC nodes with AMD Naples 7551 processor for benchmarking GSS applications in the CoeGSS project, which is detailed in the D5.8 [3] CoeGSS deliverable and the same nodes would be used for the current benchmarking also. HLRS provides AMD Rome 7742 processor through its flagship Hawk HPC system, which is detailed in Table 3. GSS HPC applications can leverage the capabilities of the Hawk system to ensure performance and scalability achieved with the new AMD Rome processors and compare it with AMD Naples, as well as Intel processors to select the best processor from the x86 architecture.

Nodes	Cores	Memory	Interconnect	Amount of Storage
5632	720,896	~1.44 PB	Enhanced 9D-Hypercube	~25 PB

Table 3: HLRS Hawk system specification.

AMD plans to release Epyc Milan processors based on Zen 3 (7nm++) microarchitecture by end of 2020, so they have to be considered for further analysis in the next deliverable to extend the list of x86 processors for benchmarking and select the best processor from the x86 family.

2.1.3 ARM

ARM-based server processors have recently arrived in the HPC market, as an alternative to the traditional x86 server architecture and continuously provide better performance-price and performance-power ratio to reduce the processor cost and operational cost with RISC (Reduced Instruction Set Computing) architecture. Huawei is continuously supporting the HPC ARM-based software-hardware ecosystem in the brand name of Kunpeng, which is based on their Taishan microarchitecture with a large number of cores in a single die. PSNC offered Knupeng 916 server CPU (Hi1616) to the CoeGSS project for GSS application evaluation, which is detailed in D5.8 [3] CoeGSS deliverable. Kunpeng 920 (Hi1620) server CPUs were introduced in the HPC market to resolve the limitations in the Knupeng 916 server CPUs and provide a competitive performance-power ratio to the traditional HPC processors. Kunpeng based HPC ARM servers are supporting RAS (Reliability Accessibility Serviceability) extension to ensure the production level HPC application execution with its mature software and hardware HPC ecosystem to reduce job execution failures. GSS application is easily ported to the Kunpeng 916 with its available HPC software stack, so the application portability for new Kunpeng 920 is expected to be similar or with minimal effort than its predecessor based on our experience in the CoeGSS project. Kunpeng 920 and Kunpeng 916 processor are compared in Table 4 to point out the microarchitectural changes.

Features	Kunpeng 920 (Hi1620)	Kunpeng 916 (Hi1616)
Fabrication Technology	7 nm HPC process based on the TaiSHan v110 microarchitecture	16 nm process based on the Cortex-A72 microarchitecture
ARM Instruction set	ARM v8.2	ARM v8.0
Release Date	January 2019	August 2017
Number of cores	64	32
Number of threads	1 thread per core	1 thread per core

Features	Kunpeng 920 (Hi1620)	Kunpeng 916 (Hi1616)
Core frequency	2.6 GHz base frequency	2.4 GHz base frequency
Number of memory channels	8	4
Memory support	DDR4-2933	DDR4-2400
Memory Bandwidth	190.7 GiB/s	71.53 GiB/s
L1 instruction cache	64 KB per core 8-way set associative	48 KB per core 8-way set associative
L1 data cache	64 KB per core 8-way set associative	48 KB per core 8-way set associative
L2 cache	32 MB 8-way set associative	8 MB 8-way set associative
L3 cache	64 MiB per socket 8-way set associative	32 MiB per socket & 16-way set associative
Advanced Vector Extension	128 bit NEON advanced SIMD Extension	128 bit NEON advanced SIMD Extension
Processor Interconnect	4-way SMP to support 4 sockets per node	2-way SMP to support 2 sockets per node
TDP (Thermal Design Power)	195W	85W
Input and Output	PCIe gen 4.0 x18, x8, x4	PCIe gen 3.0 x18, x8, x4

Table 4: Huawei Kunpeng 920 and Kunpeng 916 processors comparison at the microarchitecture level.

Kunpeng 920 server processor offers 64 cores, a large number of memory channels with fast memory DIMM from DDR4, PCIe gen v4.0 and high 2.6 GHz base frequency to provide a more or less similar configuration to AMD EPYC Rome 7742. Kunpeg 920 is claimed to be better than Intel Sky Lake in terms of performance and energy efficiency as shown in Figure 3. Kunpeng 920 SIMD Neon can speed up applications such as computer vision, HPC and deep learning with its single- and double-precision floating-point operations, so it can be leveraged by GSS applications to improve the performance overall. ARM processors are designed with the motive of power efficiency, so ARM might give better performance per watt ratio than the other families of processors.

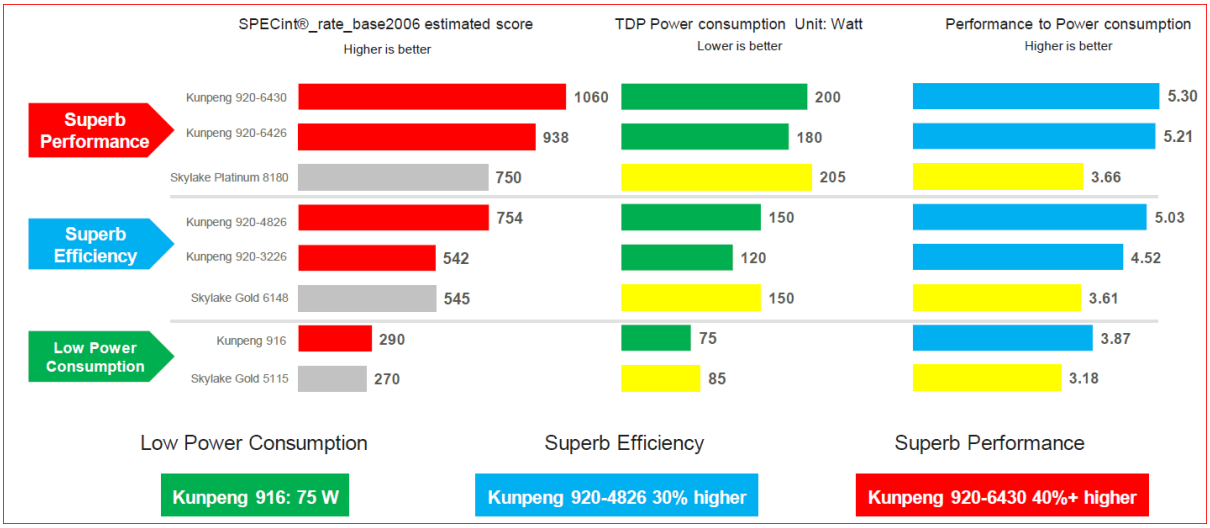

Figure 3: Huwaei Kunpeng 920 processors performance is compared with the Intel Sky Lake. Kunpeng 920 is better than Sky Lake in terms of performance and energy efficiency.

Fujitsu, Marvell and Ampere are also planning to release HPC ARM servers within 2020, so we have to take into account Fujitsu A64FX, Ampere Altra and Marvells ThunderX3 for further analysis in the next deliverable. Fujitsu A64FX is the ARM processor with 42 cores and 4 HBM2 memory channel to support high bandwidth of 1TB/s to support the post-K supercomputer. Ampere Altra is designed with 7nm process to support 80 cores in a socket, ARM v8.2+ instruction set, DDR4-3200, 3GHz base frequency and PCIe gen 4.0 to support all sort of HPC application needs. Marvells ThunderX3 is planned to be released by end of 2020, and it is based on the 7nm fabrication process to support 96 ARMV8.3+ cores with 3GHz base frequency, four threads per core, 8 memory channels with DDR4-3200, 64 lanes PCIe 4.0 and four 128-bit SIMD (Neon) unit, so we have to compare and select the best ARM-based HPC nodes in the next deliverables by analysing their HPC supports at the hardware and software level.

2.1.4 POWER – OpenPOWER

POWER (stands for Performance Optimization With Enhanced RISC) is high performance microprocessor based on the RISC (Reduced Instruction Set Computer) architecture designed by OpenPOWER Foundation (led by IBM company). The names of the next generations of processors end with consecutive numbers e.g. POWER7, POWER 8, and POWER9. The later generations utilize Power ISA (Instruction Set Architecture), an abstract model of a computer, which realization like CPU is called an implementation.

Currently, processor POWER9 is offered in two configurations: one for single and dual sockets (Scale out variant – SO) and one for four or more sockets (Scale up variant – SU) employed in NUMA servers, where larger amount of shared memory may be served. According to the

specification, the POWER9 processor is composed of 8 billion transistors and has up to 24 cores. Moreover, it is manufactured using 14nm FinFET technology, supports PCI Gen4 and has a 120MB shared L3 cache. Power9 is capable to hold 8-way simultaneous multithreading and up to 230GB/sec memory bandwidth, which results to much better performance. According to IBM, Power9 offers much better performance compared to Intel Xeon SP (x86) especially in terms of performance per core (2x), RAM per socket (2.6x), and memory per bandwidth (1.8x). Power9 using NVLink is able to achieve 9.5x better CPU to accelerator bandwidth than Intel Xeon x86.

Features	POWER8	POWER9
Fabrication Technology	22nm process	14nm process
Release Date	June 2014	2017
Number of cores	6 or 12	12 SMT8 cores or 24 SMT4 cores on die
Number of threads	8 threads per core	8 threads per core or 4 threads per core
Core frequency	2.5 GHz to 5 GHz	4.0 GHz
Number of memory channels	32	32
Memory support	DDR3 or DDR4	DDR3 or DDR4
Memory Bandwidth	204 GB/s	170 GB/s
L1 instruction cache	64+32 KB per core	32+32 KB per core
L2 cache	512 KB per core	512 KB per core
L3 cache	8 MB per chiplet	120 MB per chip
Advanced Vector Extension	VSX (Vector-Scalar Instructions)	VSX + VMX (Vector Media Extension)

Features	POWER8	POWER9
Processor Interconnect	OpenCAPI	OpenCAPI
Peak bandwidth	96 GB/s	192 GB/s

The next one in the line is POWER10 (to be launched in 2021) designed to tackle future demands in the analytics and big data domains. This processor will be manufactured in the 7nm process and offer up to 48 cores. Moreover, new memory controllers will be designed and I/O supporting OpenCAPI 4.0 [6] and NVLink3 [7] technology. Utilization of the new instruction set specification Power ISA v3.1 [8] will enable new functionality to SIMD [9] and VSX [10] instructions.

2.1.5 SPARC

SPARC stands for Scalable Processor ARChitecture. This is the architecture of RISC microprocessors developed by the Sun Microsystems and Fujitsu organization later formed into SPARC International (responsible for licensing and promoting the architecture and managing trademarks). Processors developed based on the SPARC architecture are widely used in high-performance servers, workstations, as well as embedded systems.

There are number of versions and implementations of SPARC processors [11]. There are also four main SPARC vendors: Fujitsu, Gaisler Research AB, Oracle Corporation (previously Sun Microsystems) and Texas Instruments. Since in September 2017 Oracle dissolved its development group, and now closely cooperate with Fujitsu, which become a major SPARC processor vendor. It can be noted that Supercomputer Tianhe-2, which was ranked on no. 1 of TOP500 list in 2014 is equipped with number of nodes with Galaxy FT-1500 [12] OpenSPARC-based processors.

In general, SPARC architecture defines different microprocessor software models 32-bits for SPARC version 8 and 64-bits for Fujitsu SPARC version 9. SPARC-v9 is commonly used architecture in many HPC systems (SPARC64 XIfx processor). As in many other solutions, this processor contains two types of registers: general purpose registers (integer and floating point registers) and status / control registers (program counter, processor status, trap base address and many other registers). What is very specific in SPARC-v9 is that it has multiple special instructions for conditional data transfer, forcing the initial download of data (prefetch), loading data that do not generate exceptions (nonfaulted-load), and delayed jumps.

SPARC64 XIfx processor, released in August 2014, is foreseen, as groundwork for Exascale, for massive parallel supercomputer systems to deliver utmost performance for real applications. Comparing to Sparc64 predecessors it offers major technological changes in the instruction set architecture, microarchitecture, memory modules, and embedded interconnect.

