Available online at www.prace-ri.eu

Partnership for Advanced Computing in Europe

User's needs influencing HPC technologies

E. Athanasaki [a], N. Meyer[b], M. Cestari[c], A.Tuncer Durak [d], A. Tekin [d], P. Gschwandtner [e]

[a] <eathan@admin.grnet.gr> Greek Research and Technology Network , [b]<meyer@man.poznan.pl> Poznań Supercomputing and Networking Center, [c]< m.cestari@cineca.it > CINECA, [d]< a.tuncer.durak@uhem.itu.edu.tr , adem.tekin@be.itu.edu.tr > National Center for High Performance Computing – UHEM, [e] < philipp.gschwandtner@uibk.ac.at > Universität Innsbruck – UIBK

Abstract

The user requirements imposed by modern challenges are influencing future High Performance Computing (HPC) technologies and use cases. This report analyses a wide range of user requirements and new technologies and their impact on European and worldwide HPC trends, in particular in the PRACE and EuroHPC ecosystems, as well as HPC infrastructures provided by member countries. Applications that did not require the use of advanced computing or for which HPC solutions were not possible due to excessive costs and availability are now available in HPC and cloud computing. The way of using computing infrastructure and its services is changing, and due to its unification and decrease of costs, the popularity of HPC and cloud computing will continue to increase. The present variety of applications significantly exceeds the scientific community and also applies to industry, governmental use and security. An additional driving force of future HPC technology originates from the policies of the European Commission and member countries' regional needs. Big data analytics and artificial intelligence are among the identified applications, Interactivity in computing systems can also be essential for certain workflows. FPGAs and other specialized hardware or software packages are ideal for selected classes of applications. The above examples in conjunction with other requirements, outlined within this report, affects the design of HPC systems and can be used to justify the diversification of hardware and software set ups according to their target users.

DOI: 10.5281/zenodo.6483459

Table of contents

This technical report is part of a series of reports published in the Work Package "HPC Planning and Commissioning" (WP5) of the PRACE-6IP project. The series aims to describe the state-of-the-art and mid-term trends of the technology and market landscape in the context of HPC and AI, edge-, cloud- and interactive computing, Big Data and other related technologies. It provides information and guidance useful for decision makers at different levels: PRACE aisbl, PRACE members, EuroHPC sites and the EuroHPC advisory groups "Infrastructure Advisory Group" (INFRAG) and "Research & Innovation Advisory Group" (RIAG) and other European HPC sites. Further reports published so far on the PRACE webpage [1] cover "State-of-the-Art and Trends for Computing and Network Solutions for HPC and AI", "Data Management Services and Storage", "Edge Computing: An Overview of Framework and Applications", "Quantum Computing – A European Perspective" and "Security in an evolving European HPC Ecosystem".

[1] https://prace-ri.eu/infrastructure-support/market-and-technology-watch

1. Introduction

In recent years, there has been an increase in the popularity and importance of HPC technologies. This is due to several significant factors influencing the development of this technology:

- A significant increase in HPC capabilities thanks to the development of non-conventional computing technologies.
- New perspectives in computational science and the need to solve more complex problems.
- Lower costs of hardware, both HPC and data, combined with increased availability of HPC systems have resulted in more access time for users. This does not only refer to pre-exascale or exascale systems, but also petascale systems, which are now more widely available in Europe.
- The need to use alternative technologies such as GPUs, FPGAs or hybrid systems with quantum computing.
- Much wider openness to the economy and a significant increase of the stake of companies interested in HPC -from large corporate research to SMEs.

The above-mentioned factors open new possibilities in science, industry and society, which will form further requirements towards HPC (hardware and software) – new services, architectures and software development.

These and other factors were the basis for a simplified and shortened analysis of user requirements in terms of their impact on future hardware and software solutions.
The report undertook an analysis of user's needs that will influence several levels of the HPC ecosystems:

- Computing in the areas of science and economy - Chapter 5
- New application areas
    - AI and its importance in many application areas - Chapter 2
    - Big Data Analytics – Chapter 3
- Changing hardware solutions, including FPGA, GPU, quantum computing – Chapter 6
- Software solutions
    - Interactive computing - Chapter 4
    - New development environments - Chapter 7.

There is obviously a whole spectrum of other observations that may significantly affect the development of HPC technology and their inclusion is planned in the next edition of the report, along with a need for close cooperation of data infrastructure and processing, large-scale data collection in the area of IoT (Internet of Things) and completely new application areas such as intelligent agriculture and cybersecurity, or industry 4.0.
There are several new functionalities required by end users which will influence future development, including federated approach in HPC, integration of HPC and clouds or European Open Science Cloud (EOSC) requirements and other fields that require future computing.

New HPC systems requirements have emerged due to the continued development of computing trends such as Big Data analysis, Artificial Intelligence (AI), Machine Learning (ML), Deep Learning (DL) techniques, and Interactive Computing (IC).. This report presents how both software and hardware architectures are affected and can be adjusted to follow these trends to meet future needs.
Current AI and Big Data trends suggest: significant market growth, steady migration of users to HPC platforms, and an increasing need for specialized DevOps and SysAdmin teams. Integrating AI and Big Data in an HPC project is a crucial consideration for long term architectural requirements.
Fundamental layers of migration from "Classic HPC" to AI includes primarily software architecture, compute hardware, and storage hardware, which are also reviewed with exemplary solutions in chapter 2. It introduces AI and Big Data, challenges faced, proposed solutions and future considerations for the growing needs – see chapter 3. Interactive computing refers to the capability to gain interactive access to computing resources. In reality, this simple definition poses significant questions when applied in the context of high performance computing: what can be defined as "interactive access", what is the reasonable time frame to get response from the system, and what computing resources should be made available in such a short period. Chapter 4 introduces interactive computing, the interaction methodology, usage scenarios and service requirements. The requirements of industrial and commercial users, as well as government agencies, defence contractors and academic institutions are discussed in chapter 5. Their preference of on premise segments, cloud services, or hybrid HPC systems is attributed to needs related to data sensitivity and operational costs minimizing. Furthermore, FPGAs (chapter 6) as well as other hardware accelerators and software packages (chapter 7) can offer both superior energy efficiency and higher performance in selected classes of HPC applications.

2. Artificial Intelligence

The financial predictions for AI investments in the global market shares will exceed contribution from $3.9 trillion by 2022 to $15.7 trillion by 2030 [PWC,2019]. The future use of AI methods will gain traction across multiple business sectors, even those not directly related to it, since it was used as a central keyword in a lot of unexpected places and booths at the Supercomputing and ISC conferences for the last years [SC, 2020][SC, 2021] .

Business companies which are already leveraging HPC services in their processes are leaning towards AI use cases, mainly provided by HPC operators. Therefore AI computing will improve their resource utilisation and attractiveness in the HPC market by providing hybrid solutions. Moreover, companies that did not benefit from HPC services in the past are becoming even more compelled to become AI/HPC customers. Fields that will increase benefits from using AI methods are:

- Analytics
- Business productivity
- Process automation
- Automotive industry
- Healthcare
- Hyper-personalised retail
- IoT/Telecom
- Smart agriculture
- and Quantum Computing.

There is a growing shortage of AI specialists in these fields, as apart from expanding on solutions and technologies. Large companies dedicate their resources into AI specialized courses. They were a great chunk NVIDIA's and Intel's promotion materials, and it seemed to generate a lot of attention and interest. This will generate greater interest in using advanced and specialised technologies for AI problem solving. The training applications will initially use greater resources in order to refine models, which will eventually be deployed through applications that will require much less computing resources to tackle the problems they were trained to solve. Both architectures will optimise its behaviour to utilise the resources and minimize run time.

In case of processing power, interpretation of Moore's Law is reoriented from the idea of measure the performance of a single core's speed into leveraging:

- CPU core count
- GPU's
- ASIC's
- Edge Computing.

