Developing Exascale Computing at JSC

E. Suarez, W. Frings, S. Achilles, N. Attig, J. de Amicis,
E. Di Napoli, Th. Eickermann, E. B. Gregory,
B. Hagemeier, A. Herten, J. Jitsev, D. Krause,
J. H. Meinke, K. Michielsen, B. Mohr, D. Pleiter, A. Strube,
Th. Lippert

published in

NIC Symposium 2020
M. Müller, K. Binder, A. Trautmann (Editors)

Forschungszentrum Jülich GmbH,
John von Neumann Institute for Computing (NIC),
Schriften des Forschungszentrums Jülich, NIC Series, Vol. 50,
ISBN 978-3-95806-443-0, pp. 1.
http://hdl.handle.net/2128/24435

Developing Exascale Computing at JSC

Estela Suarez[1], Wolfgang Frings[1], Sebastian Achilles[2], Norbert Attig[1], Jacopo de Amicis[1], Edoardo Di Napoli[1], Thomas Eickermann[1], Eric B. Gregory[1], Björn Hagemeier[1], Andreas Herten[1], Jenia Jitsev[1], Dorian Krause[1], Jan H. Meinke[1], Kristel Michielsen[1], Bernd Mohr[1], Dirk Pleiter[1], Alexandre Strube[1], and Thomas Lippert[1]

[1] Jülich Supercomputing Centre (JSC), Institute for Advanced Simulation (IAS), Forschungszentrum Jülich, 52425 Jülich, Germany
E-mail: {e.suarez, w.frings, th.lippert}@fz-juelich.de

[2] RWTH Aachen University, 52062 Aachen, Germany

The first Exaflop-capable systems will be installed in the USA and China beginning in 2020. Europe intends to have its own machines starting in 2023. It is therefore very timely for computer centres, software providers, and application developers to prepare for the challenge of operating and efficiently using such Exascale systems. This paper summarises the activities that have been going on for years in the Jülich Supercomputing Centre (JSC) to prepare the scientists and users for the arrival of Exascale computing. The Jülich activities revolve around the concept of modular supercomputing. They include both computational and data management aspects, ranging from the deployment and operation of large-scale computing platforms (*e. g.* the JUWELS Booster at JSC) and the federation of storage infrastructures (as for example the European data and compute platform Fenix), up to the education, training and support of application developers to exploit these future technologies.

1 The Exascale Challenge

The numerical simulation of scientific and industrial challenges has become a cornerstone of science and engineering. Sophisticated algorithms are required to deliver solutions that are accurate, fast, safe, and affordable. This traditional modelling is nowadays increasingly accompanied by and fused with data-driven modelling and simulation and complemented by artificial intelligence techniques. As a consequence, the current scientific evolution relies increasingly on advanced computing technologies to process, analyse, and translate information into knowledge and technological innovations.

This observation was the starting point for the latest Scientific Case[1] of PRACE, the Partnership for Advanced Computing in Europe, which conducted a detailed study on a number of areas of major scientific, technical and societal relevance for Europe, including fundamental sciences, climate, weather and Earth sciences, life sciences, energy, infrastructure and manufacturing, future materials and many more. In all these areas, Exascale computing will for sure be required to reach major breakthroughs, such as climate predictions with sub-kilometre resolution, the design and manufacturing of accurate digital twins for new materials, the understanding of molecular biology on biologically relevant time scales, the analysis and simulation of the human brain at the micrometre scale along with the translation of results to advanced medicine and brain inspired AI, up to fully simulating the formation of galaxies and black-holes to advance the understanding of our origin and future.

Exascale Missions

As a consequence of such analyses and conclusions being performed worldwide, High Performance Computing (HPC) systems able to execute 10^{18} floating operations per second will soon be deployed around the world. The first system is already planned for 2020 in China, with the US and Japan following in 2021. In Europe the EuroHPC Joint Undertaking (EuroHPC JU[a]) – an initiative supported already by 29 countries – aims to deploy two Exascale systems starting in 2023.

Germany's Gauss Centre for Supercomputing (GCS[b]) is a leading candidate in Europe to participate in EuroHPC's initiative for Exascale systems. The three GCS partners, together with their funding ministries, the German Federal Ministry of Education and Research (BMBF) and the corresponding ministries in the GCS partner hosting states Baden-Württemberg, Bavaria and North Rhine-Westphalia, are currently working on a participation model competitive under EuroHPC's regulations.

A large part of the mission of the JSC at Forschungszentrum Jülich (FZJ) is to collaborate with hardware and software vendors, including application experts, to conduct research in the field of supercomputer technologies by means of co-design. JSC's specific goal is to advance the novel Modular Supercomputer Architecture (MSA) for the benefit of future computing. The MSA relies on the principle of composability. On the one hand, it allows to integrate future computing modules, like quantum computers, on the other hand, its major advantage is its ability to operate disaggregated resources, which allows to include boosters of highest scalability that are more energy efficient, compute effective and affordable than previous technologies.

To achieve this, JSC has launched the Dynamical Exascale Entry Platform (DEEP[c]) series of EU-funded development projects already in 2011 and is leading them ever since. In addition, JSC is involved in the European Processor Initiative (EPI) for the creation of a European processor. As far as future computing technologies are concerned, JSC is very active in the European Quantum flagship and will host its OpenSuperQ prototype system. JSC's modular architecture and the entire data centre are designed to integrate high-end computing and data analytics with AI and quantum technologies in order to achieve highest flexibility and benefit from a EuroHPC Exascale supercomputer.

To give some examples as to Exascale applications, JSC has become the scalability spearhead of German Earth system modelling for both data and simulation applications, is a founding member of the European Human Brain Project (HBP), where it leads the HPC simulation and data activities, and is deeply liaising with materials sciences. For these fields, JSC creates joint research labs with the respective communities, anchored in its institutional funding program.

Operational Challenges

Let's take a look at the challenges an Exascale system poses to facility operations by inspecting and extrapolating the Top500 list of the fastest supercomputers worldwide. Summit – the number one at the time of this writing located at the Oak Ridge National Laboratory – achieves a peak performance of about 200 PFlop/s with a power consumption

[a] `https://eurohpc-ju.europa.eu`

[b] `https://www.gauss-centre.eu`

[c] `http://www.deep-projects.eu`

of 13 MW on about 520 m^2 usable floor space. A peak-Exaflop-system with the same technology would require more than 60 MW of electrical power and an area of $\sim$10 tennis courts (without GPUs the power and space requirements of the Summit technology would more than triple). Clearly, technological advancement improves power efficiency and integration density. So is the Japanese Fugaku system at RIKEN's R-CCS going into operation with about 500 PFlop/s in 2020, based on new far advanced Fujitsu ARM CPUs, estimated to consume between 30 and 40 MW, *i. e.* between 60 and 80 MW when extrapolated to ExaFlop/s.