SPARC64-XII processor was released in April 2017 and is offered for high-performance servers, runs at speeds of up to 4.35 GHz. It offers significant hardware and software improvements like faster memory capability, increased on-board LAN bandwidth and more and better options for PCI connections which ultimately translated into a 2.3-2.9 times improvement in core performance over the previous-generation (SPARC64 X+).

Features	SPARC64 XII	SPARC64 XIfx
Fabrication Technology	20nm process	20nm process
Release Date	April 2017	August 2014
Number of cores	12	34 (32 compute cores 2 assistant cores)
Number of threads	96 (8 per core)	No multithreading
Core frequency	4.25 GHz	2.2 GHz
Number of memory channels	two channels per controller	8x HMCs
Memory support	DDR4	DDR4
Memory Bandwidth	153 GB/sec	240GB/s x2 (in/out)
L1 instruction cache	64 KB instruction 64 KB data cache	64KB
L2 cache	512 KB	8MiB
L3 cache	32MB	---

Features	SPARC64 XII	SPARC64 XIfx
Processor Interconnect	High-speed interconnect, up to 25 Gbps per lane	Tofu2
Peak	417GIPS / 835GFlops	1.1TFlops

According to the Fujitsu Servers Roadmap [13] enhanced SPARC M12 servers will be delivered on 2021. The solution will be featured by SPARC64 XII processors with 1.5 bigger memory capacity and power efficiency improvements.

2.2 Accelerators

Scientific and Engineering HPC applications are already using accelerators for offloading specialised workloads to improve the performance and reduce power consumption by leveraging massive parallelism and efficient hardware utilization. GSS applications are currently developed with the focus on x86 CPUs with traditional MPI distributed programming, so they have to be improved in terms of scalability and performance by using a new set of accelerators introduced in the HPC market. GPGPU and Vector co-processors are the leading accelerator technologies in the HPC domain for optimizing the performance of compute-intensive tasks with massive parallelism of single- and double-precision floating-point operations, which will be detailed in the sub-sections with different vendors' new technologies.

2.2.1 FPGA

FPGA (Field Programmable Gate Arrays) is a specialized integrated circuit. Thanks to the programmable logic system, it may be repeatedly programmed without disassembly, after it is manufactured and installed in the target device. FPGAs are used in digital signal processing, aviation and the military, in the prototype phase of ASICs [14] and in many other fields (e.g. mission to Mars).

The big advantage of such solution is shorter design time and lower production costs especially when we deal with small series. HardCopy FPGAs, which as a matter of fact are integrated circuits with functionality corresponding to the project loaded into the FPGA, present better performance and consume less power. The main disadvantage of directly programmable gate arrays is usually lower performance compared to the corresponding specialized integrated circuits and more power consumption.

Contemporary FPGAs circuits offer immediate reprogramming by using a partial reconfiguration mechanism. This solution leads to the idea of a reconfigurable computer or reconfigurable system. Thanks to adaptation of their structure, they may better meet the challenges they are facing at the very moment.

FPGA integrated circuits are considered as parts of the idea of reconfigurable HPC systems and machine learning [15] [16]. This kind of approach is considered as advanced co-design process for specific applications or workflows. Careful analysis and implementation of such systems drive to more efficient solutions in both aspects of performance and power consumption. Due to the inherent costs of development this approach is limited to few use cases.

Looking for such use cases it can be noted that FPGAs installed in the Large Hadron Collider (LHC) at CERN are used to accelerate inferencing and sensor pre-processing workloads in search for dark matter [17]. FPGAs are used in combination with other computing resources to process massive quantities of high-energy particle physics data at extremely fast rates to find clues of the origins of the universe. This computationally intensive process requires filtering sensor data in real-time to identify novel particle substructures that could contain evidence of the existence of dark matter and other physical phenomena.

These processors are offered by number of manufacturers like Intel (previously Altera), Actel, Microchip Technology (previously Atmel), Cypress, Lattice Semiconductor, QuickLogic and Xilinx.

2.2.2 GPGPU

Scientific HPC applications (Gromacs, Ansys and OpenFoam, etc.) achieve remarkable performance improvements utilizing the massive parallelism of GPGPUs. Motivated by this, it is essential to compare the performance between CPU, GPGPU and other accelerators in order to conclude the best architecture for the market of GSS applications. NVIDIA is continuously releasing HPC GPGPU processors to support both HPC, AI and data analytics workloads by improving its microarchitecture to support a large number of CUDA and Tensor cores. NVIDIA Tesla V100 GPUs achieved better performance with Intel Xeon CPU processors and its specifications are detailed below.

- 12nm fabrication process with Voltas microarchitecture
- FLOPS performance (7 TFLOPS), CUDA single-precision cores (5120), CUDA double-precision cores (2560), Tensor cores (640), SMs (80), frequency (1.53 GHz) is higher than its predecessor NVIDIA Pascal P100 with the same TDP (300W) to achieve better performance-power ratio [18].

- 16 to 32 GB HBM2 (High Bandwidth Memory version 2) memory support with a bandwidth of 900GB/sec.
- New Tensor cores are introduced with CUDA cores to enable matrix operation by using FMAs (Fused Multiply and Additions). Tensor cores are introduced to speed up the training of neural networks and improve the performance of matrix operations.
- NVLink 2.0 is the interconnection link used between CPUs and other GPGPUs to provide better communication bandwidth than the PCIe gen interface, but it has to be evaluated in our experiments. NVLink 2.0 supports CPU mastering and cache coherence capabilities with IBM Power 9 CPU-based servers.
- Streaming Multiprocessor (SM) is 50% more energy efficient than the previous generation Pascal SM design, which enables the highest performance in single- and double-precision FLOPS performance per watt.
- Double-precision floating-point performance is 7 TFLOPS for PCIe based interconnect and 7.8 TFLOPS for NVLink 2.0 [19].
- Tesla V100 GPU server is five times better than the Intel Xeon Gold 6140 CPU server for Linpack benchmark, and it reflects in the other scientific, geoscience and engineering application also.

HLRS Cray Urika CS system is powered with the latest Voltas V100 GPUs with Intel Xeon Gold 6230 processor for supporting data analytics and deep learning applications, which can be used for the GSS application benchmark to compare the performance with CPUs, GPGPUs and other accelerators. Cray Urika CS is currently set up with optimized big data software stack and does not support the HPC software stack, so the evaluation of optimized HPC software stack support is needed before doing benchmark experiments. Different nodes and complete system details of HLRS Cray Urika CS is provided in Table 5.

Nodes	Cores	GPGPUs	Memory	Interconnect	Amount of Storage
8	2 sockets and 18 cores per socket	8 NVIDIA V100 GPUs per node	768 GiB DDR4	4x Mellanox CX-4	~500 TB Lustre storage, 8TB NVME local storage

Table 5: HLRS HPDA system with NVIDIA Voltas V100 GPGPU.

Dell, IBM, Intel and AMD are providing GPGPU servers for the HPC applications, so it has to be evaluated together with the HLRS HPDA system in the next deliverable to identify the best GPGPU systems for experiments. DELL provides an HPC server (PowerEdge c4140) with the combination of Intel Xeon 8280 CPUs and Tesla V100 GPUs to support optimized HPC system software stack. IBM provides an HPC system with NVLink2.0 between Power9 CPU and NVIDIA V100 GPU to improve the memory access between CPU and GPU system. Intel is planning to release its own X^e (Gen 12) GPGPU for supporting HPC applications by mid-2020 with 10 nm fabrication process and will be enhanced with the 7nm fabrication process by mid-2021, so it would be an alternative for NVIDIA GPGPU system. AMD provides an HPC system with the

combination of AMD Epyc 7742 CPUs and AMD Radeon Instinct MI50 GPGPUs to support both HPC and AI applications. AMD claimed that its GPGPU server performance is 26 times better than the server having Intel 8280 CPU and NVIDIA V100 GPUs [20], so it would be considered as an alternative for NVIDIA GPGPU systems in the market. In the next deliverable, we have to identify the GPGPU HPC servers based on the combinations of "Intel Xeon CPU and NVIDIA V100 GPGPU", "Intel Xeon CPU and Intel X^e GPGPU" and "AMD Epyc 7742 CPU and AMD Radeon Instinct MI50 GPGPU" to select the best GPGPU servers in the market for the GSS benchmark experiments.

2.2.3 Vector Co-Processor

NEC SX series offered an SX-Aurora Vector Engine (VE) to improve the performance of memory-bound HPC applications. SX-Aurora is a PCIe card-based accelerator, which contains 8 cores in a card to support 256 Words (16 KB) vector length operations with high bandwidth of 1.2TB/s. HPC application portability can be automated easily by enabling OpenMP targets (--fopenmp targets=aurora-nec-veort-unknown), which will automatically optimize the loop operations with vector operations to provide gain for OpenMP based GSS applications. Together with automatic parallelism, GSS application can be improved further manually by leveraging the SX-Aurora architectural feature during the application porting. Single SX-Aurora card's double-precision performance is 2.1 TFLOPS, and its performance is much lesser than the NVIDIA V100 GPGPUs' double-precision performance (7 TFLOPS). As well as SX-Aurora card is less expensive than the NVIDIA V100, so comparing the performance per price ratio will identify the best system from the accelerators in the market. HLRS provides 64 NEC SX-Aurora nodes with the following configuration in each node.

- 8 cores per processor/VE, 2.1 TFLOPS peak
- 1.4 GHz frequency
- 32 vector pipes per core, each doing 3 FMA per clock, resulting in 192 flops/cycle
- 64 vector registers to support 256 Words
- Little-endian data formats to communicate easily with x86 server CPUs
- Out of Order execution
- 256KB L2 cache per core
- 16MB shared LLC
- 48 GB HBM memory per node
- 6 HBM channels for 1.2TB/s memory bandwidth

2.3 Memory Technologies

Some of the GSS applications and benchmarks are memory- and I/O-bound, so the system has to support both high-bandwidth and low-latency memory and I/O operations to meet the

requirements of GSS applications. Processor speed is always higher than the memory and I/O speed, so the system has to be redesigned to bridge the gap between them to improve the performance of memory- and I/O-bound HPC applications.

2.3.1 DDR-SDRAM

DDR-SDRAM (Double Data Rate Synchronous Dynamic Random-Access Memory) is called in short DDR memory, which is used to store process details (program and data) during the execution of applications. Memory-bound applications highly depend on the speed of DDR memory, so providing high-bandwidth and low-latency memory is the need for those applications. DDR memory speed is measured in terms of bandwidth and it is increasing twice for every new DDR standards introduction. DDR4 is the memory system used in the latest HPC servers for supporting high bandwidth so that most of the servers are designed with DDR4 to support clock rate from 800MHz to 1600MHz and the memory bandwidth from 1600MB/s to 3200MB/s. DDR4 standard is available since 2014, but the highest speed and size of DDR memory (DDR4-3200) was introduced by Samsung in 2016 based on the 10nm fabrication process to support up to 32GB memory in a chip. HLRS Hawk and PSNC Eagle are powered by the latest DDR4-3200 memory.

DDR5 based memory is planned to be introduced from the mid of 2020, and will increase memory bandwidth twice, reduce power consumption by 0.1V to provide a better memory system for HPC applications. DDR5-4800 [21] is almost 1.8 times better than high-speed DDR4-3200 memory, so the server supporting DDR5-4800 and above has to be analysed in the next deliverable to finalise the best HPC nodes for GSS memory-bound applications based on the high-speed memory configurations.

2.3.2 NVRAM

NVRAM (Non-Volatile Random-Access Memory) is used as either main memory or buffer storage (I/O burst buffer), so it would be the best-fit in between the DRAM and SSD (Solid State Disk) to provide better bandwidth, locality and storage space for HPC applications. NVRAM provides both persistency and random access making it unique from DDR4 and SSD, and is classified as NVDIMM-N, NVDIMM-F and NVDIMM-P based on its nature. NVDIMM-P has both NVDIMM-N and NVDIMM-F functionalities to provide both random and block mode access. Intel 3D Xpoint is similar to NVDIMM-P, provides both memory and storage access and improved its performance with 3D integrated circuit technology to arrange the stacks of memory grids in a three-dimensional matrix. Intel Optane PMem and Intel Optane SSD are the two products based on the Intel 3D Xpoint technology to accelerate the performance of DDR4 and SSD drives.