Scalability of the problem size is one of the most important factors for growth in HPC, which can be designed in two ways:

- Horizontally - by adding new hardware to existing clusters, or by software solutions like Map/Reduce algorithms for optimizing storage IOs in-place.
- Vertically - by replacing CPU-centred hardware for dedicated hardware listed above.

Efficiency of those particular approaches is presented in the logarithmic graph in Figure 1.

As more specific solutions tailored for particular use cases seem to be most proficient in ML, the ASIC chips are optimal for computing single types of DL algorithms, with CPU- and GPU-centred designs placed on the other corner of efficiency/reprogramming axes for more "general" AI usage.

Because AI is still considered an emerging technology, an ever-expanding spectrum of AI use cases result in dynamic development of dedicated hardware solutions per usage scenario. The current examples include:

- Groq - Tensor Flow Chips with compiler.
- Graphcore – 16 nm chips with 23.6 billion transistors on PCI-E/IPU with no programming - just tensor data used with PyTorch/ONNX as a "research platform".
- SambaNova Systems - Inference-ASIC chip for general applicability of next-gen compute beyond ML and 1st Reconfigurable Dataflow Unit (RDU) with 14 billion transistors, software-first approach, tile-based architecture used with PyTorch and TensorFlow.
- Corebras - WaferScaleEngine 8.5 inch chips with 400,000 cores with local memory (16 GB SRAM on chip), Python-based programming, and 20 kW per rack with 100 GBit Ethernet, which is already used in production for medical facility.

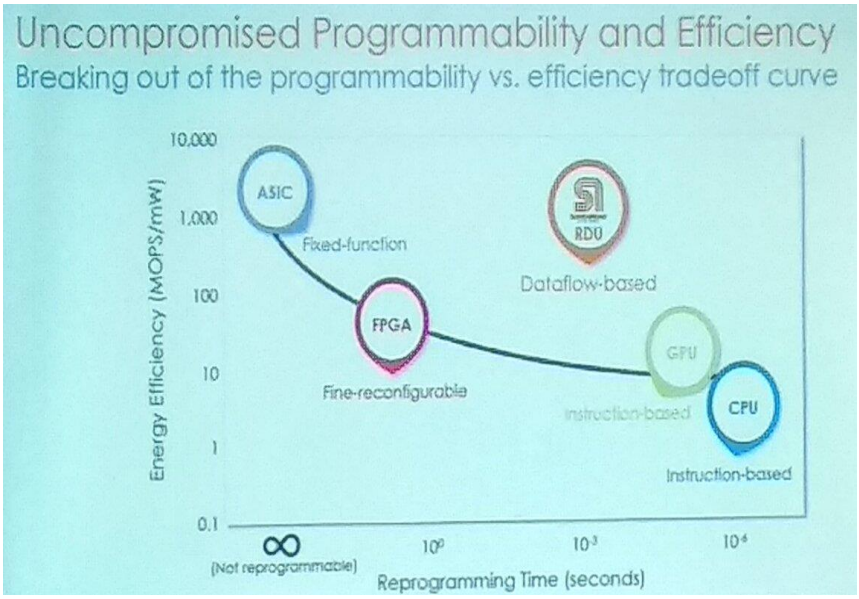

Figure 1: Uncompromised Programmability and Efficiency (Hock, Philippe Fricker and Andy. Excerpt from SC19 Forum Event titled "Building a Wafer-Scale Deep Learning System: Lessons Learned". [Source: https://sc19.supercomputing.org/presentation/?sess=sess369&id=exforum151#038;id=exforum151]

Trends for the future of AI in HPC underline four major design considerations:

- It is important to remember that AI and Big Data are tightly related. In most cases we need large datasets (natural or synthetically generated) for the purpose of DeepLearning.
- Specificity/Integration - Apart from common ML usages like Tensorflow and mostly image recognition methods, the AI solutions have to be planned and considered for dedicated problems with both hardware and software already in mind.
- Data Labelling - For ML to work, it needs data that is properly indexed. Such unstructured data has to be somehow categorized for context, unless its metadata can be identified from its content by ML itself.
- Workforce/Interpretability - Deep Learning seems to be a very distinct speciality to learn and thoroughly understand for practical use with all the tweaks and caveats. Therefore DL courses (like the ones marketed by NVIDIA or simplification/unification of the software layer by Intel's oneAPI programming model) are very helpful for building very much needed AI dev teams.

3. Big Data Analysis

Big Data is very tightly related with ML, hence its usage has similar benefits to these already mentioned in the previous chapter. Moreover, due to its growing market, it is used in a surprisingly wide variety of business sectors [INDUSTRY4.0, 2020].

Judging by prevalent promotion of Big Data and AI courses amidst SC2021 conference and booths, there also seems to be a growing niche for Big Data specialized teams with technical developers, analysts and administrators as this sector will keep expanding [SC, 2021] . The consequences of this fact should be the development of new programming environments and monitoring systems.

The value of aggregated data is still increasing, and will continue to do so as long as new ways of its practical usage are being invented. The trends of data curation will increase from 55 ZetaBytes in 2021 to 170 ZetaBytes in 2025 (Figure 2). To process the growing amount of data, new AI solutions are constantly developed.

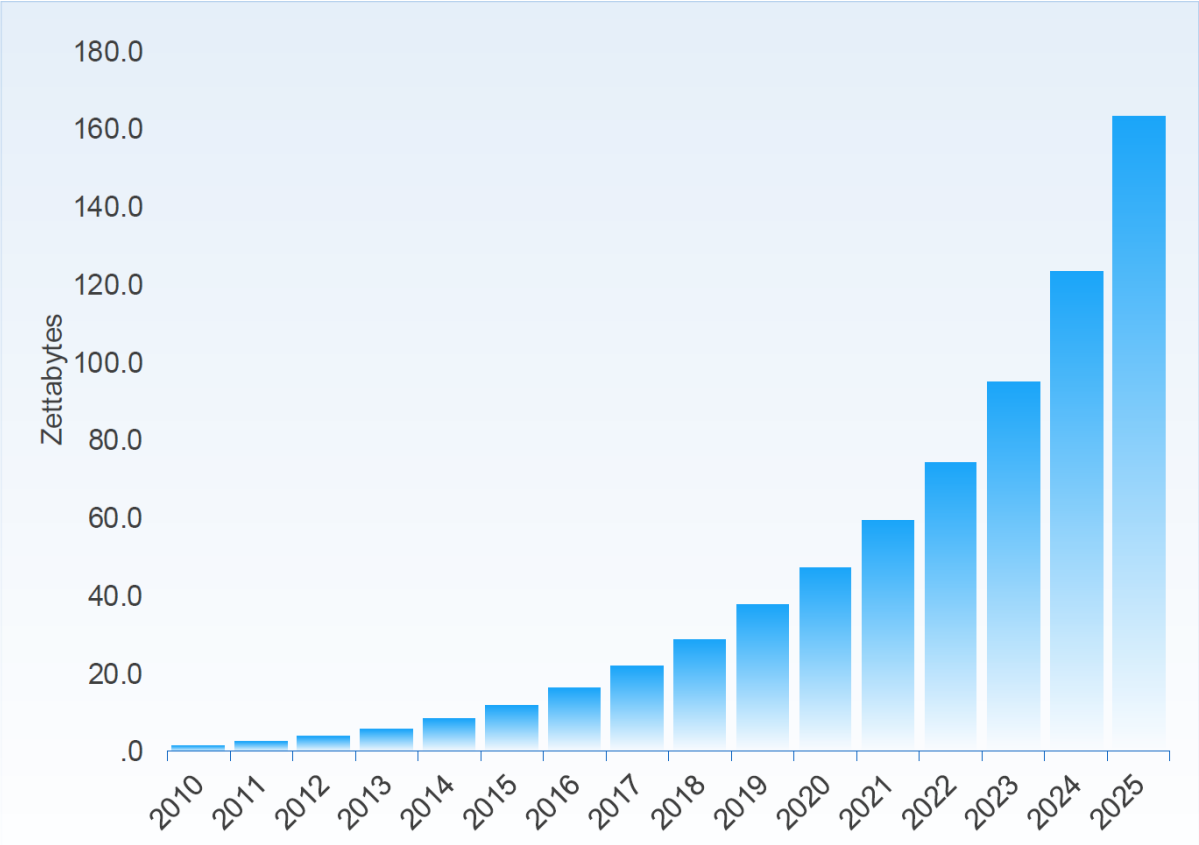

Figure 2: Worldwide global DataSphere creation and replication, 2010-2025, Source: IDC, 2021