Obviously, energy efficiency and integration density will be most crucial aspects of any Exascale system design. As a consequence, extensive infrastructure research and preparations will be needed. Moreover, measures to lower operational costs - *e. g.* through the use of warm water cooling technologies with free-outdoor cooling - will be mandatory. In these areas, JSC has collected broad experience through the operation of prototype systems (*e. g.* DEEP) and Petascale community servers (*e. g.* QPACE in the field of particle physics).

Heterogeneity Challenging Programmability

The emergence of cluster computing, starting as a disruptive technology around the year 2000 with components-off-the-shelf based on standard general purpose processors, has uplifted supercomputing from the Teraflop/s to the Petaflop/s era. However, due to the relatively high power consumption of general purpose CPUs relative to their peak performance, Exaflop-capable clusters built in this manner have extremely high energy requirements. The 3 to 5 times higher Flop/watt ratio of accelerator processors, however – of which GPUs are the most commonly used – made them very attractive and since 2010 has led to cluster systems that are heterogeneous on the node level, combining CPUs with accelerators, but are built as a homogeneous or monolithic system architecture. In fact, all published Exascale roadmaps foresee such heterogeneous nodes with monolithic system architectures – with the very new exception of the purely CPU-based Fugaku system (Japan) as to its node architecture. JSC's strategy for Exascale computing is unique: it is based on the MSA, which is able to combine CPUs and accelerated resources at system level (see Sec. 2).

Data *vs.* Flops

Despite all above arguments as to Exaflop-performance, it would be wrong to drive the Exascale race purely based on the number of floating point operations per second. Efficient data management is critical, especially now that deep learning (DL) is increasingly being used to generate knowledge directly from data rather than from theory or by modelling the data in order to simulate the phenomena based on such models.[1]

It is evident that data-driven approaches lead to applications requiring drastically increased data storage and I/O bandwidth capabilities. These codes and their users need new data-management services as well enabling them to exploit long-term storage and to share their data across sites and communities (Sec. 3 shows some developments driven at JSC to tackle these requirements). Data-intensive applications also benefit from hardware optimised to train neural networks in order to classify data and predict properties.

In particular, systems providing high performance for reduced precision (16 bit or 8 bit integer) operations in the form of tensor products on GPUs are very advantageous for such codes. These requirements, together with the need for high computing performance within a limited power budget call for modular system designs.

Application Challenges

It is becoming increasingly urgent that HPC applications take into account the complex architectures of (future) Exascale systems. A decisive factor will be the correct use of multi-level heterogeneity in parallel programming. The approaches implemented so far in a number of applications, which complement the MPI-based parallelisation with a further parallelisation level on the node, mostly via OpenMP, must be further pursued and extended. Another central goal is the inclusion of programming models for accelerators such as GPUs, for which the traditional node-level strategy still offers only limited support.

The expansion of JSC's JUWELS system with a GPU Booster mid of 2020 represents a first cornerstone for modularity (see Sec. 2.2 for a description of JUWELS, Sec. 4.2 for JSC's activities in GPU porting, and Sec. 4.4 for some examples of applications being adapted to heterogeneous architectures). JSC has designed the JUWELS MSA in such a way that the Booster can be operated as an autonomous, accelerated, homogeneous cluster system with nodes of CPU-driven GPUs. At the same time, the JUWELS Cluster with its CPU nodes can be operated autonomously. In addition, the two systems can be operated as a single machine, with the ability to dynamically reconfigure partitions that can interact with each other in workflows or even within one code on both modules. In this way, the JSC is approaching disaggregation and composability without compromising the advantages of traditional approaches.

Deep Learning is Transforming Scientific Computing

GPU-based acceleration is crucial for the strongly uprising field of DL.[14] Training the models on very large amounts of data requires running in parallel across hundreds or thousands of GPUs to be able to carry out training in reasonable amount of time. Different challenges arising from the demand to massively scale up data-driven DL model training are highlighted in Sec. 4.3. At JSC, distributed training of different DL models over hundreds of GPUs has been performed since 2017, reaching scaling close to linear without loss of task performance. Recently, Exascale-size distributed DL model training was demonstrated, using only GPUs.[2] Further development will be to transfer such successes of distributed training techniques to training of very large models that require heterogeneous accelerators, for instance, when coupling complex physics-based simulations or virtual environments and DL neural networks within closed-loop, end-to-end training pipelines. Modular Exascale supercomputers will provide the hardware and software infrastructure necessary for undertaking this effort.

The research focus and long-term agenda driven by JSC in the AI field aims on enabling large-scale self-supervised, multi-task and active continual learning. This line of research pursues to create methods capable of growing generic models from incoming streams of diverse data, extracting knowledge and skills quickly transferable across different tasks and domains – a grand and open scientific question. Executing and maintaining a continual

learning system over many weeks or months will be the task of future supercomputing centres and require performances and robust stability such as targeted by modular Exascale architectures.

The topics presented are intended to give an impression of the complexity and challenges of Exascale computing for data centres and their users. Aware of its responsibility as a candidate for the Exascale site, the JSC, together with user communities and partners from industry and academia, is making quite a number of efforts to prepare the operation of the first Exascale system.

2 The Modular Supercomputing Architecture

The JSC's strategy for Exascale is based on the MSA,[3,4] which has been developed in the DEEP series of EU-funded projects.[5] The MSA comprises a number of distinct compute modules (see Fig. 1). These modules can be potentially large parallel clusters of CPUs, CPU-accelerator nodes or AI-adapted nodes, whose hardware configuration meets the requirements of a particular type or part of an application, a large storage element, or a future computing system such as a quantum computer. Connecting all of these modules via a high-speed network and a unified software environment allows application codes to be distributed across different modules. This approach gives users full flexibility by allowing them to choose the right mix of computing resources. Of course such a system is well suited for workflows of different, increasingly complex, and emerging applications for a particular research issue.