Intel Optane PMem is using normal DDR4 memory slots to extend the size of main memory and persistency to support in-memory computing and high memory-bound applications. GSS HPC applications are required to be optimized for Intel Optane PMem to reduce input and output operations latency and improve overall turnaround time. Intel Cascade Lake is the only server processor currently supporting 3D Xpoint as a memory, so it has to be further evaluated in the next deliverable to identify the HPC servers providing support for Intel Optane PMem to improve the performance of memory-bound applications.

Intel Optane SSD is using the PCIe interface to provide caching between Intel Optane PMem and typical NAND SSD disk, which is providing better performance than the typical NAND based SSD disk to accelerate I/O operation of NAND SSD or replace the NAND SSD disks. Cray Datawarp and DDN IME (Infinity Memory Engine) used Intel Optane SSD as a burst buffer to accelerate the I/O performance of HPC applications, so it would be considered as a viable candidate for improving the performance of I/O-bound GSS applications. The HLRS Hawk system supports ~660 TB DDN IME burst buffer, so its performance can be evaluated with Cray Datawarp or NAND based SSD burst buffer to identify the best NVDIMM based burst buffer solution.

2.4 Exascale architectures and technologies

The constant growth in the performance of computer systems is becoming increasingly difficult. That is why newer and newer solutions are being sought not only in individual components but also in the framework of whole architectures. In this chapter, we concentrate on a few of them: quantum computer as a very unconventional approach and MPPAs, ARM and FPGA microservers as the evolution of the more conventional way.

2.4.1 Quantum Computing

Quantum computer uses quantum mechanics designed in the way that the result of the evolution of this system represents a solution of specific computational problem.

Evolution of the quantum system corresponds to the calculation process, data are represented by the current quantum state. Developing the quantum algorithm is seen as planning of the evolution of the quantum system. Thanks to the quantum computation results can be achieved much more effectively than using traditional computers. Any problem that a quantum computer can solve, can be solved by a classical computer (although in practice could take millions of years). However, achieved speedup could in practice significantly broaden the range of problems for which computers can be used.

Many institutions are currently attempting to build quantum computer including National Security Agency itself [22]. One of the most known implementations of this idea is one elaborated by D-Wave Systems company [23]. There is a heated discussion whether the solution proposed by D-Wave is a real quantum computer [24] [25]. The fact is that this system allows to carry out complex calculations or prove theorems in a much shorter time than previously possible [26] [27].

Europe is also focusing on quantum computing, e.g. at least two initiatives can be named: the Quantum Technologies Flagship [28], which foster the development of a competitive quantum industry in Europe and the upcoming EuroHPC JU calls [29], which are also focusing on the building of a European quantum simulator.

It must be noted that all algorithms performed on the quantum computer are probabilistic. This means that multiple executions of the same program on a quantum computer may produce completely different results due to the randomness of the quantum measurement process. Moreover, already defined HiDALGO models must undergo the process of remodelling to adapt them for quantum algorithms beforehand, which requires additional efforts.

Finally, it must be acclaimed that IBM introduced a quantum commercial IT solution (in the cloud) in January 2019. The IBM Q service has at its disposal quantum computers equipped

with a 20-qbit processor [30]. It makes this solution more approachable for wider (mostly scientific) community.

2.4.2 Massively Parallel Processor Arrays

Exascale computing needs a massive amount of computation capabilities, which can be accomplished by Massively Parallel Processor Arrays (MPPAs). The Sunway Taihulight supercomputer is the system recross-referencealised with MPPA based processor to achieve 105 PFLOPS and targets Exascale performance with the improvement in the SW26010 MPPAs architecture. SW26010 is designed with the motive of supporting compute-bound applications, so it is designed by following the System on Chip (SoC) architecture with 4 chips in a single processor. Each chip contains 64 Compute-Processing Elements (CPEs) for performing the actual computations and one Management Processing Element (MPE) for managing task scheduling, so totally 256 CPEs in a processor to achieve the theoretical peak performance of 2.9 TFLOPS.

SW26010 and Intel Knights Landing are compared in [31], because of its almost equivalent 3TFLOPS performance. SW26010 is having higher FLOPS per byte ratio of 33.84 than Intel Knights Landing 7.05, due to the bottleneck of accessing a shared memory or I/O by a large number of cores. SW26010 or MPPAs are currently designed with the motive of supporting compute-intensive applications, but it needs to be redesigned with the following suggestions to improve the performance of memory- and I/O-bound applications.

- Use UNIMEM [32] and UNILOGIC [33]to improve the memory and I/O-operations to meet the Exascale requirements.
- SW26010 currently has four memory channels with DDR3 memory. It has to be improved further with the support of a large number of memory channels, DDR4, DDR5 and HBM2 memory support in order to achieve high bandwidth memory operations.
- SW26010 has to support separate memory and I/O chips in the SoC design to improve the speed of memory, MPI and internetwork communications.

2.4.3 ARM-based Microservers with UNIMEM

Euroserver [34] is the European microserver designed to improve data-centre drastically in terms of energy efficiency, cost and performance to meet the needs of Exascale, so it is worth adapting GSS applications to plan for the future HPC system architecture. Euroserver is based on the technologies of 64-bit ARM cores, 3D heterogeneous silicon-on-silicon integration, and fully-depleted silicon-on-insulator (FD-SOI) process technology with new software techniques for efficient resource management to provide complete HPC hardware and software

ecosystem. Main improvements in the Euroserver are detailed below to assess the viability of Exascale computing support for HPC applications.

- 64-bit ARM cores and fully-depleted silicon-on-insulator (FD-SOI) are used, so it is the energy-efficient cores for supporting HPC workloads to improve performance-per-watt.
- 3D Silicon interconnect is used to achieve active imposer with SoC (System on Chip) to pack core, memory and interconnect in the single package, so the core-to-core, core-to-interconnect and core-to-memory performance will be improved dramatically.
- The system software stack is optimized to isolate the resource accessibility by multiple users by using virtualization technology.
- Remote nodes data can be accessed as same as local memory with cache coherence by using innovative UNIMEM technology [35].

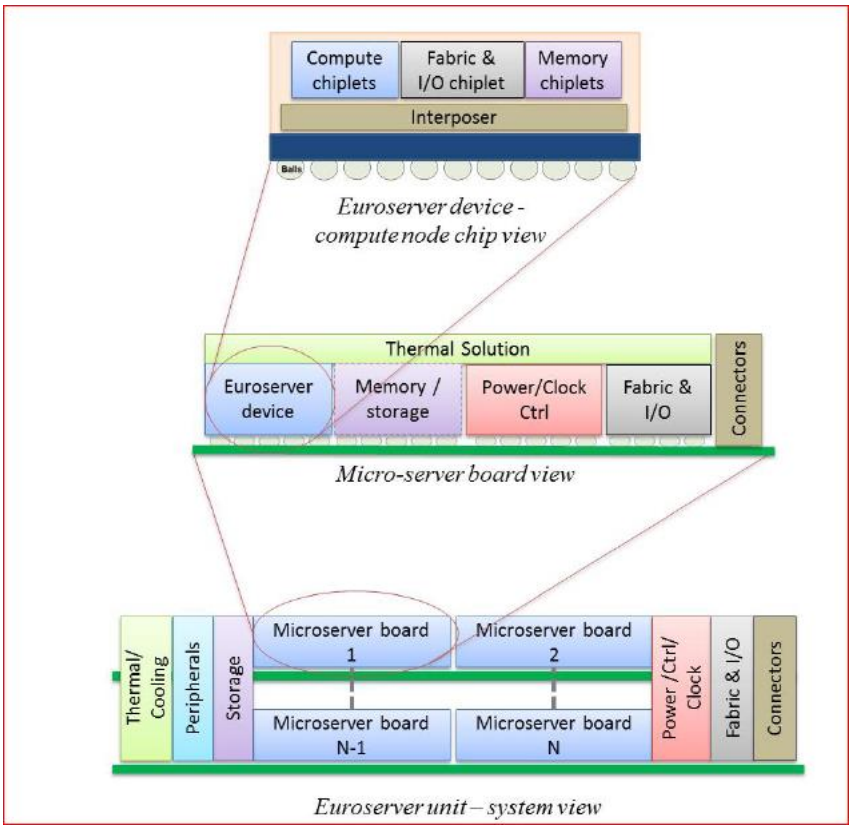

Figure 4: Euroserver HPC rack, board and processor in a high level.

Euroserver fabrication die is shown in Figure 4, which has three chiplets in a die, so the compute density and fabrication yield is increased to improve performance and energy efficiency at a reasonable cost. Silicon-on-silicon technology is used in the fabrication to customize the number of compute-, I/O- and memory-chiplets to produce the specialized

processors in the market based on the application needs, so there is a possibility to get the specialized chip for GSS HPC applications based on the initial results from the experiments. Microserver is having I/O-chiplets inside the die, so there is no need to use generic PCIe interconnect, which will improve the performance and energy efficiency of I/O-operations drastically to improve I/O-bound applications performance. UNIMEM is an important innovation in the Euroserver to support all the local, remote memory and I/O-operations are done through the DMC (Direct Memory Controller) and DMA (Direct Memory Access) with local cache coherent and multi-level global interconnect, as depicted in Figure 5. This multi-chiplet architecture with UNIMEM allows cross allocation of memory and I/O resources between multiple nodes to enable global physical address space to support PGAS and COMPSs programming model effectively. The system software is optimized to support resource isolation (memory capacity and I/O) for running multiple processes in the system, so UNIMEM memory can be allocated and accessed by single HPC applications to improve security and proper resource allocation. The HPC server is designed to provide up to 64 micro-server boards, at very high density, together with networking, I/O, storage, and power supply, into a unit compatible with standard 42U server racks.

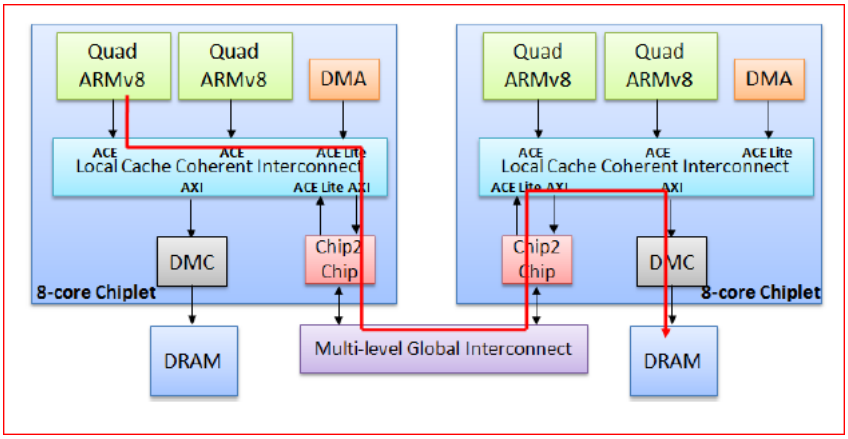

Figure 5: Data accessibility between nodes through UNIMEM technology.

2.4.4 FPGA based Microservers with UNILOGIC

FPGA is envisaged as a viable accelerator in the HPC field to improve the performance of compute-intensive applications by using custom-hardware performance with low power consumption and easy reprogrammable capability. Ecoscale FPGA prototype is designed with the motive of supporting Exascale capability with innovative UNILOGIC technology and its hardware-software ecosystem to support automatic parallelization. UNIMEM provides efficient data movement within the system, which is enhanced in the UNILOGIC to support FPGA accelerators as shown in Figure 6. UNILOGIC based microservers are not designed only

for easy data movement between CPUs and FPGAs, the computation can be offloaded to the FPGA locally and remotely to unify the capabilities of the distributed system. FPGAs located at the remote is accessed by physical (PHY) to Global and Global to PHY address translation process, which is further explained in the paper with more details regarding the data movement between processors, memory, FPGAs locally and remotely in a cache coherent manner.