Efficiency of data operations on a big scale is heavily dependent on the I/O throughput, and is usually optimized by both hardware and software layers. On the hardware side, apart from faster disks, hot/cold storage diversification is the most cost-effective method of speeding up data access with smart allocation prioritization. On the software side, apart from managing the aforementioned hot/cold hardware pools, pre-processing of data and queries is a good way of paralleling the load to multiple working nodes. There are solutions that have already been created with these two layers in mind, for example Intel DAOS (Hardware Tiering) or Hadoop (Processing Parallelization). However it will require further improvements due to the continuing speed-up of the data capacity size and the problem of delivering all necessary input data for computing both in HPC and HTC (High-Throughput Computing).

For multiple user-agents it is important to provide a unified API with ACID transactions (as for data operations assuring its Atomicity, Consistency, Isolation, Durability), which will also be the key for joining larger “meta-clusters” in the near future, and additionally shorten the skill-gap for usage and development.

The last problem is related to large chunks of unstructured data, as ML requires accurately labelled input tailored for a specific function. If provided data sources lack important metadata, and cannot be added manually at scale, then in some cases it should be possible to generate it by implementing some other ML model. If not, we have to ensure data labelling beforehand, during the collection phase.

When anticipating the future of Big Data, it has to be considered in correlation with AI, as these technologies are already related in a growing number of use cases. We can already witness the aforementioned issues receiving more attention, therefore a growing variety of solutions are being created to address these issues.

With AI-oriented design for Big Data clusters - incorporation of ACID transactions and unified API access for different use scenarios can prove to be very helpful.

Similarly, being aware of emerging structures of network/data/compute frameworks (like Edge Computing) may turn out to be crucial not only for AI but also for Big Data. Almost every category of IoT device usually has its own storage and provides an opportunity for different kinds of storage cluster expansion or data aggregation, which may become even easier to access by utilizing IPv6 networking. The influence of the mentioned problems will result in the integration of edge computing, clouds, and centralised highly efficient high performance computing technologies.

4. Interactivity in HPC

This chapter describes how the user's needs of interactive analysis, processing, and workflow steering impacts how modern HPC are designed and which additional services need to be provided. In this regard, and in the context of HPC, we can define interactive computing as a service that is able to provide, in a timely fashion, sufficient computing and storage resources and an interaction mechanism that allows users to interact with responsive HPC systems.

Traditionally users interact with high performance computing systems through a batch job system. Jobs submitted to the system are allocated on the system via a workload manager (WLM) application that has the primary goal to maximize the throughput of the HPC system while making sure that wait time remains within acceptable limits for all jobs. This usage model typically consists of the following elements:

- Users prepare their simulation jobs
- Jobs are submitted to the queueing system
- WLM receive the user requests, assign priority, and dispatch the job on computing resources
- Users wait for completion of the batch jobs to post process output data.

Employed for decades in the HPC field, this usage model reflects the significant cost of supercomputing systems and is justified by aiming at a very high utilization (over 90%) of the computing resources. This efficiency in use comes with a cost to users: the time required for launching a batch job and obtaining the results can be of the order of days depending on the load of the machine and the setting of the queues. While this time-frame is very reasonable for large simulations that need hundreds or thousands of computing servers to run for a large amount of time, such a long response time may not be adequate for modern workflows - composed of different simulation and processing steps that might require human intervention - and may reduce the capability of users to react quickly and manually steer the planned simulations.

Interactive Computing in HPC tries to cover the needs of modern workflows with a different usage model approach where the system is designed to respond promptly to user requests. This approach can be summarized in the following steps:

- Users ask for an interactive session
- The system provides computing resources in reasonable time-frame (minutes)
- Users have access to the interactive session.

Key aspects in the above usage model are the kinds of interactive sessions - in terms of computing frameworks and allocated resources – which are of interest for HPC user communities and the response time to obtain access to those computing resources. The latter aspect in particular is very dependent on HPC site productivity policies and service provisioning. Some relevant aspects to be considered in regard to configuring the interactive computing service from the system management standpoint are discussed later in this chapter.

Given the general framework just described, some practical use case scenarios are outlined below to demonstrate the benefit of providing interactive computing services with HPC systems.

- **Scenario I.** Visualization, processing, and reduction of large amounts of data especially when the processing steps of the workload cannot be standardized or implemented statically in the workflow.
- **Scenario II.** Use of interactive frameworks and scripting languages to complement the traditional compute and data processing applications running in batch, e.g. the use of R, Stata, Matlab/Octave or Jupyter Notebook, etc. The time spent in this activity is a non-negligible component in the "time to science". These frameworks not only play a role in analysing and post-processing the data generated from jobs, but can be used directly for on-the-fly monitoring and to pilot new HPC simulations [Stone, 2018]. In general, application workflows can become very complex and can be composed of arbitrary scientific tools, including non-traditional HPC tools [Stone, 2018].

A service capable of addressing the above scenarios would be in position to support workflows such as:

- Real-time interaction with a program runtime or running simulations
- Estimation of the state of a program and its future tendency
- Access and processing of intermediate results
- Steering the computation by manually modifying input parameters or boundary conditions as more input data (live data) comes into play
- Visualization and manipulation of large amounts of data.

In the future, other interactive computing use cases may need to play a major role in HPC as the complexity of the workflows developed in the scientific communities increases.

Applied to the HPC context, offering interactive computing service to users means thinking and designing the computing infrastructure with this goal in mind. Some key system requirements derive from the previously described usage scenarios. In practice, an interactive computing service in HPC would rely on fast (order of minutes) access to the following infrastructure resources:

- Capable computing resources (state-of-art servers)
- Data produced on HPC computing resources
- Capable storage infrastructure (high IOPS, high bandwidth and reduced latency) for data intensive workloads
- Remote desktop for GUI applications
- Interactive frameworks (e.g. Jupyter notebook, R, stata, Octave, Julia).

Examples of interactive computing service implementation have been carried out as part of the ICEI-Fenix infrastructure project (Interactive Computing E-Infrastructure for the Human Brain Project) [FENIX,2021] . For this project, some key design elements of the supercomputing infrastructure to implement the service were defined [FENIX, 2021b] .