To maximise the performance in this pool of computer resources, code adjustments are of course required. To support developers, JSC introduced specific support mechanisms to port code to acceleration technologies (especially GPUs) and promote the adoption of the MSA paradigm. In this context, a series of seminars on MSA began in 2019. Among the topics covered were the historical development of the MSA in the DEEP project series, the description of hardware systems used or to be used according to this paradigm, first application results on the modular JURECA system, specific use cases for machine and deep learning, *etc.* The aim of the seminar series was to raise awareness and pass on knowledge about the MSA to all members of the JSC. The high participation in the weekly presentations was a clear sign of the increased interest and triggered additional application porting activities around JSC codes (see some examples in Sec. 4.4), which serve as precursors for future actions for external code developers.

2.1 JURECA: The First Ever (Tier-1) Modular Supercomputer

The JURECA Cluster module put into operation in 2015 was completed with the installation of the Booster module in early 2018.[6] While the cluster uses multicore processors (Intel Xeon *Haswell*) and 100 Gb/s Mellanox (EDR) InfiniBand, the Booster uses multicore processors (Intel Xeon Phi *KNL*) and 100 Gb/s Intel Omni-Path. Bridging the two different high-speed connection technologies is possible in JURECA through a customised development in the ParTec ParaStation Software Suite[d] and is continuously researched and optimised.[7] While the use of different processor technologies for clusters and boosters is a

[d] `http://www.par-tec.com/products/parastationv5.html`

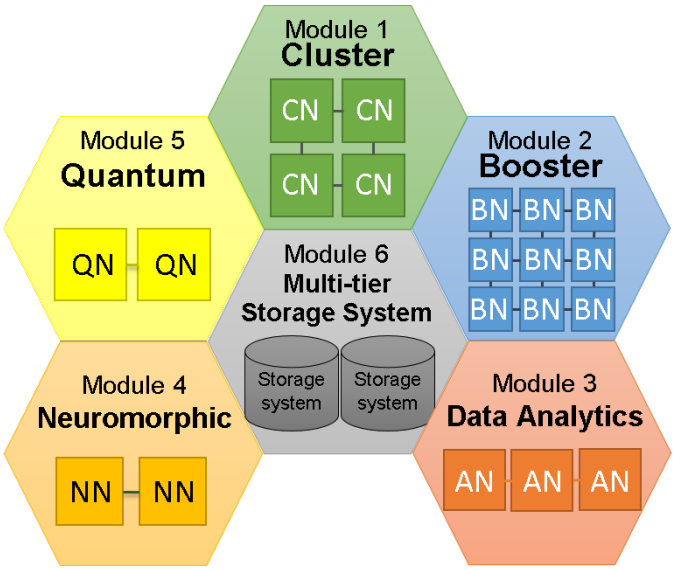

Figure 1. Sketch of the Modular Supercomputing Architecture. Note that this diagram does **not** represent a specific computer, but the general concept. Example modules: Cluster (*CN*: Cluster node), Booster (*BN*: Booster node), Data Analytics (*AN*: Data Analytics node), Neuromorphic (*NN*: Neuromorphic node), and Quantum (*QN*: Quantum node). The amount of modules in the future JSC Exascale platform and their configuration is subject to formal decisions.

key principle of MSA to optimise the highest scalability, it is also possible to benefit from MSA when considering the interaction of two or more partitions on the same machine.

2.2 JUWELS: The First Tier-0 Modular Supercomputer in the World

With the JUWELS system, JSC has further developed the MSA for the design of efficient, scalable and versatile high-performance platforms for national and European users.[8] It builds on the results of the DEEP projects and JURECA developments, with an emphasis on wider application. The Cluster module replaced the aging BG/Q system JUQUEEN in summer 2018, and with the Booster module planned for mid 2020, the modular system continues JSC's tradition of leadership class computing, particularly associated with the three past deployments of the BlueGene/L, /P and /Q generations. Provided by JSC as a GCS member, JUWELS serves as a national Tier 1 and European Tier 0 resource.

The configuration of both modules is summarised in Tab. 1. The JUWELS Cluster was designed to provide computing resources to a very wide range of application and to enable a swift and tight integration with the future Booster module, which targets workloads exposing a very high degree of parallelism.

The high-speed interconnects of Cluster and Booster are based on compatible Mellanox Infiniband interconnect technologies and will be directly integrated to enable easy and efficient inter-module MPI jobs without additional software routing. The system software stack based on the ParTec ParaStation Modulo is used to enable users to fully leverage the capabilities of the JUWELS modular supercomputer.

JUWELS	Cluster	Booster
Time of deployment	2018	2020
Project partner(s)	Atos, ParTec	Atos, ParTec, Nvidia, Mellanox
Node count	2559	>900
Peak performance	10.4 PFlop/s (plus 1.6 PFlop/s from GPU)	>60 PFlop/s
Accumulated mem. capacity	271.6 TB	>500 TB
Avg. power consum.	1.4 MW	2 MW (estimated)
CPU type	Intel Xeon "Skylake"	AMD EPYC "Rome"
Cores per processor	24	24
Accelerator	56 nodes with 4× Nvidia V100	4× Next-gen Nvidia GPUs per node
Network technology	Mellanox 100 Gb/s EDR-IB	Mellanox 200 Gb/s HDR-IB
Network topology	Fat-tree	DragonFly+
Cluster-Booster interface	Single global high-speed network domain	
Bandwidth ...		
... to capacity storage tier	250 GB/s	>400 GB/s
... to performance storage tier	>1 TB/s	>1 TB/s
Integration technology	Atos BullSequana X1000	Atos BullSequana XH2000
Cooling technology	Direct-liquid cooling with warm water (free cooling possible)	
System Software	ParaStation Modulo	
MPI	ParaStation MPI, Open MPI	
Job Scheduler	SLURM	
Resource Manager	SLURM + ParaStation psmgmt	
Others	Various compiler and optimisation libraries	

Table 1. Key features and characteristics of the modular Tier-0/1 supercomputer JUWELS.

It is again emphasised that the JUWELS modules can be operated autonomously without performance losses, *i. e.* as a pure CPU system for the Cluster and as a CPU-driven multi-GPU system for the Booster. The new dimension of modularity leads to the advantage of the combined use of both modules in overlapping workflows and split codes. This can be achieved with very little additional hardware effort.

2.3 From JUWELS to Exascale

The JUWELS system also serves as a prototype of JSC's vision for the first European Exascale system. Despite its much smaller size, JUWELS has the key components required to cross the Exascale limit in 2023 and shows an evolutionary path to an affordable and effective MSA-based Exascale device.