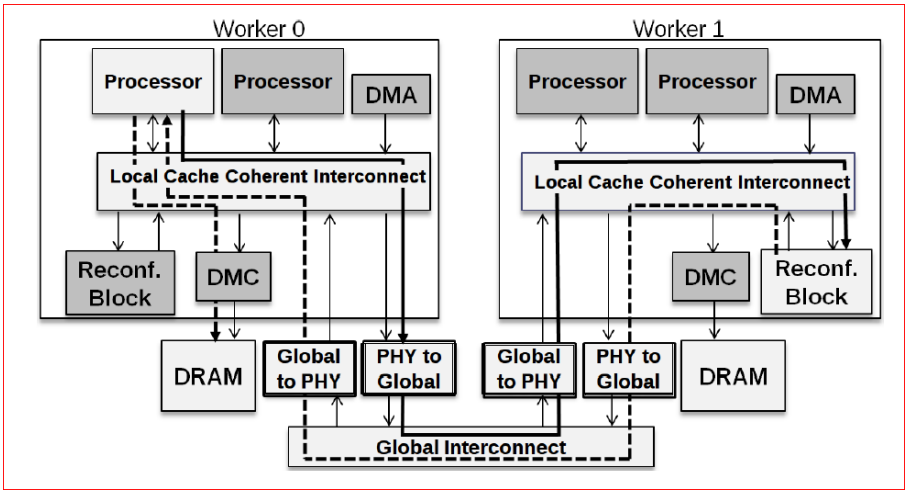

Figure 6: FPGA with UNILOGIC to accelerate data movement between multiple nodes and offload the operations to local and remote FPGAs. Reconfigurable Block in the figure means FPGA.

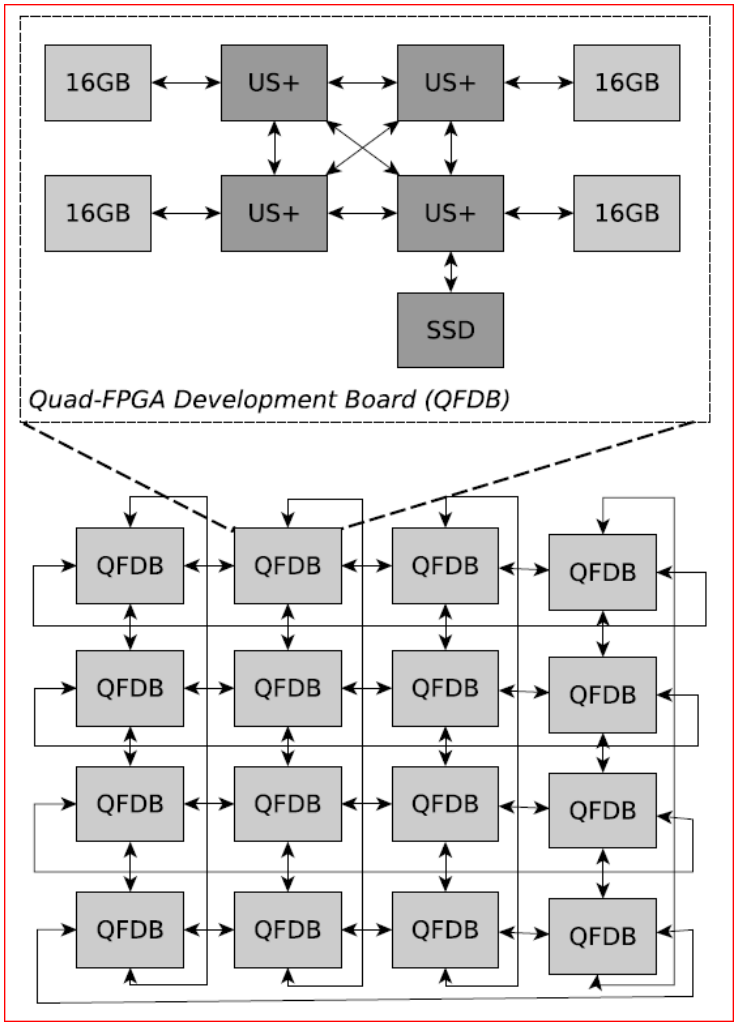

Figure 7: ECOSCALE FPGA prototype node with 16 boards and 4 FPGAs per board to provide a large number of FPGAs for HPC computation.

Computationally intensive HPC applications can be spread out onto the Ecoscale hardware resources and executed in parallel by using many FPGAs to accelerate the performance. The data required for the application's processing can also be spread out in order to bring them closer to the local FPGAs to improve locality between local and remote FPGAs. Ecoscale has designed the QFDB (Quad-FPGA Development Board) prototype node to support a large number of FPGAs as shown in Figure 7, which is used by compute nodes to provide central access like I/O nodes in a typical HPC system. Ecoscale prototype system used UltraScale+ (US+) FPGA from the vendor Xilinx, which is designed by using 16nm fabrication process for HPC applications, so it would be a viable candidate for GSS applications and meet the needs of Exascale HPC system. Ecoscale supports automatic parallelization with OpenCL, so the OpenCL based GSS applications can be easily ported and tested in the prototype with the Ecoscale software suite (compiler, library and virtualization technology).

3 Tools and libraries

In this chapter, we present an overview of available tools and libraries used in HPC with particular focus on supporting high scalability which paving the way to Exascale systems. Following are the most mature examples of performance tools, mathematical and programming libraries and workload managers.

3.1 Performance tools

In this section, we present a variety of software designed to evaluate speed and efficiency of HPC systems. For each tool, suitability for large-scale applications is considered.

3.1.1 Exa-PAPI

PAPI (Performance Application Programming Interface) is a low-level interface for performance supervising tools ([36], section 2.2.2). It allows monitoring of interactions between software and hardware on HPC system. It is used as an intermediate layer in numerous projects, e.g. TAU, Scalasca, Vampir, HPCToolkit, Score-P and many more, and its main functionality is linking application-defined events to various hardware counters (like CPU, GPU, I/O or energy usage) with minimal overhead (30-40ns per event [37]).

Exa-PAPI [38] is an extension of regular PAPI, which helps managing systems at an exceedingly large scale. It includes features like providing support for Exascale hardware and software, power management [39] and variable event granularity. What is more, the redesigned software-defined events have reduced overhead during measurements to levels close to hardware counter performance monitoring [37].

3.1.2 HPCToolkit

HPCToolkit [40] is a suite of tools for measurement and visualization of performance of distributed programs. The calculations use sampling in order to minimize overhead and maximize scalability. It supports applications with sequential, threaded, distributed and hybrid code.

Within the Exascale Project, progress has been made concerning GPU support, recovering control flow graphs from machine code and new interface implementation, which now uses native Linux performance monitoring substrate *perf events* [41].

Other improvements include: large scale, lightweight hardware counters, OS activity monitoring, measurement data storage, parallelisation of performance data analysis, and

integrating varying approaches to visualization (focused on code, time, data or resources) [42].

3.1.3 TAU

TAU [43] (Tuning and Analysis Utilities) is a framework for analysing parallel performance on multiple platforms. It uses probes added inside the program code (instrumentation) or interrupts (sampling) for measurement. The first approach is time and work consuming but grants significant control over data granularity. The second one does not require modification of executables but is considerably less detailed and flexible. Currently TAU supports FORTRAN, C++, C, UPC, Java, Python, Chapel and Spark [44].

With this tool, users can trace and identify sources of performance bottlenecks in their parallel application. It is being actively maintained and recently, runtime monitoring and tuning mechanisms have been added in order to prepare for Exascale operation within the Extreme-scale Scientific Software Stack (E4S [45]) [46].

3.1.4 Score-P

Score-P [47] is a cross-tool measurement infrastructure. It implements a common interface (Open Trace Format 2 standard) that allows numerous tools to perform their work while using single measurement system. It supports multiple ways of instrumentation as well as sampling [48].

Using it requires rebuilding an application, but on the other hand, measurements can be performed while the application is still running.

The mechanism for global system representation creation and storing has been reworked in order to improve scalability. Due to parallel creation and storing of this meta-data, it is now possible to conduct performance reporting from large-scale systems (458752 cores) [49].

3.1.5 VampirServer

Vampir [50] is a performance visualization tool, working in tandem with Score-P or TAU. It works with trace data, which is collected after measurements are done (post-mortem). At first a thumbnail of the entire dataset is presented, which allows the user to select a subsection that they are interested in for closer analysis using detailed timelines. This gives more control over granularity of the displayed data.

VampirServer is a parallel implementation of Vampir, using client-server architecture. In VampirServer all, the costly data preparation calculations are performed in parallel with using MPI, pthreads and sockets. The bulky data is kept close to the server and only after it has been processed, the visualization results are sent to the client machine [51].

As writing full tracefile (post-mortem) at large scale is not feasible, in order not to overwhelm I/O system, monitoring can be run in online mode, that is to say analysis is conducted entirely in main memory [52]. In tandem with Score-P as measurement system, VampirServer has been run on 224256 cores [53].

3.1.6 Darshan

Darshan [54] is a tool specialized in profiling I/O operations behaviour. It introduces minimal overhead as it is implemented as a set of user-space libraries added to the program during linking phase.

Both system-wide and application-specific behaviour can be monitored in order to better correlate I/O activity and produce an overview of I/O performance [55]. This way it is possible to investigate statistics and cumulative timing of application I/O by showing e.g. memory access patterns, sizes and number of operations.

Because tools using statistical sampling may prove inaccurate, Darshan monitors each file operation and only when reaching limits of scale, it resorts to coarser readings. Thanks to this mechanism, it has been able to analyse performance on a 163840-core system [56].

3.1.7 Relevance to HiDALGO

Performance tools find a wide use in Hidalgo project. Score-P, PAPI and Vampir already confirmed to be adopted by all the pilots. Additionally, in case I/O performance proves to be a bottleneck, Darshan can be adopted in order to analyse the issue. As the project increases in scale, Exascale extensions of those programs (Exa-PAPI power management or VampirServer distributed visualization) are sure to be integrated and extensively utilized.

3.2 Mathematical libraries

As most HPC applications are using complex numerical methods, it is necessary to use appropriate tools that facilitate calculations at ever-increasing scale.

3.2.1 NumPy

NumPy [57] is a popular and well-supported scientific computing Python package. It enables broad functionality like powerful n-dimensional arrays, sophisticated broadcasting functions, basic linear algebra functions and Fourier transforms, sophisticated RNG, as well as tools for integrating Fortran and C/C++ code for increased performance.

NumPy arrays offer significant speed up over standard Python ones. It is achieved by automatically vectorising function calls, which means operation on all members of an array is performed simultaneously by distributing the workload [58].

Currently several extensions (CuPy, Bohrium, JAX, Weld) make use of GPU, lazy evaluation and JIT-compilation, which further increase speed and scaling [59]. NumPy is used on ARCHER2 supercomputer with 748544 CPU cores [60].

3.2.2 SuperLU

SuperLU (Supernodal LU) [61] is a library for solving linear equations and Fourier transforms in parallel (OpenMP, CUDA) or distributed manner (MPI) [62].

SuperLU takes advantage of the sparsity structure of the matrix – it can automatically determine which matrix entries are zeros and thus can be ignored. It also uses a static pivoting strategy. The matrix is permuted beforehand, so that the largest elements of the input matrix are placed on the diagonal, instead of swapping at runtime and performing partial pivoting [63].

Because the current algorithm (MC64) for this operation is serial, there are attempts to develop parallelized version (AWPM – approximate-weight perfect matching), which provides up to 2500x speedup on large scale machines [64].

3.2.3 PetSc

PetSc/TAO [61] (The Portable Extensible Toolkit for Scientific Computations/Toolkit for Advanced Optimization) is a scalable mathematical library for solving partial differential equations [62]. It works in tandem with the TAO optimization library, which handles efficient and scalable function handling [65].

PetSc supports MPI, and GPUs through CUDA or OpenCL, as well as hybrid MPI-GPU parallelism for massively parallel applications running on hybrid hardware [66]. It is used on 200PFlop/s OLCF Summit supercomputer [66].