Moreover, the system response time for providing access to computing resources is a critical aspect of the interactive computing service. This access time clearly depends on how the HPC system is designed and configured. For example, to reduce the access response time, a dedicated portion of a system could be reserved for interactive computing. This has the clear advantage of providing resources quickly within the range of the dedicated computing resources. On the other hand, sizing a dedicated partition correctly needs monitoring in order to provide an adequate level of service and avoid wasting computing resources. Moreover, from the system management standpoint, a dedicated portion of the system for interactive computing allows for the implementation of specific WLM productivity policies, such as job pre-emption or oversubscription, to improve the availability of the service to the end users. In general, an interactive computing service solution that could leverage a dedicated partition of the system, and possibly scale to the whole system – still providing an adequate service level - would be obviously preferable. The recommendation is to start with a flexible configuration and monitor the user requests and the service provided, adjusting the configuration over time.
Data analysis is a field that sees continuous evolution of applications and techniques. Therefore, designing and deploying a futureproof interactive computing service for HPC would inevitably require a reliance on an Open-Source project - with a large community of developers and users - providing a data analysis, processing and visualization framework. The interactive computing service should be designed to be flexible and able to seamlessly integrate the most recent developed features of such frameworks. In this way, users would benefit from up-to-date data processing features, while the service implementation and maintenance over time would only focus on adapting the framework and implementing it in HPC systems. A common solution to provide interactive computing services is JupyterHub [JUPYTER, 2021]. Started in 2015, this community project is strongly developed and supported. It can be easily installed and configured on front-end nodes of HPC systems in order to cover all the user needs and satisfy all system requirements previously reported.
JupyterHub is provided with multiple tools that allow interaction with either the batch scheduler, container orchestration, or another interaction mechanism to spawn interactive sessions. User interaction is completely web-based through the web browser, therefore easing the maintenance and eliminating operating system incompatibilities that may affect other interactive computing solutions. JupyterHub can be easily configured to communicate with identity providers to restrict services to authenticated and authorized users. Thanks to JupyterHub, users have access to multiple python-based computing environments (JupyterLab, Tensorflow, PyTorch, RAPIDS, etc.) that can be easily extended as needed. Via the Jupyter-server-proxy feature, natively provided with JupyterHub, it is possible to reach services running on the backend compute nodes, such as a flask instance or paraview web, via http. Moreover, Jupyter-server-proxy allow remote desktop capabilities via VNC or Xpra protocols through the web browser.

HPC centres have started to offer tailored solutions based on JupyterHub to their user's community [FZJ, 2021] . The authors believe that this trend will grow over time and many more HPC sites will deploy their own JupyterHub solution on the front-end nodes of their supercomputing systems (Figure 3).

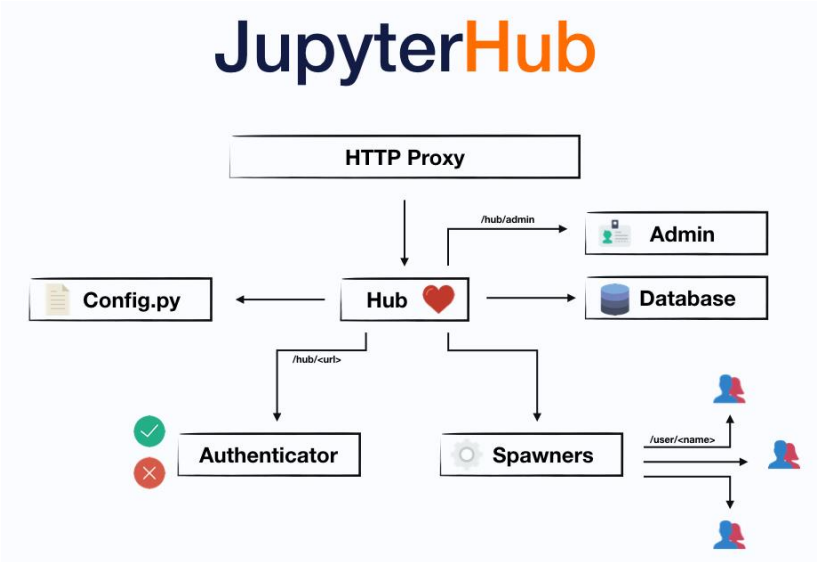

Figure 3: The core components of JupyterHub. JupyterHub is a set of processes that together provide a single user Jupyter Notebook server for each person in a group. [Source: https://jupyterhub.readthedocs.io/en/stable/]

5. The use of HPC by industry and government

The requirements of industrial and commercial users have been changing significantly over the last years for several reasons including:

- Continuous increase of the amount of analyzed data (Big Data analytics).
- The possibility of using artificial intelligence components such as machine learning, for more effective decision-making processes in many areas of economic and social life, like smart agriculture, environmental protection, medicine (extensive research program on COVID-19).
- The shift of business to the Internet, especially amid the pandemic, made cybersecurity and network protection of services a key element in securing new business models.
- Emergence of new areas using HPC as well as data and computing clouds, e.g. the aforementioned cybersecurity, smart agriculture, computer games market, use of supercomputers for training AI applications.
- Dissemination of HPC technology and connection with cloud services and hybrid HPC solutions.

The HPC market was valued in 2019 at 39 billion USD and it is expected to grow at a compound annual growth rate (CAGR) of 6.5% from 2020 to 2027. The growing need for high-efficiency computing, continued diversification and expansion of the IT industry, advances in virtualization and a rising preference for hybrid HPC solutions are some of the factors expected to fuel the growth of the HPC market. The ability of HPC systems to process large volumes of data at higher speeds is prompting government agencies, defence contractors, academic institutions, energy companies, and utilities to adopt HPC systems - which also bodes well for the growth of the HPC market [BlueWave, 2021]. The anticipated growth of the IT market will also influence several sectors of the economy, partially due to significant changes required by COVID-19. The market is expected to gradually pick up its pace from 2021 owing to the increasing demand for high-performance computing.

At the same time, the rising popularity of cloud computing [HPC, 2021] , coupled with the digitisation initiatives being adopted by various governments, would play a decisive role in catapulting the market growth over the forecast period. The cloud segment is expected to expand at the highest CAGR of 7.6% over the forecast period. Cloud deployment helps organizations in minimizing their operational costs as they do not have to invest in any additional computing resource or infrastructures. The market will benefit from enhanced efficiency of provided IT technology and cost-effectiveness which is also expected to fuel the growth of the cloud segment.

Currently, many insurers already use cloud platforms and services in the field of creating databases and data warehouses as well as big data analysis. Cloud computing is a great fit for digitisation in insurance companies, which increased significantly in 2020 due to the pandemic and lockdown. The solutions that could be launched immediately turned out to be especially useful as they require no hardware, software, installation, configuration or complicated implementation hosted and managed in-house. Thanks to them, insurers were able to react faster to the changing market situation and the needs of customers and stakeholders, while reducing the costs of IT infrastructure.

In 2020, there was a breakthrough in the way insurance companies embraced digitisation. Cloud computing has been an important part of this process. Previously, insurers viewed their digitazion of assets as only a test or backup environment. It is now a permanent part of business strategies. The trends since 2020 have shown that over time, insurance companies will use cloud services to an even greater extent, especially since there are more and more new tools and solutions for users. In addition, in big data analysis and statistics, the ad-hoc use of HPC is also foreseen but rather in a hybrid model including clouds and HPC.

On the other hand, the market for on-premise computing held the largest revenue share of 59.24% in 2019. This is attributed to the fact that governments remain keen on securing the sensitive data related to the national security and personal data of the citizens. Enterprises are also concerned about the protection of their respective organizational data. As a result, on-premise infrastructure is still being preferred over a cloud-based infrastructure by certain entities

The European Union's current priority is to recover the economy after the pandemic. Therefore, priority has been given to regulations favouring development and investments, including those supporting digital transformation. From the perspective of EU institutions, the use of HPC and cloud computing is one of the key factors increasing Europe's sovereignty. The European Commission is therefore working to establish a European industrial data and cloud alliance that will enable the development of several projects. These include joint investments in cross-border infrastructure, HPC but also cloud services, as well as unifying the framework of cloud regulations in the form of the EU Cloud Rulebook. There must be clear requirements for outsourcing agreements between financial entities and cloud service providers. That is why the European Commission is working on defining standard contractual clauses.