The exact configuration of the modular Exascale platform has not yet been determined, but as with JUWELS, it is expected that a cluster component with standard CPUs will be combined with a booster module characterised by an accelerator-oriented node design and a balanced network architecture. Further modules for data analysis applications could be integrated, while the possible use of disruptive technologies such as quantum computers and neuromorphic devices intended for inclusion in the modular system over a larger period of time is being investigated. A unified common software stack supports the execution of heterogeneous jobs across modules and facilitates applications to map their execution to hardware according to requirements. In addition, tight component integration and an efficient cooling approach will be critical to the cost-effective operation of the future Exascale platform. Following JUWELS' example, the Exaflop-computer will operate with high-density system integration technology and direct hot water cooling.

3 Data-Centric Activities

The Exascale era is by far not just about achieving a certain peak performance. In recent years, there has been a paradigm shift from computational to data-centric HPC applications, and scientists rely heavily on data services in their daily work. This applies both to large amounts of experimental data generated in large-scale facilities and to the mere exchange of everyday documents or intermediate data. The boundaries of these needs disappear over the data life-cycle and require tighter integration between data and HPC infrastructures. Important elements of this combined computing data environment are the secure delivery of the software environment required by data-driven applications, the homogenisation of access to data management services, the optimal organisation of storage diversity and storage technologies in advanced storage infrastructures, and the federation of such storage infrastructures across data centres to facilitate access to and sharing of data sources. The following subsections describe some of the activities carried out at JSC in these areas.

3.1 Containerisation

Containers – a lightweight virtualisation technology – are a convenient way to simplify the deployment of complex software stacks (*e. g.* for deep learning frameworks), increase portability for users using multiple systems, and place specific system software requirements over the bare-metal software stack of supercomputing centres. Today, JSC uses the runtime system *Singularity*[e] to support the execution of containers on its supercomputers. Singularity is a container runtime system designed with the security and functional requirements of HPC environments in mind. In early 2020, a comprehensive container infrastructure will be deployed and research into the most appropriate container support model for the future is underway. Options range from cloud-like *bring-your-own-container* to community-managed images. Pilot applications for containerised workload execution are currently being explored with selected JURECA and JUWELS projects and communities, and accessibility will be extended to the wider JSC user community by 2020.

3.2 OpenStack

JSC has complemented its Tier-0/1 HPC infrastructure with a versatile environment based on the OpenStack cloud platform, enabling users to deploy a variety of community-specific services, which can interact with data or even running applications in the existing HPC infrastructure. Most prominently, three types of infrastructure resources are available: compute, storage, and network.

To interact with the memory in the HPC domain, individual virtual machines of the OpenStack cluster can be connected to the HPC memory layer (see Sec. 3.3), which is also available as a file system on the HPC systems. It is now possible to use data generated in the high-security HPC domain – without compromising security – within the cloud environment and *e. g.* via a service. Conversely, it is possible to offer service interfaces for data interventions that are more sophisticated than a simple copy command. This can

[e] `https://singularity.lbl.gov`

help set up quality assurance processes before the data is finally transferred to a repository. In addition, the approach allows the combination of service-based, community specific environments with the computational HPC environment, a scenario that has been difficult or impossible until now.

3.3 JUST: Storage Hierarchy

Since 2015 the Jülich Storage (JUST) Cluster is the central storage provider for supercomputers at JSC.[9] Over the years, JUST has evolved rapidly in response to technological developments and increasing user demand for capacity, performance and access requirements. The use of the multitude of storage technologies – each with different functions and price categories in terms of capacity and bandwidth per euro – requires customised architectures. JSC has organised these storage options into layers or tiers that are characterised by the performance, capacity, and retention time of the target data.

At the top of the performance hierarchy sits the high-performance storage tier, an I/O acceleration system installed in 2019 to provide very high bandwidth using non-volatile memory technology. A unique feature of this tier will be the tight integration with the client systems, still maintaining a coherent name-space across computer platforms to facilitate high-performance workflows spanning multiple systems in the modular supercomputing facility. The mid-range tiers consist of the existing high-performance storage components – updated in 2018 to achieve $2\times$ the bandwidth and over $3\times$ the capacity – and a new capacity-focused storage cluster – providing a campaign storage file system `DATA` that enables online access to data for longer time frames. The storage hierarchy is rounded off by the archival service, which provides reliable long-term storage.

Portions of the `DATA` file systems can be readily exported to a cloud infrastructure to support the deployment of community data services. Utilising the same underlying storage system, JSC will provide an object-storage data management service in 2020. This service will be accessible to the users of the Fenix federation (see Sec. 3.4) as well as the users of the local facility.

JUST will continue to evolve to meet the increasing performance of the supercomputers and further expand its service portfolio to provide the required ingress and egress points, enable data management and sharing as well as support community services and platforms providing data access and analysis capabilities.

3.4 Data Federation

An important step towards facilitating sharing data between computer sites and communities is the federation of storage infrastructures. JSC is involved in various projects that target this goal at national and European level.

Helmholtz Data Federation (HDF)

In the HDF project, six Helmholtz research centres Alfred-Wegener-Institut (AWI), Deutsches Elektronen-Synchrotron (DESY), Deutsches Krebsforschungszentrum (DKFZ), FZJ, Helmholtzzentrum für Schwerionenforschung (GSI), and Karlsruher Institut für Technologie (KIT) have enhanced their computing and data-management facilities with considerable resource capacities and capabilities to satisfy the additional resource needs of a wide

spectrum of scientific domains. Two major components to satisfy these needs have been installed at JSC: an OpenStack-based cloud environment (see Sec. 3.2) and an extended capacity storage tier (see Sec. 3.3). These new elements complement the already existing HPC resources to build an infrastructure supporting the full life-cycle of data-centric science.

Helmholtz Infrastructure for Federated ICT Services (HIFIS)

Whereas HDF is targeted at data from large-scale scientific challenges, HIFIS[f] has been created to support a more general set of use cases. Eleven Helmholtz centres have partnered in this activity to offer cloud, backbone, and software services for a broader scope of activities. HIFIS aims to integrate services that already exist at a subset of the partner sites and make them available for a wider audience, sharing the load of offering individual services among the partners. At JSC, the existing HDF resources are made available in the HIFIS infrastructure.

Federated European Network for Information eXchange (Fenix)

The Fenix research infrastructure is a federated data and computing infrastructure with components geographically distributed across Europe.