3.2.4 SLATE

SLATE [67] (Software for Linear Algebra Targeting Exascale) is designed to solve dense linear algebra systems using distributed-memory environments (including GPU-accelerated ones). It leverages distributed programming models and runtime scheduling systems [68].

It is intended to replace existing LAPACK and ScaLAPACK libraries by supporting modern, heterogeneous HPC systems with multiple hardware accelerators per node and implementing parallel matrix storage [69]. Changes to multi-threaded performance result in 30% improvement over legacy ScaLAPACK using analogous algorithms [70].

It is used on the ORNL Summit supercomputer [71].

3.2.5 Relevance to HiDALGO

Because of large amount of computation, mathematical libraries find numerous applications in Hidalgo project. Social Network pilot uses both NumPy and PetSc and Migration pilot utilizes tools like FabSim, Muscle, and Flee, which utilize NumPy. In addition, should a need for linear algebra tool arise, SLATE can be adopted in place of widespread ScaLAPACK for better scalability.

3.3 Open standards and programming libraries

This section presents commonly used programming models of both node-level and distributed parallelism.

3.3.1 MPI

MPI [72] (Message Passing Interface) is a standard communication API commonly used for inter-node synchronization by large-scale systems. With low consumption of resources per process, multithreaded communication and high fault-tolerance, it was designed with scalability in mind [73]. It has been designed to take advantage of high-speed communication interfaces like NVIDIA GPUDirect and Mellanox InifiniBand networks [74].

MPI-3.1 is the latest standard of this technology, which introduces support for hybrid programming, improves remote-memory access, and brings in better support for parallel debuggers and profiling software [75]. The changes in the standard are supported by all major implementations [76] and already new goals have been proposed for the next iteration (MPI 4.0) [77].

Further, as failures increase as the number of nodes grows, new fault resilience mechanisms have been proposed [78].

3.3.2 OpenMP

OpenMP [79] (Open Multi-Processing) is a directive-based standard for developing parallel, shared-memory applications. It is commonly used in tandem with MPI for hybrid computation both within and between multi-CPU machines.

It also comes with OMPT – an interface for performance monitoring and debugging tools, which provide support for asynchronous sampling and instrumentation monitoring for runtime events at negligible cost [72].

OpenMP API 5.0 brings full accelerator support (GPU, FPGA), which allows for offloading a block of code to the coprocessor. It maps OpenMP abstractions onto lower-level mechanisms managed by the accelerator, which provides an ease of use for the cost of suboptimal default control values [80].

Several shortcomings for reaching Exascale level performance, like portable data layout abstractions, performing deep copies to/from GPU, portability and updating to latest C++ standard have been addressed by the SOLLVE project [81].

3.3.3 CUDA/OpenCL

OpenCL [82] (Open Computing Language) and CUDA [83] (Compute Unified Device Architecture) are general-purpose, parallel programming frameworks designed for GPU computing. They share programming model, however runtime API for CUDA is higher-level than OpenCL and despite being platform dependent it is more common [73].

When it comes to tool integration, CUDA comes with CUDA Profiling Tools Interface (CUPTI), providing callback API, which is used for example by Score-P. OpenCL comes with no similar interface [84].

OpenCL 3.0 is the newest iteration, which allows deploying applications onto platforms without native drivers for increased flexibility [85].

3.3.4 Relevance to HiDALGO

In order to scale the application properly, both MPI and OpenMP are used extensively in the project. They enable both parallel and distributed processing of data at exceedingly large scales. Additionally, when infrastructure with dedicated accelerators becomes available it is worth considering adopting CUDA/OpenCL for increased parallelization. However, the switch would require a major change in the architecture of the programs.

3.4 Workload managers

Workload managers are common tools on HPC environments, which help to manage resources by orchestrating job execution in reasonable manner. They grant users' jobs access to resources, allow work management, accounting, and maintain a queue of awaiting jobs in order to prevent conflicts for resources. They also enable multiple users to use these resources easily by providing a transparent interface.

3.4.1 Slurm

Slurm [86] is a modular, extensible and scalable job scheduling system and cluster resource manager. It uses multifactor fair-share queue with sophisticated scheduling algorithms like elastic scheduling, gang scheduling and pre-emption. It provides accelerator support and uses backup daemons for resilience.

Additionally, Slurm provides third party plugins API, which allow new resilience and scheduling algorithms to be added in [87]. It is featured on largest supercomputers (Sunway TaihuLight 10649600 CPU cores) with up to 1000 job submissions per second [88].

3.4.2 Torque/Moab

Torque [89] (Terascale Open-Source Resource and Queue Manager) is another distributed resource management suite. It is a fork of OpenPBS with additional features like GPU scheduling, extensive diagnostics and monitoring, and programmable queue. It is integrated with Moab meta-scheduler (a closed source commercial product) [90].

It features extensions made by multiple leading edge HPC organisations and is used on Kraken supercomputer (112896 computing cores) [73].

3.4.3 Altair PBS Professional

PBS Pro [91] is a version of PBS maintained by Altair. It is offered in two variants - paid support and open-source. It comes with fully configurable queue, topology-aware scheduler, GPU scheduling, performance data analysis and EAL3+ security certification [92].

It has been tested in environments with over 70000 nodes before performance started to deteriorate [93]. Its source code has been opened to the HPC community in May 2016 [90].

3.4.4 Relevance to HiDALGO

As Hidalgo project applications all run in HPC environments it is necessary to pick a cluster management system. As all alternatives support large scale, they can be utilized to schedule tasks of any pilot. These solutions are already adopted by project participants - HLRS has Torque and PSNC uses Slurm.

4 AMD Rome benchmark

In this chapter, we present preliminary findings of testing the scalability of HiDALGO Pilot Applications on AMD EPYC Rome CPUs. The newly acquired Hawk supercomputer at HLRS is built from AMD EPYC Rome 7742 CPUs. Additionally, an AMD EPYC Rome 7702 CPU was briefly available at PSNC. In this phase, we evaluate the single-node performance of Hawk, which embraces the new AMD EPYC Rome architecture, in comparison to our prior findings, on the Intel Xeon nodes of Eagle at PSNC.

Table 6 summarizes the comparison between the node architecture of the two systems. It is evident that AMD EPYC Rome offers ample parallelism, with 128 cores and 256 hardware threads in a two-socket NUMA setup. However, the per core memory bandwidth is lower on the AMD EPYC than on the Intel Xeon node, which can be a limiting factor for the scalability of applications with low operational intensity.

	Hawk (HLRS)	Eagle (PSNC)
CPU model	AMD EPYC Rome 7742	Intel Xeon E5-2697 v3
CPUs/node	2	2
Cores/CPU	64	14
RAM/node	256GB	64GB

Table 6. Node comparison - Hawk and Eagle

4.1 Migration Pilot

We have evaluated and compared the performance of Flee on a single node of Hawk against a single node of Eagle. For details on the Flee version in use, we refer the reader to Deliverable 3.3, Section 2.1.1. We have performed a micro-scale simulation with Flee for 10 days (t=10). We have used synthetically generated graphs, and have simulated two cases, one where the initial number of agents is 100K and 1000 new agents are added per time step, and one where the initial number of agents is 2M and 10K new agents are added per time step. We use the "advanced" parallelization mode and the "high-latency" communication mode of the Flee code.

Figure 8 and Figure 9 show the execution time for Flee on two synthetic graphs, the 10-10-4 graph (Figure 8) and the 50-50-4 graph (Figure 9), for two different simulation settings with respect to the number of initial agents and added agents. We first note that on the AMD Rome

processor, the execution time of Flee is smaller than the equivalent on the Intel Xeon processor for all core counts up to 32 cores. In addition, for these core counts, we observe the same scalability behaviour for Flee on both types of architectures. In Figure 8, where the location graph is smaller, Flee demonstrates better scalability when the available work per core increases, i.e. when the number of agents in the simulation is higher. In this case, we also observe that Flee continues to scale on the AMD Rome node when adding more cores. Contrarily, in Figure 9, where the location graph is larger, Flee scalability significantly decreases on both architectures. This is due to Flee becoming communication-bound, as the frequency of data exchanges depends on the location graph, and in particular, the number of locations. We note that on AMD Rome, for the larger location graph, Flee performance deteriorates as we add more cores. This effect is more evident when the number of agents in the simulation is smaller, as the available work per core is equivalently smaller, thus the communication/computation ratio is higher.

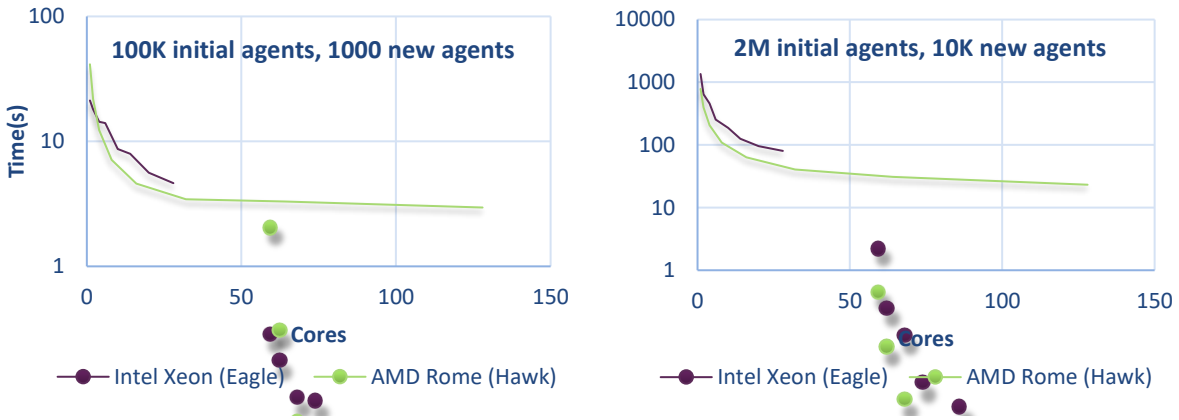

Figure 8: Evaluating execution time of Flee on a synthetic 10-10-4 graph using 100K initial agents and 1000 new agents per time step (left) and 2M initial agents and 10K new agents per time step (right) (logarithmic y-axis)

Figure 9: Evaluating execution time of Flee on a synthetic 50-50-4 graph using 100K initial agents and 1000 new agents per time step (left) and 2M initial agents and 10K new agents per time step (right) (logarithmic y-axis)

Comparing the two architectures, for the case of Flee, large-scale simulations can benefit from the high number of cores per node, since this version of Flee is not memory-bound. On the other hand, in this version of Flee, frequent communication limits the application scalability even on a single node. We note, however, that Flee has been re-engineered and we expect the impact of communication to be less severe on the latest release. In addition, the architecture of AMD EPYC Rome is highly hierarchical. Fine-tuning of MPI as well as a more sophisticated placement of processes on the node can potentially improve the locality of the application and improve its performance.

4.2 Urban Air Pollution Pilot

We have evaluated and compared the performance of the OpenFOAM version of the Urban Air Pollution Pilot workflow on two nodes of the Eagle supercomputer against a single AMD EPYC Rome 7702 node, provided by PSNC. For details on the simulation setup, we refer the reader to Deliverable 3.3, Section 2.2. The simulation benchmark is done with steady and transient state calculations of wind field and pollution dispersion using synthetic data for boundary conditions of traffic and wind speed. Scaling is measured on a mesh for the city of Gyor, with a mesh of 921K cells. The mesh is generated using OpenFOAM's own mesh generator, snappyHexMesh beforehand. Figure 10 shows the execution time of the simulation on the two architectures. Note that for Eagle, the measurement on 56 cores correspond to two nodes. Overall, the AMD Rome node provides better execution time for up to 32 cores, compared to the Intel Xeon node. However, the execution time increases when more than 32 cores are used. Therefore, the simulation in its status is not able to take advantage of the available parallelism on a single AMD Rome node. However, as with the Migration pilot, fine-tuning of MPI as well as more sophisticated process placements could potentially lead to better performance results.