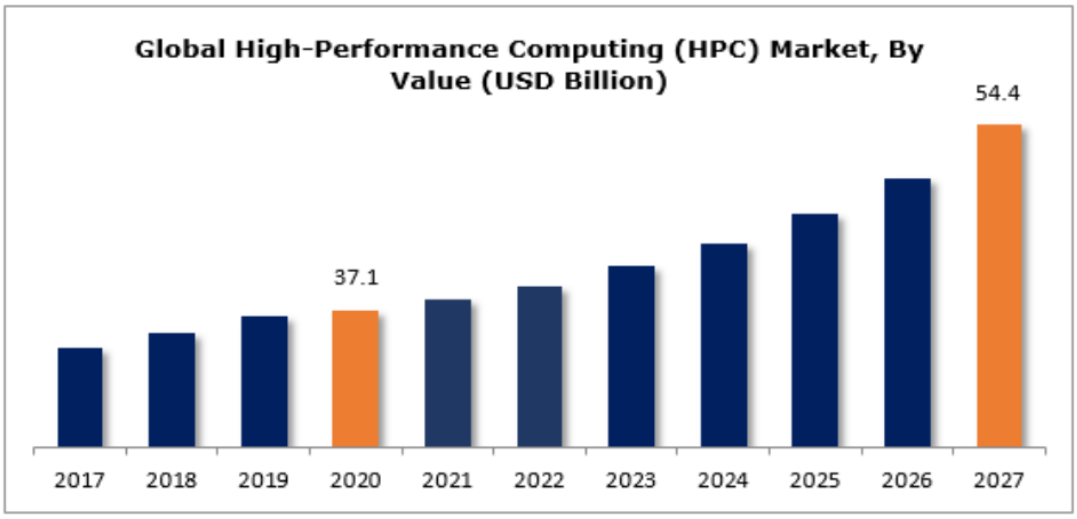

Figure 4: The anticipated growth of the global HPC market [Source; BlueWave]

Further analysis of industry requirements is built on interviews made on the market with 101 companies, which makes a good statistical overview of what is the current state:

- The know-how in industry
- Expectations towards the computing environment
- Areas where we find applications to be used directly at the HPC.

The survey was targeted at companies of manufacturing industries of Small and medium-sized enterprises (SMEs) and Midcap size including software and creative industry, construction and furniture production, production of industrial machines, production of consumer machines, automotive industry. Nearly 40% of the companies are using or considering the use of HPC technologies to improve their business. Looking at sectors represented by companies, software development, graphics, video and game development sectors are widely represented. When excluding software and creativity industry, manufacturing industries, mainly industries related to the furniture manufacturing, construction components and consumer goods production far outweigh other ones. Further analysis also recognised new branches, not taken into account during the first stage of analysis: institutions delivering smart agriculture technology (mainly governmental entities), the financial sector and insurance companies, as well as cybersecurity companies.
The goal of the analysis was set out to determine the awareness, capabilities and readiness to use HPC or correlated technologies like cloud computing by the companies. In addition, questions dealt with classes of software that companies use at work and the barriers preventing businesses from using these above mentioned technologies. The questions posed in the survey enable delineation of an overall direction for information and promotional effort

both for entrepreneurs (how technology can be useful to them), and infrastructure providers (how to advertise their services and what tools need to be shared).

Barriers and restrictions against the use of HPC technology should be treated as a necessary condition for improving the IT software and hardware on many levels for better use of HPC infrastructure by entities from many sectors of the economy:

1) Barriers to the use of technology HPC / HPCC

   The order of magnitude of production, which does not require the use of HPC technology was the most commonly chosen as the main barrier for companies that took part in the survey. Out of the 61 companies that indicated that they do not use this technology, 42 of them indicated that the reason is the scale of production, which does not require involvement of such solutions. Only 6 of these 42 indicated that they are not aware of the existence of such solutions, which leads to the assumption that the vast majority of these companies consciously decided not to use this technology in business development.
   The second barrier is low entrepreneurs' awareness of the existence and use of HPC and HPCC technologies. 23% of companies indicated that they do not know the technology and do not know how it can help them in doing business.
   On a similar level, another barrier was associated with fear of processing sensitive data with external parties and the possible loss and/or theft of data processed this way. Such a barrier was chosen by 20% of respondents.
   Financial barriers associated with the use of HPC and HPCC technologies were reported by approx. 10% of respondents
   The least likely indicated barriers to entry (<10%) were those related to the technological aspects associated with the use of HPC and HPCC technologies, i.e. lack of a technology partner who could introduce these new technologies to their businesses and the lack of skills necessary to operate the tools required to use computing technologies (Figure 5).

2) Recognizing opportunities for the increase of the company's position in the market through the use of HPC and HPCC technologies

   In terms of perceiving the possibilities offered by HPC / HPCC technology, 32% of respondents said that increasing the availability of HPC / HPCC will result in improving the position of their company on the market. 6% of respondents answer the opposite, and 62% do not have an opinion on this topic.
   The significant number of neutral responses can be explained by a lack of knowledge and low awareness of the benefits of HPC and HPCC technologies. This is confirmed by the fact that 85% of the group of respondents who reported a lack of opinion, also reported that they did not use or had not even considered use of this technology.

3) Ways of accessing HPC/HPCC infrastructure

   There are 3 options in the usage of HPC / HPCC services as the most preferable business model:
   - Own infrastructure – 74% of responders
   - The use of infrastructure commercially available on the market (on demand, eg. commercial clouds) – 33%
   - Constant cooperation with the supplier of technology and computing power – 23%.

   The results show that the predominant way to use the HPC / HPCC technology (for 74% of respondents) is the use of their own infrastructure, which may be one of the main reasons for the low popularity of this technology. This is because on-premise infrastructure is associated with high investment costs, which are justified mainly when a business forecasts heavy usage.
   The second most popular answer was the ad hoc use of infrastructure offered by suppliers in the market. This answer was chosen by 33% of respondents.

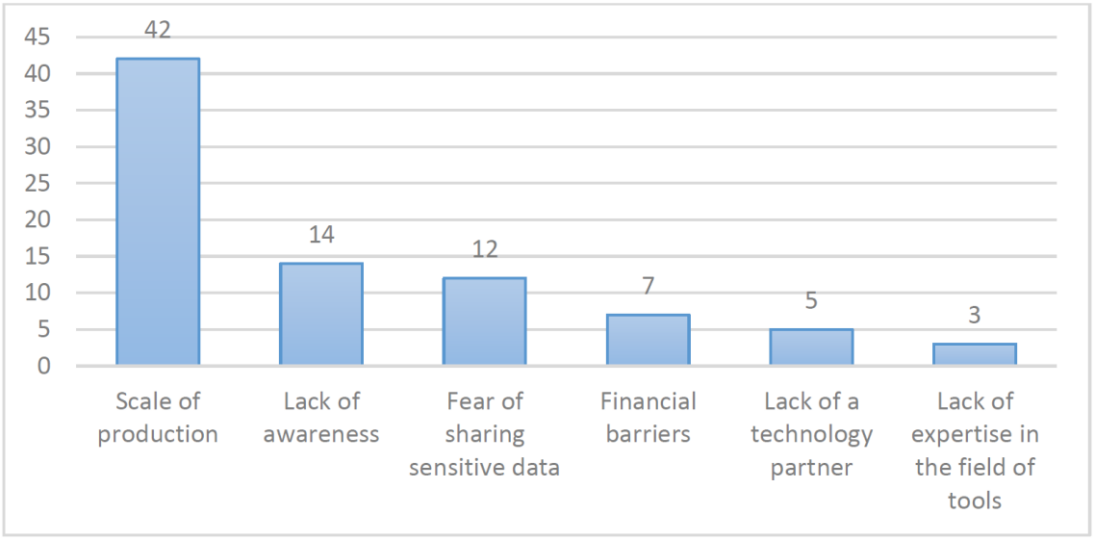

Figure 5: Most often indicated barriers to using HPC technology

4) Areas where entrepreneurs use (or intend to use) HPC and HPCC technologies
The research on survey results shows that nearly 70% of enterprises use high performance computing for design and prototyping. The second most popular area in which this technology is used is the analysis and interpretation of data. This can be explained by an increasing dependence of all industries on the amount of data being generated and processed. 46% of respondents admitted using or having the intention to use HPC for simulation purposes. What is surprising is the low percentage of companies (only 34%) that use HPC specifically to test their products. This element will probably require a more in-depth analysis. Slightly more than a quarter of respondents indicated that they use or intends to use HPC technology to evaluate their products and keep an eye on quality management.
Figure 6 describes the areas of business the companies use or intend to use technologies of HPC and HPCC.