Fenix, built since 2018 as part of the EU-funded ICEI (Interactive Computing e-Infrastructure) project, brings together five major European supercomputer centres: BSC (Spain), CEA (France), JSC (Germany), CSCS (Switzerland) and CINECA (Italy). Fenix combines interactive computing, cloud-based virtual machine hosting and a distributed data infrastructure with or close to supercomputing resources. This unique combination of services enables Fenix to support the construction of higher-level, community-driven platform services, such as an analysis or simulation workload that accesses large data pools controlled by a web front-end. The first user community of this infrastructure is the HBP, where resources in Fenix are available to European neuroscientists. However, Fenix is generally beneficial to European scientists through PRACE, as it offers a new program-driven access model with peer-review-based resource allocation. At the same time, the project will continue the development and deployment of key infrastructure components.

In order to enable a truly federated infrastructure, without introducing a strong operational dependency between centres, a decentralised approach with web and cloud technologies was chosen. A federated authentication and authorisation infrastructure is being devised that shall enable the use of multiple site and community identity providers.

Fenix data stores are separated into two different classes: Local *active data repositories* and *archival data repositories*. While the former are strictly site-local and typically optimised for high-performance, the latter are federated and exposed at all sites via the same type of object-storage interface. This enables distributed data access with a uniform user interface and consistent control of access to data objects.

While resources have been provided since 2019, several components of the Fenix infrastructure are in development and will be incrementally deployed in the next two years.

[f] https://www.helmholtz.de/en/research/information-data-science/helmholtz-federated-it-services-hifis/

This encompasses a centralised web-portal for the management of users and resources of the infrastructure as well as data movement and transfer technologies.

A key focus of the Fenix infrastructure is sustainability, both in the technical design and in the governance structure. Fenix is expected to be an important part of the European supercomputing and cloud strategy well beyond the end of the ICEI project in 2023. Fenix will be able to provide the adhesive between the geographically distributed EuroHPC systems in Europe, including an Exascale system at JSC.

4 Application-Related Activities

The increasing parallelism and complexity of HPC systems has motivated large user groups to modernise their entire code base and apply more flexible and portable programming strategies. At the same time, it has become clear to HPC architects that the requirements and limitations of end users must be carefully considered when building future HPC systems. This close interaction between system, software and application developers – coined with the term *co-design* – has therefore become a decisive element in the HPC landscape. Co-design, application modernisation and porting must be intensified in view of the expected high heterogeneity in Exascale systems on all levels, the desire for compatibility and the still unclear landscape of programming paradigms for support. Some of the activities carried out at JSC in these areas are described below.

4.1 Co-Design

The JSC team is actively driving technology development for Exascale in Europe, in collaboration with key industrial and academic partners. Projects such as the DEEP series, the European EPI[g], Mont-Blanc 2020, Maestro, or Sage are just some examples in which JSC plays a major role in the co-design strategy.

The DEEP projects – coordinated by JSC – have delivered a blue-print for a stringent co-design approach for system-level architectures. This approach included twelve application domains and over twenty individual codes. Their requirements, collected through numerous interviews and intensive co-design discussions, determined the components used in the hardware prototypes and the functionalities to be provided by the software stack, strongly influencing the evolution of the MSA over time.

JSC is also the driving force behind the co-design effort in the EPI project, which aims at developing processor technologies with IP-control in Europe and constitutes a key element of the EuroHPC JU Exascale strategy. A large selection of applications, mini-apps, and benchmarks have been chosen to be part of the EPI benchmark suite. In combination with processor-architecture models, simulators, and emulators the EPI benchmark suite will serve to evaluate design trade-offs and find the best chip configuration for the future HPC users.

4.2 Programming GPUs

The JUWELS Booster (see Sec. 2.2) will be the first large-scale GPU system at JSC, following the GPU partitions of the JUWELS and JURECA Clusters.

[g] https://www.european-processor-initiative.eu

Several programming models are available to enable the use of GPUs. The simplest way is to replace libraries with GPU-accelerated versions, *e. g.* using Nvblas during runtime as a GPU-accelerated drop-in replacement for a dynamically linked BLAS library, or using cuBLAS during compilation for the same effect. If libraries are not available, compiler directives can be used to mark the parts of the code that should run on the GPU, as it is done in CPU code when parallelising loops with OpenMP. In fact, OpenMP provides directives that enable offloading of code segments to GPUs. OpenACC offers an alternative set of directives that enable a higher level of abstraction specifically for GPUs. In both cases it is important to pay attention to memory management and data structures.

Nvidia's own programming environment CUDA is available for multiple languages including C++, Fortran, and Python. It provides tools and documentation, and it is the platform that exposes all new hard- and software features and makes them available first. Unfortunately, CUDA is limited to Nvidia GPUs. A more portable alternative is HIP, developed as an Open Source project by AMD. HIP is very similar to CUDA and compiles for Nvidia and AMD GPUs. Last, but not least, a number of programming models based on C++ programming paradigms (like functors and lambdas, C++ data containers, and basic algorithms) are currently developed to enable programmers to write code that will run efficiently and is portable on GPUs from different vendors and other parallel platforms including CPUs. Examples of these high level abstractions for C++ are Kokkos[h] and SYCL[i].

None of the models discussed above deal with more than a single node, but, luckily, they play nicely with MPI. The MPI versions to be installed on the JUWELS Booster module are GPU-aware. This means that they can transfer data directly from a GPU belonging to rank A to a GPU belonging to rank B. Depending on the locations of ranks A and B, the data is transferred in the most efficient way. This is transparent to the user, making MPI a useful model for programming multi-GPU systems both for intra-node as well as inter-node communication.

While the abundance of programming alternatives is indicative of the thriving GPU ecosystem, it may be hard for the programmer to decide which one to choose for their project. JSC offers courses for learning CUDA and OpenACC. Investigation of the other available programming models are also ongoing to be able to advise our users on the best choice for their applications and to extend the range of training that we can offer. Many of JSC's GPU activities are concentrated through a GPU-related Exascale lab, but consultation on a per-application basis is also provided. In addition, application-oriented high-intensity training for GPU-enabling is offered in the form of week-long GPU Hackathons hosted at JSC. At the multi-node level, a large amount of available parallelism needs to be expressed. As part of the *High-Q club*[j], applications have managed to scale up to about 2 million tasks and threads. This corresponds to the parallelism needed to feed about 100 of the GPU nodes. JSC plans to enable more applications to achieve this kind of parallelism and hopes to see some that will scale far beyond this.

JSC is not alone in its effort to target GPUs for highly scalable systems. CSCS in Switzerland, and the current and upcoming systems in the US are using GPUs to accelerate

[h] `https://github.com/kokkos`

[i] `https://www.khronos.org/sycl/`

[j] `https://www.fz-juelich.de/ias/jsc/EN/Expertise/High-Q-Club/_node.html`

their scientific applications. Many of the improvements made at other sites will benefit JSC's users as well. This is already obvious for users of GROMACS, AMBER, and of a variety of DL methods, who were able to take advantage of the new GPU nodes right away.