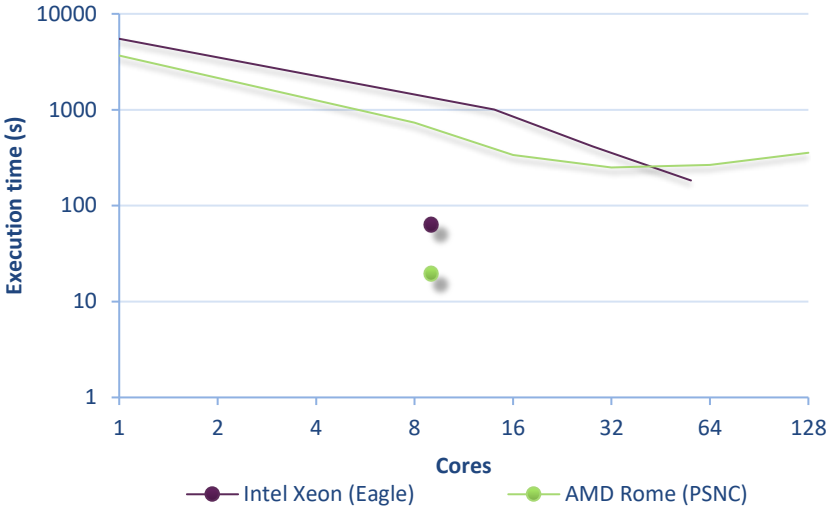

Figure 10: Execution time of the OpenFOAM Air Quality Dispersion Model with OpenFOAM, with a generated input mesh of 921K cells. Both axes are logarithmic.

5 Conclusion

Due to the wide spectrum of computation and analysis performed by HiDALGO applications in this report, we had to cover many aspects of data processing. It started with the most important system components, which is the CPU. Paying attention to the latest achievements of major manufacturers, we deduce that GSS applications may significantly benefit from Intel Cascade Lake and AMD Rome processors. Accelerators are also a very promising solution to improve application performance, especially GPGPU and FPGA dies. Unfortunately, though they also require significant amount of effort to cast the existing code to programming paradigm required by certain technologies, especially in respect delegating specific code areas for special processing and inter-process communication.

Quantum computing is another promising path, however, due to its development status, accessibility and meaningful entry work (algorithms adaptation), it cannot be considered as option for the next few years. Other architectures such as MPPA and ARM seem to be more accessible and realistic to use for GSS computation, especially the first one.

In chapter 3, we discussed a set of tools, which could help to achieve better performance gains. Presented high scalability scores of them give us good perspective potential boundaries in profiling and parallelization according to certain aspects.

From benchmarking, we learned that the Flee application could benefit in large-scale simulations from the high number of cores per node, since this version of Flee is not memory-bound. For the OpenFoam application, AMD Rome node provides better execution time for up to 32 cores, compared to the Intel Xeon node. However, the simulation in its status is not able to take full advantage of the available parallelism on a single AMD Rome node (the execution time increases when more than 32 cores are used). Nonetheless, in both cases fine-tuning of MPI as well as more sophisticated process placements could potentially lead to better performance results.

Based on the information provided in Annex 1 we may select projects working on the same ground in order to create collaborations for our Exascale endeavour.

In the next steps, we are going to establish liaison with manufacturers and vendors, which could supply us with edge technologies. They will be tested on pilot applications in order to assess the understanding of the benefits of using them. More findings will be presented in the consecutive deliverable.

References

[1] “Cascade Lake microarchitecture details from wikichip,” [Online]. Available: https://en.wikichip.org/wiki/intel/cores/cascade_lake_sp.

[2] “CoeGSS project,” [Online]. Available: http://coegss.eu/.

[3] “CoeGSS D5.8 deliverable,” [Online]. Available: http://coegss.eu/wp-content/uploads/2018/11/D5.8.pdf.

[4] “AMD Optimizes EPYC Memory with NUMA, Tirias Research,” [Online]. Available: https://www.amd.com/system/files/documents/TIRIAS-White-Paper-AMD-Infinity-Architecture.pdf.

[5] “2nd Gen AMD EPYC “Rome” CPU Review: A Groundbreaking Leap for HPC,” [Online]. Available: https://www.microway.com/hpc-tech-tips/amd-epyc-rome-cpu-review.

[6] “OpenCAPI Consortium,” [Online]. Available: https://opencapi.org/.

[7] “NVLink,” [Online]. Available: https://www.nvidia.com/en-us/data-center/nvlink/.

[8] “IBM Power ISA,” [Online]. Available: https://openpowerfoundation.org/?resource_lib=power-isa-version-3-0.

[9] “Single instruction, multiple data (SIMD),” [Online]. Available: https://en.wikipedia.org/wiki/SIMD.

[10] “Vector-Scalar Floating-Point Operations,” [Online]. Available: http://openpowerfoundation.org/wp-content/uploads/resources/Vector-Intrinsics-4/content/sec_power_vector_scalar_floatingpoint.html.

[11] “SPARC in Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/SPARC.

[12] “Galaxy FT-1500 processor,” [Online]. Available: https://en.wikipedia.org/wiki/FeiTeng_(processor).

[13] “Fujitsu SPARC Servers Roadmap,” [Online]. Available: https://www.fujitsu.com/global/products/computing/servers/unix/sparc/key-reports/roadmap/.

[14] "ASIC: application specific integrated ciruit," [Online]. Available: https://www.electronics-notes.com/articles/electronic_components/programmable-logic/what-is-an-asic-application-specific-integrated-circuit.php.

[15] "FPGAs and the Road to Reprogrammable HPC," [Online]. Available: https://insidehpc.com/2019/07/fpgas-and-the-road-to-reprogrammable-hpc/.

[16] "Accelerating High-Performance Computing With FPGAs," [Online]. Available: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/wp/wp-01029.pdf.

[17] "Artificial Intelligence Accelerates Dark Matter Search," [Online]. Available: https://www.xilinx.com/publications/powered-by-xilinx/cerncasestudy-final.pdf.

[18] "NVIDIA TESLA V100 GPU ARCHITECTURE," [Online]. Available: https://images.nvidia.com/content/volta-architecture/pdf/volta-architecture-whitepaper.pdf.

[19] "TESLA V100 PERFORMANCE GUIDE Deep Learning and HPC Applications," [Online]. Available: https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/tesla-product-literature/v100-application-performance-guide.pdf.

[20] "AMD corporate presentation," [Online]. Available: https://ir.amd.com/static-files/fd06c15e-0241-424d-9fd9-5a469d96012d.

[21] "Introducing Micron® DDR5 SDRAM: More Than a Generational Update, Scott Schlachter and Brian Drake," [Online]. Available: http://www.micron.com/-/media/client/global/documents/products/white-paper/ddr5_more_than_a_generational_update_wp.pdf.

[22] "NSA Trying To Build A Quantum Computer To Crack Encryption," [Online]. Available: https://techcrunch.com/2014/01/02/report-nsa-trying-to-build-a-quantum-computer-to-crack-encryption/.

[23] "D-Wave Systems," [Online]. Available: https://www.dwavesys.com/.

[24] "Assessing claims of quantum annealing: Does D-Wave have a quantum computer?," [Online]. Available: http://meetings.aps.org/Meeting/MAR14/Event/211739.

[25] "D-Wave chip passes rigorous tests," [Online]. Available: https://phys.org/news/2014-03-d-wave-chip-rigorous.html.

[26] "Experimental determination of Ramsey numbers," [Online]. Available: https://arxiv.org/abs/1201.1842.

[27] "Quantum computer built inside a diamond," [Online]. Available: https://www.sciencedaily.com/releases/2012/04/120404161943.htm.

[28] "Quantum Technologies Flagship," [Online]. Available: https://ec.europa.eu/digital-single-market/en/quantum-technologies.

[29] "The EuroHPC Joint Undertaking," [Online]. Available: https://eurohpc-ju.europa.eu/participate.html.

[30] "IBM unveils its first commercial quantum computer," [Online]. Available: https://www.newscientist.com/article/2189909-ibm-unveils-its-first-commercial-quantum-computer/.

[31] Z. Xu, "Benchmarking SW26010 Many-core Processor," *IEEE International Parallel and Distributed Processing Symposium Workshops,* 2017.

[32] Y. H. D. L. Kai Wu, "Unimem: runtime data managementon non-volatile memory-based heterogeneous main memory," [Online]. Available: https://www.researchgate.net/publication/320938366_Unimem_runtime_data_management_non-volatile_memory-based_heterogeneous_main_memory.

[33] UNILOGIC, "D3.1 Specifications of HW Architecture and Prototype from ECOSCALE project".

[34] Y. Durand, "EUROSERVER: Energy Efficient Node for European Micro-servers," *17th Euromicro Conference on Digital System Design,* 2014.

[35] M. Marazakis, "EUROSERVER: Share-Anything Scale-Out Micro-Server Design," *Proceedings of the 2016 Design, Automation & Test in Europe Conference & Exhibition,* 2016.

[36] "HiDALGO D3.1 Report on Benchmarking and Optimisation," [Online]. Available: https://hidalgo-project.eu/sites/default/files/2019-04/HiDALGO_D3.1%20Report%20on%20Benchmarking%20and%20Optimisation_v1.0.pdf.

[37] H. Jagode, A. Danalis, H. Anzt and J. Dongarra, "PAPI software-defined events for in-depth performance analysis," *International Journal of High Performance Computing Applications 33(6),* 2019.

[38] "Exa-PAPI home page," [Online]. Available: http://icl.cs.utk.edu/exa-papi/. [Accessed 27 April 2020].

[39] A. Haidar, H. Jagode, P. Vaccaro, A. Yarkhan, S. Tomov and J. Dongarra, "Investigating power capping toward energy-efficient scientific applications," *Concurrency and Computation Practice and Experience,* 2018.

[40] "HPCToolike home page," [Online]. Available: http://hpctoolkit.org/. [Accessed 27 April 2020].

[41] J. Mellor-Crummey, "HPCToolkit at Exascale Project," [Online]. Available: https://www.exascaleproject.org/wp-content/uploads/2020/02/ECP_ST_HPCToolkit.pdf. [Accessed 27 04 2020].

[42] J. Mellor-Crummey, "Performance Analysis of MPI+OpenMP Programs with HPCToolkit," March 2015. [Online]. Available: http://hpctoolkit.org/slides/hpctoolkit-og15.pdf.

[43] "TAU home page," [Online]. Available: https://www.cs.uoregon.edu/research/tau/home.php. [Accessed 27 April 2020].

[44] S. Shende, "TAU performance system," 8 November 2017. [Online]. Available: https://www.exascaleproject.org/wp-content/uploads/2017/05/Tau-Performance-System.pdf. [Accessed 27 April 2020].

[45] "Extreme-scale Scientific Software Stack home page," [Online]. Available: https://e4s.io. [Accessed 27 April 2020].

[46] S. Shende, "Tuning and Analysis Utilities (TAU) Seminar," 14 October 2019. [Online]. Available: https://www.cs.uoregon.edu/research/tau/CERN19.pdf. [Accessed 27 April 2020].

[47] "Score-P home page," [Online]. Available: https://www.vi-hps.org/projects/score-p/. [Accessed 27 April 2020].

[48] A. D. Malony and F. G. Wolf, "Performance Refactoring of Instrumentation, Measurement, and Analysis Technologies for Petascale Computing. The PRIMA Project," 2014.

[49] D. Lorenz and C. Feld, "Scaling Score-P to the next level," *Procedia Computer Science,* vol. 108, pp. 2180-2189, 2017.

[50] "Vampir home page," [Online]. Available: https://vampir.eu/. [Accessed 27 April 2020].

[51] M. S. Müller, A. Knüpfer, M. Jurenz, M. Lieber, H. Brunst, H. Mix and W. E. Nagel, "Developing Scalable Applications with Vampir,VampirServer and VampirTrace,"

Parallel Computing: Architectures, Algorithms and Applications, vol. 38, pp. 637-644, 2007.

[52] M. Weber, J. Ziegenbalg and B. Wesarg, "Online Performance Analysis with the Vampir Tool Set," *Tools for High Performance Computing ,* 2017.

[53] W. E. Nagel, "From Tera- to Peta- to ExaScale: Performance-Optimization with VAMPIR," [Online]. Available: http://www-hpc.cea.fr/en/Wotofe/docs/18-121002-Nagel-CEA-Toools12.pdf. [Accessed 27 April 2020].