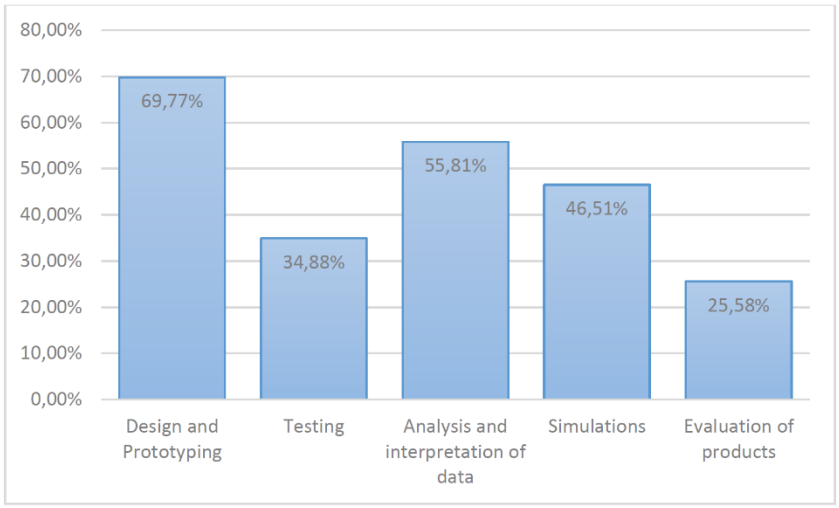

Figure 6: The most promising areas where companies use or intend to use HPC and HPCC technologies

6. Non-conventional hardware solutions (FPGA)

While GPUs emerged as accelerators for many HPC applications, often providing orders of magnitude of increased performance relative to CPU-only performance, they are still ill-suited for a number of use cases in the HPC landscape [BUSTIO, 2021] . They include applications exhibiting irregular parallelism, for example emerging sparse DNN (deep neural network) algorithms and/or algorithm implementations employing custom data types [ACM-DL, 2017] . But also specific classes of regular parallelism such as structured grid codes can benefit from employing FPGAs [KAMAL, 2021]. FPGAs (Field-Programmable Gate Arrays) are hardware devices that offer a programmable circuitry through hardware description languages (HDL). Therefore, these FPGAs provide

hardware technology that can be customized and adapted with respect to the target application they should run. For all the aforementioned classes of applications, FPGAs can provide superior energy efficiency and higher performance relative to GPUs.

While FPGAs are ideal hardware platforms for selected classes of HPC applications, they too have adoption barriers. One such barrier is requiring specialized tools, such as VHDL (Very High-Speed Integrated Hardware Description Language), to port applications to FPGAs. Another barrier is the lack of domain-specific library and parallelism support (e.g. BLAS, OpenMP), which has reduced their applicability to HPC in general, especially given the competing market of general-purpose computing on GPUs coupled with an ever-increasing plethora of specifically tailored tools and libraries. However, changes in FPGA hardware and software have increased their potential as well as their applicability to a wider range of HPC applications [KOBAYAS, 202].

Generally, HPC applications often require IEEE 754-compliant floating-point math for proper computation. With the inclusion of single-precision IEEE 754 units during the last decade, FPGAs have become an attractive competitor for many HPC applications. However, there are additional aspects besides floating-point math. For suitable applications, FPGAs can show a high energy efficiency compared to other hardware architectures due to their streaming architecture, in contrast to the classical von Neumann architecture of CPUs, FPGAs reduce the amount of data movement during program execution – a key factor in energy efficient computing. While not directly related to HPC user requirements, energy efficiency is an important factor in system design and maintenance. Furthermore, recent FPGA advances such as higher memory bandwidth via on-chip RAM as well as off-chip memory such as High Bandwidth Memory (HBM) specifically address increasing memory bandwidth requirements and further improve the generally high energy efficiency of FPGAs [INSIDE,2019] . These developments suggest attempts to lower the adoption barrier of FPGAs in HPC, as indicated by the example of Intel's multi-chip modules that hold both a Xeon Scalable Processor 6138P Gold alongside an Arria GX 1150 on the same package. Both chips are interfaced via UPI (Ultra Path Interconnect), which allows cache-coherent operation between both package domains for improved usability. Nevertheless, evidence seems to suggest generally slow adoption of FPGAs into larger HPC sites, as recent efforts are restricted to tier-2 HPC systems [TOP500, 2021] , [PU, 2021].

Future proofing FPGAs requires both hardware and software endeavours in order to solidify this technology in the future HPC market. In terms of hardware, the aforementioned efforts in increasing on- and off-chip bandwidth and data type-specific features coupled with research and development in clock frequency improvements and advances in process technologies will help keep FPGAs on par with GPU and CPU technologies. Regarding software support, a key factor in hardware technology sustainability, ambitions such as Intel's OneAPI and high-level programming models such as SYCL will help mitigate traditional adoption barriers [INTEL, 2021] . While other programming models based on OpenCL or Java exist, [BISPO, 2021], it remains to be seen whether one of them will become the state-of-the-art choice. Figure 7 shows a selection of popular SYCL implementations, including support for FPGAs by Intel's DPC++, CodePlay's ComputeCPP and XILINX' triSYCL implementations. Using its single-source programming model and designed as a C++-embedded domain-specific language, SYCL is able to provide a better integration of host and kernel code, enabling and facilitating use of a large number of existing debugging and performance tools. This improves upon the traditional drawback of new domain- or target-specific programming languages, which entail a the lack of user support due to the engineering effort involved in providing a mature toolchain for any new programming language. Based on pure C++, SYCL does not suffer from these issues. Despite its generality, SYCL still allows access to FPGA-specific optimizations such as loop pipelining or dataflow optimization [KERYELL, 2018] .

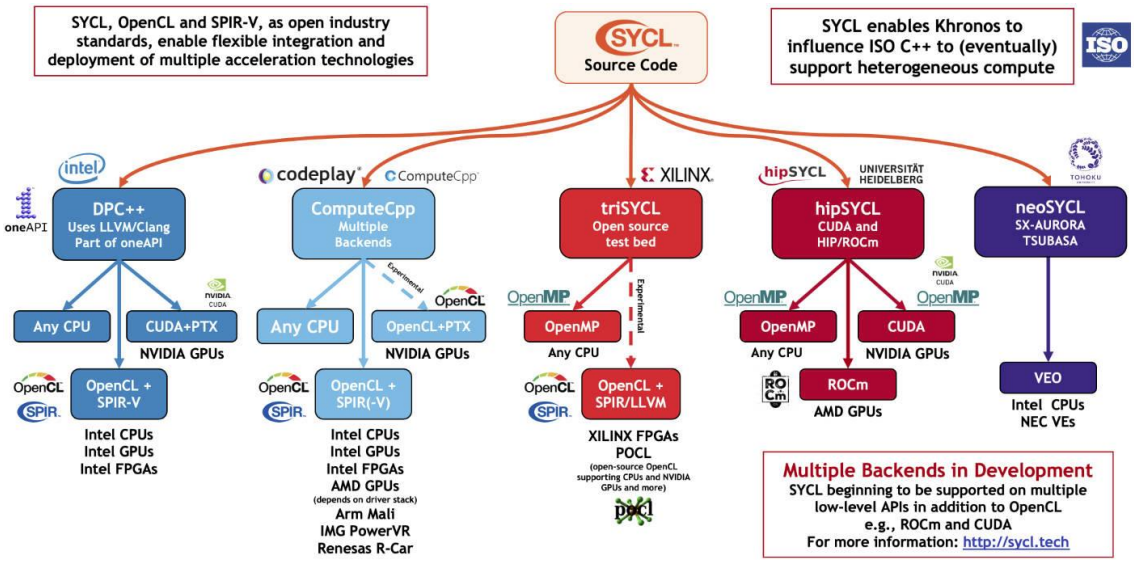

Figure 7: A selection of popular SYCL implementations. As indicated by the figure, DPC++, ComputeCPP and triSYCL all offer the possibility of running SYCL applications on FPGAs. [Source: https://www.khronos.org/assets/uploads/apis/2020-05-sycl-landing-page-02_3.jpg (page: https://www.khronos.org/sycl/)]

7. New Programming Environments

Due to the relative slowdown in the rate of advances regarding raw processing speed, i. e. processor operation frequency, the focus for chip manufacturers has shifted towards providing advanced features that allow equivalent workloads to be processed in a shorter amount of time, when their highly specialized instructions are utilized. These features cover a wide range, from advanced vector instructions that are part of the central processing unit, to the specialized standalone chips that abandon most general purpose instructions in order to excel at highly parallel operation or even hardware that can be altered in software for executing a specific code path more efficiently. As a result, programmers are presented with a plethora of hardware options that promise superior performance for specialized tasks compared to the traditional, general purpose processor. Unfortunately, access to these resources in software is not straightforward. This section will try to familiarize the reader with various approaches dealing with a wide range of hardware. For a more specific use case, focusing on accessing Field Programmable Gate Arrays via SYCL, review of the chapter “Non-conventional hardware (FPGAs)” is recommended.