4.3 Deep Learning

Especially in DL[14] GPUs became the dominant accelerators used by all major software libraries like TensorFlow, PyTorch, MxNet and Chainer. A widespread class of learning models called convolutional neural networks (CNNs)[15] relies heavily on the efficient execution of very large sets of higher order matrix or tensor operations during learning, which are efficiently supported by GPU libraries such as cuDNN. State-of-the-art DL methods require extensive training on the available data. Even with the latest generation of GPUs, this takes a long time to create a suitable model and achieve a satisfactory level of performance for a particular task, such as visual object recognition or speech understanding. To train such a SOTA DL model, even a workstation equipped with the latest generation NVIDIA graphics processors (V100) will require a runtime of many hours to many days or even weeks, depending on the model type.

Acceleration can be achieved by increasing training to a large number of GPU nodes[10] without compromising trained model accuracy. This so-called distributed training requires efficient communication between the different nodes during learning. In this case, efficient inter-node communication is handled via CUDA-aware MPI routines implemented in low-level libraries like NCCL[k], which in turn can be used by high-level libraries like for instance Horovod[11] to implement efficient `allgather` and `allreduce` operations necessary for the updates of the learning model during distributed training.

At JSC, the Cross-Sectional Team Deep Learning (CST-DL) has employed Horovod since its appearance in 2017, enabling successful distributed training for different DL methods across hundreds of GPUs, reaching almost linear scaling without loss of model accuracy. Recent work suggests that scaling can be extended also to Exascale machines using the same methods.[2] Therefore, JSC is well prepared to perform distributed training of DL models over the thousands of GPUs available with the arrival of the JUWELS Booster. The next challenge will be to employ distributed training on MSAs that will contain heterogeneous accelerators.

With new computational demands put forward by effort to couple physics-based simulations and DL, parallelisation and scaling issues will face new challenges, as the type of computations to be performed efficiently may differ from component to component of such a closed-loop end-to-end simulation-AI pipeline. These challenges have to be met by the design of the MSA that has to provide means to orchestrate the parallel execution of such heterogeneous components in an efficient manner.

4.4 Porting Applications to Modular Supercomputers

A number of applications are being adapted at JSC to test them on the JURECA modular supercomputer (see Sec. 2.1), to demonstrate its capabilities, and to show, on the one hand, the possibility, and on the other hand, the advantages of mapping different parts of an

[k] `https://developer.nvidia.com/nccl`

application onto different compute modules of an MSA system. A selection of applications and results obtained in this area is described below.

4.4.1 1D-NEGF

The 1D-NEGF code, based on the Non-Equilibrium Green's Functions framework, is used to simulate inelastic quantum charge transport phenomena in semiconductors such as novel nano-transistors and quantum photovoltaic devices. The need of open boundary conditions to account for scattering effects, non local dependencies of the Green's function variables, and two nested self-consistent loops results in a large numerical problem, which demands a highly parallel and efficient code, scalable up to hundreds of thousands of cores.

The structure of the NEGF code has a number of distinct tasks exhibiting different computational properties. Tasks are divided into two classes based on their scalability, and the corresponding kernels are partitioned accordingly: Less scalable kernels are mapped to the Cluster and higher scalable kernels to the Booster.

The rationale for this Cluster-Booster mapping comes from the observation that cluster CPUs have fewer cores, higher frequencies, and are better suited to computing tasks that can degrade performance and scalability. On the other hand, booster GPUs have a much higher core count at lower frequencies and are better suited for highly parallel tasks. In a conventional design, the serial tasks set an upper limit for acceleration by Amdahl's law. In modularity, the division of tasks between cluster and booster nodes, which we call "modular execution", minimises the influence of serial tasks on scalability. This leads to an increased overall scalability of the application, better energy efficiency and an improved FLOPs/W performance ratio.

The modular design of the application requires a few further steps compared to the normal cluster-only workflow. Based on the same source code, an executable file with different architecture-dependent compiler options is created for each module. To run a modular job on an MSA platform such as JURECA, two separate jobs must be started simultaneously – one for each executable file – via a co-schedule mechanism.

Since communication between the modules is crucial, the modular implementation of the 1D-NEGF code performs a modular logic that splits the common `MPI_COMM_WORLD` into a sub-communicator for the Cluster, one other for the Booster, and an inter-communicator, which is used for inter-module communication and separating tasks.

The modular execution of 1D-NEGF so far has been tested on 1 Cluster node (Haswell) and by increasing the number of Booster nodes (KNL) to up to 64. The Haswell node has a theoretical peak performance of about 1 TFlop/s while a KNL node provides about 3 TFlop/s. The modular execution used only one additional Haswell node – meaning merely $960/(64 \times 3046.4) \sim 0.5\,\%$ additional resources in terms of peak performance – and remarkably turns out to be $6\,\%$ faster than the conventional execution running on the corresponding number of KNL nodes only.

4.4.2 xPic

xPic is a particle-in-cell code developed by KU Leuven as a new implementation of iPic3D[12] to perform kinetic simulations of non-collisional plasma. The main application field is space weather forecasting, which simulates the interaction of solar plasma with the Earth's magnetosphere in order to predict conditions on Earth and in orbit.

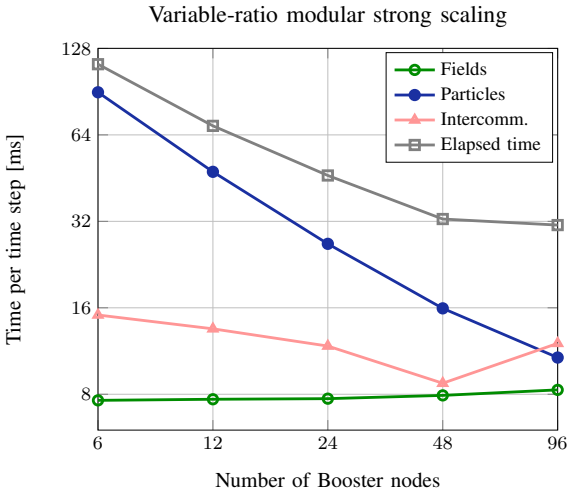

Figure 2. xPiC: strong scaling experiments running the field-solver on four nodes of the JURECA Cluster and the particle-solver on a varying amount of JURECA Booster nodes. In the last point of the graph a configuration is reached, in which a very similar amount of compute time is employed on Cluster and Booster, opening up the opportunity to largely overlap executions. This new option could not yet be exploited in traditional supercomputing.