[54] "Darshan home page," [Online]. Available: https://www.mcs.anl.gov/research/projects/darshan/. [Accessed 27 April 2020].

[55] T. P. Straatsma, K. B. Antypas and T. J. Williams, "Exascale Scientific Applications: Scalability and Performance Portability," in *Exascale Scientific Applications: Scalability and Performance Portability*, Chapman and Hall/CRC, 2018, p. 36.

[56] P. Carns, K. Harms, W. Allcock, C. Bacon, S. Lang, R. Latham and R. Ross, "Understanding and Improving Computational Science Storage Access through Continuous Characterization," 2011.

[57] "NumPY home page," [Online]. Available: https://numpy.org/. [Accessed 27 April 2020].

[58] R. H. Landau, M. J. Páez and C. C. Bordeianu, "Computational Physics: Problem Solving with Python Third edition," in *Computational Physics: Problem Solving with Python Third edition*, 2015, p. 252.

[59] M. Bauer and M. Garland, "Legate NumPy: accelerated and distributed array computing," 2019.

[60] T. Rollin, W. Scullin and M. Belhorn, "Python in HPC," 7 June 2017. [Online]. Available: https://www.exascaleproject.org/wp-content/uploads/2017/05/IDEAS-Python-in-HPC-Thomas-Scullin-Belhorn.pdf. [Accessed 27 April 2020].

[61] "SuperLU home page," [Online]. Available: https://portal.nersc.gov/project/sparse/superlu/. [Accessed 27 April 2020].

[62] H. Anzt, E. Boman, R. Falgout, Ghysels Pieter, M. Herous, X. Li, L. C. McInnes, R. T. Mills, S. Rajamanickam, K. Rupp, B. Smith, I. Yamazaki and U. M. Yang, "Preparing sparse solvers forexascale computing," *Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences,* no. 378, 2020.

[63] M. Bernhardt, "Exascale Computing Project Spotlights ExaGraph and STRUMPACK/SuperLU Collaboration," Lawrence Berkeley National Laboratory, 29

August 2018. [Online]. Available: https://crd.lbl.gov/news-and-publications/news/2018/exascale-computing-project-spotlights-exagraph-and-strumpacksuperlu-collaboration/. [Accessed 27 April 2020].

[64] A. Azad, A. Buluc, X. S. Li, X. Wang and J. Langguth, "A distributed-memory approximation algorithm for maximum weight perfect bipartite matching," 2018.

[65] S. Benson, L. C. McInnes, J. J. Moré and J. Sarich, "TAO Users Manual," 20 November 2003. [Online]. Available: https://digital.library.unt.edu/ark:/67531/metadc780051/m2/1/high_res_d/822565.pdf. [Accessed 27 April 2020].

[66] R. T. Mills, "Progress with PETSc on Manycore and GPU-based Systems on the Path to Exascale," 6 June 2019. [Online]. Available: https://www.mcs.anl.gov/petsc/meetings/2019/slides/mills-petsc-2019.pdf. [Accessed 27 April 2020].

[67] "SLATE home page," [Online]. Available: http://icl.utk.edu/slate/. [Accessed 27 April 2020].

[68] J. Kurzak, P. Wu, M. Gates, I. Yamazaki, P. Luszczek, G. Ragghitanti and J. Dongarra, "Designing SLATE," Innovative Computing Laboratory, 2018.

[69] J. Kurzak, M. Gates, A. Charara, A. YarKhan and J. Dongarra, "Least Squares Solvers for Distributed-Memory Machines with GPU Accelerators," *Euro-Par 2019: Parallel Processing,* pp. 495-506, 2019.

[70] M. Gates, A. Charara, A. YarKhan, D. Sukkari, M. al Farhan and J. Dongarra, "Performance Tuning SLATE," Innovative Computing Laboratory, University of Tennessee, 2020.

[71] R. Harken, "OLCF Offers New Workload Capabilities with Slate Service," Oak Ridge National Laboratory, 27 March 2020. [Online]. Available: https://www.olcf.ornl.gov/2020/03/27/olcf-offers-new-workload-capabilities-with-slate-service/. [Accessed 27 April 2020].

[72] "MPI home page," [Online]. Available: https://www.mpi-forum.org/. [Accessed 27 April 2020].

[73] J. S. Vetter, "Contemporary High Performance Computing: From Petascale toward Exascale," in *Contemporary High Performance Computing: From Petascale toward Exascale*, CRC Press, 2013, p. 130.

[74] J. S. Vetter, "Programming Models," in *Contemporary High Performance Computing: From Petascale toward Exascale*, Boca Raton, Taylor & Francis Group, LLC, 2013, p. 130.

[75] Message Passing Interface Forum, "MPI: A Message-Passing Interface Standard Version 3.1," 2015.

[76] M. Schulz, "The Message Passing Interface: On the Road to MPI 4.0 & Beyond," November 2018. [Online]. Available: https://www.mpi-forum.org/bofs/2018-11-sc/intro.pdf. [Accessed 27 April 2020].

[77] "MPI 4.0," [Online]. Available: https://www.mpi-forum.org/mpi-40/. [Accessed 27 April 2020].

[78] N. Losada, P. Gonzáles, M. J. Martín, G. Bosilica, A. Bouteiller and K. Teranishi, "Fault tolerance of MPI applications in exascale systems: The ULFM solution," *Future Generation Computer Systems,* vol. 106, pp. 467-481, 2020.

[79] "OpenMP home page," [Online]. Available: https://www.openmp.org/. [Accessed 27 April 2020].

[80] L. Grinberg, C. Bertolli and R. Haque, "Hands on with OpenMP4.5 and Unified Memory: Developing Applications for IBM's Hybrid CPU + GPU Systems (Part II)," *Lecture Notes in Computer Science,* vol. 10468, 2017.

[81] "SOLLVE: Scaling OpenMP with LLVm for Exascale performance and portability," Brookhaven National Laboratory, [Online]. Available: https://www.bnl.gov/compsci/projects/SOLLVE/. [Accessed 27 April 2020].

[82] "OpenCL home page," [Online]. Available: https://www.khronos.org/opencl/. [Accessed 27 April 2020].

[83] "CUDA home page," [Online]. Available: https://www.geforce.com/hardware/technology/cuda. [Accessed 27 April 2020].

[84] R. Dietrich, R. Tshüter, G. Juckeland and A. Knüpfer, "Analyzing Offloading Inefficiencies in Scalable Heterogenous Applications," *High Performance Computing: ISC High Performance 2017 International Workshops,* pp. 457-476, 2017.

[85] Khronos Group, "Khronos Group Releases OpenCL 3.0," 27 April 2020. [Online]. Available: https://www.khronos.org/news/press/khronos-group-releases-opencl-3.0. [Accessed 27 April 2020].

[86] "SLURM home page," [Online]. Available: https://slurm.schedmd.com/. [Accessed 27 April 2020].

[87] "Enhancing Energy Production with Exascale HPC Methods," 29 August 2016. [Online]. Available: https://hpc4e.eu/sites/default/files/files/presentations/HPC4E_CARLA16-.pdf. [Accessed 27 April 2020].

[88] T. Sterling , M. Brodowicz and M. Anderson, "The Essential Resource Management," in *High Performance Computing: Modern Systems and Practices*, Cambridge, Elsevier Inc., 2018, pp. 146-171.

[89] "Torque home page," [Online]. Available: https://adaptivecomputing.com/products/. [Accessed 27 April 2020].

[90] T. Sterlin, M. Brodowicz and M. Anderson, "The Essential Portable Batch System," in *High Performance Computing: Modern Systems and Practices*, Cambridge, Elsevier Inc., 2018, pp. 172-190.

[91] "Altair PBS Professional home page," [Online]. Available: https://www.pbspro.org/. [Accessed 27 April 2020].

[92] "PBS Professional Commercial-grade HPC workload and resource management," [Online]. Available: https://resources.altair.com/pbs/images/solutions-zh-CN/PBS-Pro_Datasheet.pdf. [Accessed 27 April 2020].

[93] "Tooling up for exascale," [Online]. Available: https://www.nextplatform.com/2019/11/12/tooling-up-for-exascale/. [Accessed 8 May 2020].

[94] "MaX Project website," [Online]. Available: http://www.max-centre.eu.

[95] "ChEESE Project website," [Online]. Available: https://cheese-coe.eu.

[96] "Mont-Blanc 2020 Project website," [Online]. Available: https://www.montblanc-project.eu.

[97] "DEEP-EST Project website," [Online]. Available: https://www.deep-projects.eu/.

[98] "ESiWACE-2 Project website," [Online]. Available: https://www.esiwace.eu/.

[99] "EuroEXA Project website," [Online]. Available: https://euroexa.eu.

[100] "NEXTGenIO Project website," [Online]. Available: http://www.nextgenio.eu.

[101] "ESCAPE-2 Project website," [Online]. Available: http://www.hpc-escape2.eu/.

[102] "EPiGRAM-HS Project website," [Online]. Available: https://epigram-hs.eu/.

[103] "EXCELLERAT Project website," [Online]. Available: https://www.excellerat.eu/wp/.

[104] "EoCoE-II Project website," [Online]. Available: https://www.eocoe.eu/.

Annex 1 - Exascale projects

We present in this annex a subset of existing Exascale projects of interest. All information presented here is freely accessible and relies only on public sources such as the corresponding project websites. For each project, potential relevance to the HiDALGO project is discussed. Nevertheless, in order to assess applicability thoroughly a deeper analysis is required (which is not the point of this report), including dialog with the third party.

5.1 MaX

MaX [94] stands for Materials design at the Exascale. In preparing the flagship codes for the transition to Exascale, the project will address the technological challenges related to hardware architectures becoming more complex and heterogeneous. This requires a modernisation of the codes and the adoption of new programming models. To overcome the limitations of OpenMP, new features of the OpenMP5 standard (such as the taskloop construct) are considered. Besides OpenMP, other possibilities are being investigated, particularly FPGA approaches and one-sided communication techniques and frameworks like HPX.

Many solutions are being investigated in order to address the heterogeneity of the systems, including both open standards (OpenACC or the offload construct in OpenMP) and proprietary solutions (CUDA and CudaFortran, as well as Intel ONEAPI and AMD ROCm and HIP).

Relevance to HIDALGO

Since the HiDALGO project is fervent to Exascale approach, especially newly developed technologies and software solutions, the MaX project seems to be a perfect candidate to learn from. One of the lessons learnt should be that related to modern programming models, which enable to achieve the highest possible application performance. A very good example is that can be used is effective utilization of OpenMP5 when processes of the simulator operate on the same node. Moreover, whenever GPGPUs are available on HPC nodes tips for adept using the CUDA library can be very valuable for pilots' simulators yield.

5.2 ChEESE

ChEESE [95] stands for the Centre of Excellence for Exascale in Solid Earth and its role is to enable services such as urgent computing, hazard assessment and early warnings using

flagship simulation codes that run efficiently on upcoming pre-Exascale and Exascale European Exascale HPC system for the solid Earth community.

The main objective of the ChEESE project is to address scientific, technical and socio-economic Exascale computational challenges in the domain of solid Earth. This is done by preparing community flagship European codes to run efficiently and developing 12 pilot demonstrators that require Exascale computing that will serve as proofs of concept towards enabling future services on urgent computing, early warning forecasts of geohazards, hazard assessment and data analytics. This will allow users in the solid Earth community to access these codes and toolkits easily. The CoE will also provide specialist training on services and capacity building measures.

Relevance to HIDALGO

ChEESE and HiDALGO projects may benefit from cooperation on many levels. Taking into consideration that both are dealing on challenges relevant to global domains there are a number of facets, which can in the common point of attention. Issues related to the effective launch of tasks and optimization of data flow between processes may be of particular interest

Both are also aiming for Exascale HPC systems, which can be very informative in the aspect of the path chosen to achieve this goal.