The resources provided to the software programmer for achieving high performance can be divided into two categories: (a) facilities that enable access to specialized hardware, also widely known as accelerators, and (b) software packages that aim to exploit the extended compute abilities provided by the Central Processing Unit (CPU) to a fuller extent. The distinction between these categories, however, is not absolute, especially when the efforts to unify the programming models between CPUs and other accelerators are taken into consideration. As such, the facilities showcased in the two respective chapters should not be taken as alternatives, but complementary facilities, with the first section focusing on specific adaptation layers for specialized hardware, and the second section exploring various approaches on using the provided compute capabilities within the applications being developed.

The most prominent member of the accelerator hardware family is the General Purpose Graphics Processing Unit (GPGPU), which can be likened to a many-core CPU with a focus on vector processing, but reduced instruction complexity. As a result, these accelerators tend to excel in Single Instruction Multiple Data (SIMD) operations. A second member, that has long been utilized in application specific use cases, but has only been recently included in the HPC installations, is the Field Programmable Gate Array (FPGA). FPGAs, as the name suggests, can be considered as hardware (as opposed to software) designed to execute a specific application, but the design can be altered by the programmer, hence the “Field Programmable” part in the name. A similar type of hardware is the Application Specific Integrated Circuit (ASIC) that is also specific to a single application that forfeits alterable design, in order to reduce cost, and possibly increase performance. Finally, the most recent addition to the family is the Data Processing Unit (DPU) [MSV, 2021] , which, unlike the previously listed, is defined by its role, controlling data flow between compute elements, rather than its hardware make-up, which can be a complete System-on-a-Chip (SoC) featuring FPGA or ASIC components.

Similar to the GPU being the most widely deployed, and most familiar, hardware accelerator, the most widely used framework for utilizing discrete hardware accelerators is the Compute Unified Device Architecture (CUDA) by NVIDIA [NVIDIA, 2021] . However, this framework is limited to a single type of accelerator, GPGPU, produced by a single company, NVIDIA. Nevertheless, the prominence of this framework despite its specificity,

should not be ignored, as it is used as an argument for its ease of use compared to the competition. In a similar vein, NVIDIA has released DOCA [NVIDIA, 2021b] as the framework for controlling its own brand of Bluefield DPUs [NVIDIA, 2021c].

An alternative framework for programming hardware accelerators, including the GPUs (both by NVIDIA and other vendors), FPGAs and CPUs, is the Open Computing Language (OpenCL) [OPENCL, 2020] . Despite its wider range of target platforms, programming in OpenCL has been perceived as harder to use and requiring more boilerplate code, compared to its main competitor, CUDA [DEMIDOV, 2013]. As a result, recent efforts involving OpenCL have been focusing on providing frameworks that operate at a higher level of abstraction, and use OpenCL as the lower level target language.

One of the higher level frameworks for targeting OpenCL supported devices is SYCL [SYCL, 2020]. The main advantage of SYCL, from a programming language perspective, is that it uses standard ISO C++ language for both the host and kernel code, instead of relying on non-standard extensions on a language bound to a dedicated compiler. This paradigm, also often referred to as "single-source", offers a number of advantages, such as facilitating the re-use of mature and readily-available tools such as debuggers.

Another contender that employs the single source file for both host and kernel code approach, but prefers to rely on language extensions, is the OneAPI framework by Intel [INTEL 2021b] . This framework builds on the core provided by SYCL, but also provides the Data Parallel C++ (DPC++) specification as an extended language, and an abstraction layer to its hardware that is compatible with OpenCL [INTEL, 2021c]

Similarly, AMD provides its own framework for targeting hardware accelerators, Heterogeneous-Compute Interface for Portability (HIP) that is based on C++ and targets OpenCL runtime at the lower levels [HIP, 2021] . However, AMD also provides automated tools for converting existing CUDA code to HIP. The name for the backend that runs OpenCL code on AMD hardware is Radeon Open Compute (ROCm) [AMD, 2021] .

The latter three frameworks, SYCL, DPC++ and HIP, all provide access to hardware accelerators using minimal extensions to the standard programming languages. Furthermore, these frameworks are able to target traditional CPUs as well, exploiting the vector processing capabilities in particular. However, as mentioned before, additional options besides directly programming in these languages are also available. The most common approach for obtaining higher performance when developing a new application is using libraries that are optimized for the specific hardware, usually provided by the hardware vendor. For the x86_64 platform, both hardware manufacturers that are widely deployed in HPC installations provide libraries for common mathematical operations optimized for their own brand of hardware: Math Kernel Library (MKL) for Intel [INTEL, 2021d] and Optimizing CPU Libraries (AOCL) for AMD [AMD, 2021b] . These libraries have bindings for languages widely used in HPC, such as Fortran, C and C++.

Alternatively, instead of writing the entire codebase in a performance focused mindset, the application can be developed in a language that provides a higher level of abstraction at the cost of lower performance, which is alleviated by offloading the compute intensive operations to kernel functions written in the aforementioned languages and/or libraries [SYCL, 2021b] . This is the approach employed by the TensorFlow framework that allows programs to be developed in Python, an interpreted language not as efficient as the languages mentioned above (Figure 8) [TFLOW, 2021] However, the Python codebase in such applications mostly acts as a glue for binding high performance libraries that typically use a GPU as an accelerator.

Finally, there is an ongoing effort on the development of high level languages that are designed with high performance computing as the primary use case. Due to the nature of progress in the computer science and engineering field, any attempt to compile a list of such languages would result in an incomplete and biased collection. However, the Julia language can be given as an arbitrarily chosen example [JULIA, 2021] .

As mentioned in the opening paragraphs of this section, slowing advances in increasing processor operating frequency draws a picture leaning towards qualitative improvements over quantitative improvements, as in, providing new ways to do work instead of doing the same work faster. Unfortunately, qualitative improvements are harder to predict and exploit, compared to the straightforward method of running the same code on a faster processor. The greatest danger for the software at hand would be being crystallized in the hardware that is available now, and unable to exploit the performance promised by newer hardware in the future.

The main method for future proofing the software being developed is maintaining succinct and portable code that is separated from the intricacies which allow harnessing the performance of a specific piece of hardware. However, in order to allow such code, even more concentrated effort should be put into the design and implementation of the interfaces and backends (i. e., libraries/APIs (Application Programming Interfaces) and compilers, respectively), that allow execution of portable code. Finally, the open standards should be encouraged for these interfaces in order to avoid vendor lock-in and allow migration via minimal effort when better hardware resources become available.