The xPic code is essentially divided into two components: the field solver and the particle solver. Given the absence of interaction between the computational particles, each particle can be treated independently, making the particle mover in the second component embarrassingly parallel. This makes it particularly suitable for accelerators and co-processors. On the JURECA Booster, a Hybrid MPI/OpenMP approach is used to distribute work over the cores of the KNL processors, while using SIMD directives to exploit the large vector registers available. The field solver consists of a sparse linear solver, and it is expected to be less scalable than the particle solver. Therefore, on JURECA the field solver runs on the Cluster module. The Cluster-Booster implementation of xPic uses MPI (non-blocking point-to-point intercommunication) to exchange data between the field and particle solvers. Depending on the size of the problem and on the numerical algorithm, this intercommunication can take a substantial portion of the runtime.

In previous works it was shown how the particle solver could benefit from the KNL features.[13] More recent tests on JURECA with larger node counts highlight the different scalabilities of field and particle solver, and show how the flexibility of the Cluster-Booster approach could be used to control the scaling of the two sides of the applications independently (see Fig. 2).

4.4.3 Lattice QCD

The SimLab Nuclear and Particle Physics (SLNPP) group prepares LQCD codes for new and future architectures, focusing on GPU porting and MSA.

A large part of the effort to optimise LQCD codes for GPU-usage is based on QUDA[l], a CUDA-based library of LQCD kernels and sparse matrix inverters. Working together with QUDA developers (including Nvidia experts), the SLNPP team aims to identify further code hot spots which could be offloaded to GPUs.

In parallel to the GPU-focused activities, the SLNPP group is preparing community LQCD simulations codes to run on future modular supercomputers. As an initial step, the group has developed the *QMOD* library containing functions that can pass global data structures (*e. g.* a lattice gauge fields, or propagators) from one hardware partition to another. This library interfaces with the widely-used USQCD software stack[m] and its various architecture-specific back-ends, such as QUDA. QMOD should make it possible for anyone using a LQCD code linked against the USQCD software stack to develop and test concurrency schemes in a modular supercomputing environment.

5 User Support

Supporting users is a key element to reach the efficient use of the current HPC system and upcoming Exascale systems. In addition to offering training courses and workshops – described at the end of this section – direct contact with users and cooperation with them in solving problems and optimising application codes are essential.

The three GCS partners Höchstleistungsrechenzentrum Stuttgart, JSC and Leibniz-Rechenzentrum have developed a HPC application support structure as an evolution of well-established user support services, tailored to suit the need of HPC-users on Exascale systems. As shown in Fig. 3 the services are offered at four different levels.

In *support Level 1* the service desk is the most important component. Its staff and user administration can rely on a ticket system as the main tool to process service requests. More complex issues are handled at *support Level 2*. The close collaboration within JSC between HPC experts (cross-sectional teams and system administrators) and experts from various scientific fields (SimLabs) has proven to be an efficient strategy for solving problems or improving user applications by means of code analysis and optimisation. JSC provides tools and web-based services (LLview job-reporting[n]) which enable users to monitor performance and other HPC-relevant metrics of their jobs. In this way staff members as well as users themselves can detect if codes are consistently running inefficiently and initiate support actions. JSC provides in *support Level 3* detailed support in restructuring and optimising user codes, again using in-house tools like Score-P or Scalasca. Within the scope of joint research projects, the priority of *support level 4* is to establish long-term collaborations among the GCS centres, academia, and researchers. Every large-scale project is assigned a *mentor* who acts as a permanent point of contact and has a detailed understanding of the users project, its history, its resource consumption, as well as the projects associated challenges. The mentor coordinates communication between users and support staff and serves as a complementary service to the support offered at levels 2 and 3. The user profits from a more efficient and personalised support structure for his or her project while the centre benefits from more efficient use of valuable HPC resources.

[l] `http://lattice.github.io/quda`
[m] `https://www.usqcd.org/software.html`
[n] `http://www.fz-juelich.de/jsc/llview`

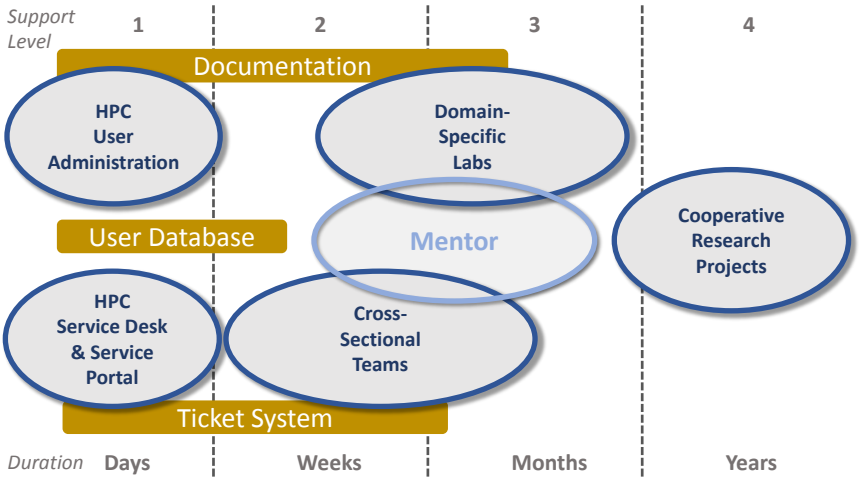

Figure 3. The new HPC application support structure developed by GCS. Level 1: support for basic functionalities and services; Level 2: more complex issues, *e. g.* application code analysis and optimisation; Level 3: detailed support in restructuring and optimising user codes; Level 4: long-term collaboration with users.

To be prepared for Exascale architectures and in particular for data-centric computing, JSC introduced a new user model for the HPC systems in 2018. In contrast to the former model, data management has been moved from a project-centric to a user-centric view that allows for more flexibility, and self-manageable and easier mapping of users to compute and data projects.

In addition to all these support vehicles, JSC provides a wide portfolio of training courses and workshops addressing all relevant topics of HPC from a user perspective. In particular, to foster the collaboration with users on performance optimisations, the VI-HPS workshop series on performance tuning and Porting and Optimisation workshops have been organised. In addition, extreme scaling workshops, Big Blue Gene Weeks and the High-Q Club – all initiated during the Blue Gene era at JSC – focus on extreme scaling of HPC applications.