5.3 Mont-Blanc 2020

Mont-Blanc 2020 [96] intends to pave the way to the future low-power European processor for Exascale. Following on from the three successive Mont-Blanc projects since 2011, the aim of Mont-Blanc 2020 is to trigger the development of a next-generation industrial processor for Big Data and HPC. The project will address three key hardware challenges in order to achieve the Exascale performance and power requirements: first, designing an efficient processing unit able to deliver large performance in terms of floating point computations; second, using an innovative on-die interconnect able to supply enough bandwidth to the processing units with minimum energy consumption; and, finally, having a high-bandwidth and low power memory solution with sufficient capacity and bandwidth for Exascale applications.

Relevance to HIDALGO

The HiDALGO project is very keen on the knowledge and practical experience that could be offered by Mont-Blanc 2020 in developing Global Challenges applications on newly emerged processors. Solutions related to increasing processor capabilities (e.g. interconnect bandwidth) seem to be of project interest especially that HiDALGO's pilots are struggling with huge data transmission during simulation procedures. Another key aspect, which is in area of HiDALGO interest, is related to development of applications that are power-awareness.

Technics and software solutions towards low-power memory utilization may affect how simulations are implemented by HiDALGO pilots.

5.4 DEEP-EST

DEEP-EST [97] stands for Dynamical Exascale Entry Platform - Extreme Scale Technologies and is the third member of the DEEP Projects family. It builds on the results of its predecessors DEEP and DEEP-ER, which ran from 2011 to 2017. The aim of all three projects is to develop a new breed of flexible, heterogeneous HPS systems to support a broad range of HPC and HPDA applications.
Their Modular Supercomputer Architecture (MSA) creates an HPC system by coupling various compute modules according to the building-block principle. Each module is tailored to the needs of a specific group of applications and all modules together behave as a single machine. They are connected through a high-speed network and operated by a uniform system software and programming environment, enabling each application to be distributed over several modules, running every part of the code on the best-suited hardware. DEEP Projects are using Co-Design to address two significant Exascale computing challenges: highly scalable and efficient parallel I/O and system resiliency. These challenges will be addressed through integrated development of new hardware and software components, fine-tuned with actual HPC applications in mind.

Relevance to HIDALGO

The HiDALGO project is dealing with the analogous challenges as DEEP-EST does, of course in the different (smaller) scale and range. Since coupling of HPC and HPDA is also in the core of the HiDALGO's pilots, resolutions how to define, a uniform and effective environment for processing could be of highest benefit for HiDALGO. Furthermore, the definition of co-design is also shared amongst both projects where resiliency and efficient I/O are stressed.

5.5 ESiWACE-2

ESiWACE [98] stands for Excellence in Simulation of Weather and Climate in Europe**.** Climate modelling groups and their partners from the HPC industry are working together in the project to improve workflows of weather and climate modelling to prepare them for running on the upcoming (pre-)Exascale supercomputers. Within the project, open HPC user-services to the Earth system modelling community in Europe will be provided in order to improve model efficiency and to enable porting of models to existing and upcoming European tier0 systems. One of the services will build on ESCAPE2 (https://projectescape.eu/) project results, which is developing a benchmark suite that isolates key elements in the workflow of the model to

improve performance and to allow a detailed performance comparison for different hardware. ESiWACE2 will extend this suite to represent a wider range of community models and test the performance on different HPC systems, including the pre-Exascale EuroHPC systems.

Relevance to HIDALGO

A core component of ESiWACE2 is the DYAMOND initiative. DYAMOND stands for DYnamics of the Atmospheric general circulation Modeled On Non-hydrostatic Domains and describes a framework for the inter-comparison of an emerging class of high-resolution atmospheric circulation models. As part of the DYAMOND2 set of experiments, ECMWF's IFS global model will be run for a test period at a resolution of approximately 5km. This data will be available on 75 model levels and will give pilot applications which couple with ECMWF's weather and climate data the opportunity to experiment with integrating such high-resolution data with their model. This could serve to act as a stepping stone for integrating with ECMWF's data at a future date when the IFS is planned to run operationally at a comparable resolution. Integrating with such large volumes of data will be a necessary step on the path to coupling with an Exascale system, a key component of HiDALGO's long-term vision.

5.6 EuroEXA

EuroEXA [99] focuses on the computing platform as a whole as opposed to just component optimization or fault resilience in order to co-design a platform capable for scaling peak performance to 400 PFLOP with a peak system power of 30MW. A balanced architecture for both compute and data intensive applications is envisioned, with a modular integration approach enabled by a EuroEXA processing unit with FPGA integration for data-flow acceleration. A PUE parity rating of 1.0 will be targeted through the use of renewables and immersion-based cooling. A key set of HPC applications from across climate/weather, physics/energy and life-science/bioinformatics domains will be used to demonstrate the results of the project through the deployment of an integrated and operation peta-flop level prototype hosted at STFC. Components will manage local failures while communicating with higher levels of the stack. EuroEXA aims to demonstrate its co-design solution by supporting both existing pre-Exascale and project-developed Exascale applications.

Relevance to HIDALGO

Some of development paths of EuroExa and HiDALGO are similar in respect of implementing data intensive applications that are feasible to reach utmost performance on balanced architecture. Both projects understand the significance of co-design role as one of the promising paths to efficient utilization of Exascale systems. The cooperation can be facilitated by the fact that climate/weather domain applications play a crucial part in scenarios development.

5.7 NEXTGenIO

The NEXTGenIO (Next Generation I/O for the Exascale) project [100], which ran from 2015 to 2019 and was co-funded under the European Horizon 2020 R&D funding scheme, was one of the first projects to investigate the use of Intel's new Optane DC Persistent Memory Modules (DCPMM) for the HPC segment in detail. A major challenge in achieving Exascale computing is the I/O bottleneck, where overall performance is limited by how quickly the system can read and write data. NEXTGenIO aimed to widen and ultimately eliminate this bottleneck by bridging the gap between memory and storage using the DCPMM technology, which sits between conventional memory and disk storage on a hierarchy of types of storage.

The main goal of the project was to build a system with 100x faster I/O than current HPC systems, with the DCPMM technology offering storage-type capacity at near-DRAM speeds. At the end of the project the potential of DCPMM was demonstrated in the context of two high-performance scientific applications in terms of outright performance, efficiency and usability for both its Memory and App Direct modes.

Relevance to HIDALGO

For ECMWF's IFS, it was demonstrated that a distributed object-store over NVRAM reduces the data contention created in weather forecasting data producer-consumer workflows. When running in ensemble mode, the IFS transfers data between the 52 forecast model instances acting as data producers and final product generation instances acting as data consumers using the Fields DataBase (FDB) library. The FDB is both a software library and a service which provides the final output stage for the I/O stack used by the IFS as well as controlling meteorological data output for other tools.

During the NEXTGenIO project the FDB was modified to enable the use of NVDIMMs, and to operate as a distributed system over a pool of storage nodes. The project prototype was then configured to use this NVDIMM-enabled distributed FDB. As a result, an improvement of 30x of the I/O throughput was observed compared with ECMWF's current operational file system, Lustre. Such an improvement in I/O performance demonstrates the ability of the FDB to scale sufficiently to enable efficient I/O capability at Exascale. This ability of the FDB to scale efficiently to Exascale will benefit HiDALGO, as the FDB is a core component of ECMWF's WCDA RESTful API, which will be used to deliver real-time forecast data to pilot applications within the project.

5.8 ESCAPE-2

ESCAPE-2 [101] stands for Energy-efficient SCalable Algorithms for weather and climate Prediction at Exascale and is the follow-on initiative of the ESCAPE project. The project focuses on three main sources of enhanced computational performance, namely: (i) developing and testing bespoke numerical methods, which optimally trade off accuracy, resilience and performance, (ii) developing generic programming approaches that ensure code portability and performance portability, (iii) testing performance on HPC platforms offering different processor technologies.

Relevance to HIDALGO

ESCAPE-2 will prepare a set of weather and climate domain benchmarks, which will be specifically tailored to pre-Exascale and Exascale HPC infrastructures. Such benchmarks will be of benefit for **HiDALGO** in paving the way for weather and climate modelling at Exascale, which will allow global weather models, such as ECMWF's IFS, to run at much higher resolution than today. Integrating with the resulting large data volumes that such modelling will produce will be a critical step on the path to coupling with an Exascale system, which is part of HiDALGO's long-term vision. Therefore, the progress and results of ESCAPE-2's benchmarking will be of interest to HiDALGO.

The project will also combine ensemble-based weather and climate models with uncertainty quantification tools originating from the energy sector to quantify the effect of model and data related uncertainties on forecasting.

5.9 EPiGRAM-HS

EPiGRAM-HS [102] is a three-year European Commission funded project started in September 2018. The aim of the project is to design and deliver a programming environment for Exascale heterogeneous systems to support the execution of large-scale applications. This will be achieved by extending the programmability of such heterogeneous systems with the use of GPUs, FPGAs, as well as HBM and NVM. The project aims to apply such extensions to both MPI and GASPI HPC systems. Finally, the project intends to maximize application development productivity in such heterogeneous environments by:

- providing auto-tuned collective communication
- a framework for automatic code generation for FPGAs
- a memory abstraction device comprised of APIs
- a runtime for automatic data placement on diverse memories and
- a DSL for large-scale deep-learning frameworks

Relevance to HIDALGO

Paving the for Global Challenges scenarios is paramount goal of the HiDALGO projects which participants are very keen on any solution this rather difficult subject. Especially programming environments, which could facilitate developing process, code generation frameworks for accelerators as well as tuning communication mechanisms, are in the area of HiDALGO interest.

5.10 EXCELLERAT

The aim of the EXCELLERAT project [103] is to provide a single point of access for expertise on using high-performance computing in the field of engineering. The project focuses on providing service solutions in the form of knowledge, computational power, and infrastructure necessary to address the ever-increasing complexity in both industry and research. In particular, the project will focus on offering support as increasingly complex engineering simulations advance towards Exascale. Of particular interest is supporting engineering solutions in the aeronautics and automotive sectors.

A core aim of the project is to offer an HPC-based service to assist in the time-consuming process of product development and improvement by simulating the product's characteristics and behaviour. The goal is to improve the product or production cycle through high-performance data analytics leading ultimately to shortened production times and quicker readiness for market.

Relevance to HIDALGO

EXCELLERAT and HiDALGO projects overlaps in some number of goals especially related to engineering solutions and access to HPC services. In the first matter HiDALGO participants already started collaboration on profiling the OpenFOAM application on HPC systems available in the project. OpenFOAM is Computational Fluid Dynamics (CFD) software used for simulation in the Urban Air Pollution pilot.

Scenarios offered in the HiDALGO are composed of many cooperating applications and stages. We may benefit from knowledge acquired by EXCELLERAT in the matter of optimization of product cycle implementation.

5.11 EoCoE-II

EoCoE-II [104] stands for Energy-oriented Centre of Excellence for computing applications and is the follow-on initiative of the EoCoE project. It aims to build on the expertise gained during the first project at the intersection of renewable energy and high-performance computing. Its core aim is to accelerate the digitization of future energy systems and to achieve this it will focus on five key energy sectors: wind, meteorology, materials, water and nuclear fusion. It will do so by re-designing selected application codes to enable them to exploit Exascale computing architectures.

In an interdisciplinary approach where technical expertise will complement the scientific challenge, the following goals have been set:

- enable modelling breakthroughs in renewable energy domains
- foster digitalization in energy technologies to reduce greenhouse gas emissions
- apply state-of-the-art cutting-edge mathematical and numerical methods, algorithms and visualisation tools to re-engineer modelling applications for Exascale computing platforms
- establish a single “stop-shop” to effectively exploit simulation codes
- encourage HPC best-practices and reduce the skills gap in HPC competencies
- support Europe to improve its competitiveness in carbon-free energy production through the use of HPC
- improving the know-how in applying European software tools and methods for Exascale computing

Relevance to HIDALGO

One of the obvious truths in Exascale computing is that energy consumption in this type of system must be strictly controlled. It drives us to conclusion that applications developed for this purpose must use a new tools and methods that consider power utilization. In order to facilitate the process of transition from present systems to Exascale ones, simulation applications must be re-engineered towards power-aware mathematical and numerical methods, algorithms and visualisation tools. This knowledge can be taken from EoCoE-II to HiDALGO pilots.