Finally, JSC is member of the Performance Optimisation and Productivity Centre of Excellence (POP CoE[o]), which offers free performance assessment services for parallel scientific applications, performance measurement and analysis training (also in the form of webinars), and develops new and simplified methods to portably describe and assess the performance of parallel applications.

6 Outlook

JSC is preparing to establish itself as one of the first sites in Europe to operate an Exascale system. Although formal decisions are still pending, the JSC team takes up the responsibility to already prepare for the technical challenges that such a task poses for the data

[o] `https://pop-coe.eu`

centre and its users. In this context, a variety of activities are underway in the areas of hardware, software and application development. This paper describes a subset of important activities ranging from general code modernisation strategies for adaptation to modular systems including GPU programming to approaches aimed at coping with the flood of data emanating from existing applications. Areas and upcoming data-oriented approaches will contribute to the Exascale Computer and Computing Centre.

The overarching concept on which JSC's entire Exascale strategy is based is the MSA. Through the novel and composable arrangement of traditionally heterogeneous computing resources, it offers solutions adapted to a broader range of applications, from traditional, closely coupled simulations to emerging data analysis, machine learning, and artificial intelligence codes. The further development of the MSA, its implementation in the JUWELS Tier-0 system and the preparation of applications are core elements of the JSC Exascale strategy.

Disruptive technologies such as neuromorphic and quantum computer devices are unlikely to provide the computational and data management functions required to build an Exascale system alone and in a short time. However, they evolve rapidly and can already be used very efficiently to solve specific application tasks such as pattern recognition or problem optimisation. JSC's focus is therefore on enabling their integration into an HPC environment, facilitating their introduction, and investigating new use cases for these technologies. Here, too, the MSA concept plays a decisive role, as it allows *disruptive* modules without disturbing the operation of the *traditional* compute modules.

The next immediate step in JSC's Exascale strategy is to fully deploy the JUWELS Booster and help users leverage its capabilities in conjunction with the existing JUWELS Cluster. In parallel to the operational and support tasks, JSC will further develop the MSA and its software and programming environment, understand the specific requirements of the users and bring this knowledge into the design of the modular Exascale platform.

Acknowledgements

The authors gratefully acknowledge the funding provided by the Helmholtz Programme *Supercomputing & Big Data* to realise the JURECA Cluster and Booster, as well as the project funding provided by the Ministry of Education and Research (BMBF) and the State of North Rhine-Westphalia for the procurement of the JUWELS Cluster and Booster (SiVeGCS). Part of the research presented in this paper has received funding from the Deutsche Forschungsgemeinschaft (DFG) under Grant GSC 111. The European Community funded the research performed in the series of DEEP projects through the Seventh Framework Programme (FP7/2007-2013) and the Horizon 2020 (H2020-FETHPC) Programme, under Grant Agreements n° 287530 (DEEP), 610476 (DEEP-ER), and n° 754304 (DEEP-EST). This publication reflects only the authors' views. The European Commission is not liable for any use that might be made of the information contained therein.

References

1. *The Scientific Case for Computing in Europe 2018 2026*, PRACE, E. Lindahl (Editor), published by Insight Publishers, Bristol, UK, 2018, ISBN: 9789082169492, `http://www.prace-ri.eu/third-scientific-case/`

2. N. Laanait *et al.*, *Exascale deep learning for scientific inverse problems*, 2019, arXiv:1909.11150 [cs.LG].
3. E. Suarez, N. Eicker, and Th. Lippert, *Supercomputer Evolution at JSC*, Proceedings of the NIC Symposium 2018, K. Binder, M. Müller, A. Trautmann (Editors), 1–12, 2018, ISBN:978-3-95806-285-6.
4. E. Suarez, N. Eicker, Th. Lippert, *Modular Supercomputing Architecture: from idea to production*, Contemporary High Performance Computing: From Petascale toward Exascale **3**, J. S. Vetterm (Editor), CRC Press, 223–251, 2019, ISBN:978-1-1384-8707-9.
5. N. Eicker, Th. Lippert, Th. Moschny, and E. Suarez, *The DEEP Project - An alternative approach to heterogeneous cluster-computing in the many-core era*, Concurrency and computation: Practice and Experience **28**, 2394-2411, 2016, doi:10.1002/cpe.3562.
6. D. Krause and P. Thörnig, *JURECA: Modular supercomputer at Jülich Supercomputing Centre*, Journal of large-scale research facilities **4**, A132, 2018, doi:10.17815/jlsrf-4-121-1.
7. C. Clauss, Th. Moschny, and N. Eicker, *Allocation-Internal Co-Scheduling - Interaction and Orchestration of Multiple Concurrent MPI Sessions*, Advances in Parallel Computing **28**, 46–68, 2017.
8. D. Krause, *JUWELS: Modular Tier-0/1 Supercomputer at the Jülich Supercomputing Centre*, Journal of large-scale research facilities **5**, A135, 2019, doi:10.17815/jlsrf-5-171.
9. D. Krause, S. Graf, and O. Mextorf, *JUST: Large-Scale Multi-Tier Storage Infrastructure at the Jülich Supercomputing Centre*, Journal of large-scale research facilities **5**, A136, 2019, doi:10.17815/jlsrf-5-172.
10. R. Mayer and H.-A. Jacobsen, *Scalable deep learning on distributed infrastructures: Challenges, techniques and tools*, 2019, arXiv:1903.11314 [cs.DC].
11. A. Sergeev and M. Del Balso, *Horovod: fast and easy distributed deep learning in tensorflow*, 2018, arXiv:1802.05799 [cs.LG].
12. G. Lapenta, S. Markidis, S. Poedts, and D. Vucinic, *Space Weather Prediction and Exascale Computing*, Computing in Science Engineering **15**, 68–76, 2013, doi:10.1109/MCSE.2012.86.
13. A. Kreuzer, N. Eicker, J. Amaya, and E. Suarez, *Application Performance on a Cluster-Booster System*, 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), 69–78, 2018, doi:10.1109/IPDPSW.2018.00019.
14. Y. LeCun, Y. Bengio, and G. Hinton, *Deep learning*, Nature **521**, 436–444, 2015, doi:10.1038/nature14539.
15. A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Imagenet classification with deep convolutional neural networks*, Proceedings of the 25th International Conference on Neural Information Processing Systems **1**, 1097–1105, 2012.
16. M. Stephan, and J. Docter, *JUQUEEN: IBM Blue Gene/Q Supercomputer System at the Jülich Supercomputing Centre*, Journal of large-scale research facilities **1**, A1, 2015, doi:10.17815/jlsrf-1-18.