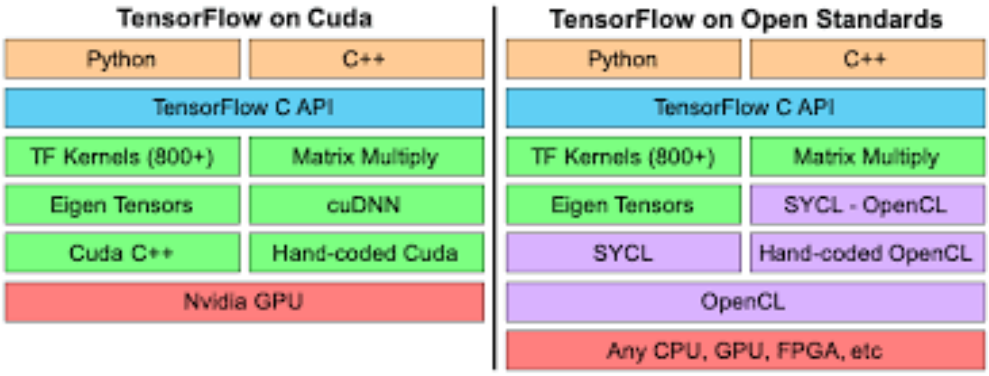

Figure 8: Porting TensorFlow to open standards. SYCL enables TensorFlow (TF) and other AI frameworks to run on any type of hardware using open software stacks.

[Source: Gwennap, L. (2019, September). *Accelerating AI Performance in an Open Software Ecosystem.* The Linley Group. https://www.linleygroup.com/uploads/codeplay-accelerating-ai-performance-wp.pdf]

8. Summary

This report includes analysis at several architecture levels of HPC ecosystems. The aim is to present user requirements influencing future technologies (software and hardware). The potential changes on HPC architecture is due to certain requirements and trends that can be observed nowadays. However, the list of requirements analysed is limited due to the capacity of the report. It is worth mentioning that the changes we will notice in future hardware and software may not be exactly the same as those outlined here. The report rather presents the current trends that will influence future changes that we may observe in the HPC and cloud markets according to the views of the authors.

We can summarise the outcomes in the following topics:

1) A significant increase in HPC capabilities due to a rapid development of technologies other than conventional computing opens for users new applications areas with again their potential influence on new development:
   - The main method for future proofing the software being developed is **maintaining succinct and portable code that is separated from the intricacies** that allow harnessing the performance of a specific piece of hardware. A more concentrated effort should be put into the design and **implementation of the interfaces and backends, i.e., libraries/APIs and compilers**, respectively, that allow its execution.
   - There is a need to use alternative technologies together under a complex environment such as GPUs, FPGAs and **hybrid systems** with quantum computing. This approach will dominate in the next few years.

2) Future programming environments and software development in HPC:
   - The market will follow various approaches for exploiting traditional CPUs and dedicated hardware: advanced compilers, low level languages with **finer control over hardware, high performance libraries and HPC oriented high level languages**.
   - **Open standards should encourage software interfaces and programming environments** themselves in order to avoid vendor lock-in, and allow migration via minimal effort, when better hardware resources become available, especially in hybrid infrastructures mentioned above.
   - The started already effort (SYCL) has to provide a **unified interface to specialised but specific hardware** like GPU, FPGA, ASIC.

3) New perspectives in computational science and possibilities to run more complex problems based on AI and Big Data analysis:
   - AI and Big Data are tightly coupled together. In most cases we need large datasets (natural or synthetically generated) for the purpose of Deep Learning where a **closer integration of highly efficient data services and HPC services will be required**.
   - Apart from common ML usage like Tensorflow and mostly image recognition methods, the AI solutions have to be planned and considered for dedicated problems with both hardware and software already in mind. A common approach we will follow to use **large HPC systems for training of AI**

based networks and **smaller systems to be used for edge computing of already trained applications**.

- The process of education including **training courses and studies at universities will influence the effectiveness of future use of AI and BD** analysis in wide spectrum of data science and data scientists.

4) Much more wider openness to the market – increases the companies interest in using HPC - from large corporate research to the use of HPC by SMEs:
   - Widespread access to HPC infrastructure (there is access not only to pre-exascale or exascale systems, but also petascale systems located in EU countries) will unlock HPC for science and industry in areas which could not afford it before, with new challenges and demanding requirements of production based SLAs **– higher robustness and reliability.**
   - The growing need for highly-efficient computing, continued diversification, and expansion of the IT industry, **advances in virtualization, and rising preference for hybrid HPC solutions** are some of the factors expected to fuel the growth of the market. The ability of HPC systems to process large volumes of data at higher speeds is prompting government agencies, defence contractors academic institutions, energy companies, and utilities to adopt HPC systems, which also bodes well for the growth of the HPC market.

5) Interactivity in HPC:
   - Nowadays, more and more simulation workflows require non-negligible required steps - in terms of time and computing resources - involving interactive analysis, processing and workflow steering. This clearly impacts how modern high performance computing systems (HPC) should be designed and which additional services they should provide. In the context of HPC and as opposed to traditional batch mode we can **define interactive computing as a service able to provide, in a timely fashion, sufficient computing and storage resources and an interaction mechanism** that allows users to interact with responsive HPC systems.
   - This definition of interactive computing contains two key aspects: i) the need to support the service through HPC class hardware, i.e. through dedicated portion of systems equipped with specific hardware tailored for visualization and data analysis (GPUs, high IOPS storage, state-of-the art CPUs); ii) **the need to provide a framework where interactive computing is made possible through adequate response time** (access in order of a minute, responsivity of the system) and suitable application environment (widely spread data analysis framework).
   - Data analysis and visualization is a field that sees continuous evolution of applications and techniques. Therefore, designing and deploying a futureproof interactive computing service for HPC would require **relying on open-source projects with a large community of developers and users**. The interactive computing service should be designed to be flexible and able to seamlessly integrate the most recently developed features of such projects. In this way, users would benefit from up-to-date data processing features, while the service implementation and maintenance over time would only focus on adapting the framework and implementing it in HPC systems.

References

[ACM-DL, 2017] ACM DL, Can FPGAs Beat GPUs in Accelerating Next-Generation Deep Neural Networks, https://dl.acm.org/doi/10.1145/3020078.3021740

[AMD, 2021] AMD, Welcome to AMD ROCm™ Platform, https://rocmdocs.amd.com/en/latest/

[AMD, 2021b] AMD Developer, AMD Optimizing CPU Libraries (AOCL), https://developer.amd.com/amd-aocl/

[BISPO, 2021] João Bispo, et al. "Best Practice Guide Modern Accelerators." PRACE 2021, https://orbilu.uni.lu/bitstream/10993/47744/1/Best-Practice-Guide-Modern-Accelerators.pdf

[BlueWave, 2021] https://www.blueweaveconsulting.com/report/global-high-performance-computing-(hpc)-market

[BUSTIO, 2021] Bustio-Martínez, Lázaro, et al. "FPGA/GPU-based Acceleration for Frequent Itemsets Mining: A Comprehensive Review." ACM Computing Surveys (CSUR) 54.9 (2021): 1-35.

[DEMIDOV, 2013] Demidov, D., Ahnert, K., Rupp, K., & Gottschling, P. (2013). Programming CUDA and OpenCL: A case study using modern C++ libraries. SIAM Journal on Scientific Computing, 35(5), C453-C472

[FENIX,2021] FENIX, Delivering e-infrastructure services federated as the Fenix Infrastructure, https://www.fenix-ri.eu

[FENIX, 2021b] Implementation of the Fenix Interactive Computing Service. Fenix-ICEI internal document

[FZJ, 2021] https://openondemand.org/, https://github.com/FZJ-JSC

[HIP, 2021] HIP Programming Guide, https://rocmdocs.amd.com/en/latest/Programming_Guides/HIP-GUIDE.html

[HPC, 2021] High Performance Computing Market Size, Share & Trends Analysis Report By Component (Servers, Services), By Deployment (On-premise, Cloud), By End-use, By Region, And Segment Forecasts, 2020 – 2027, Published Date: Sep, 2020, https://www.grandviewresearch.com/industry-analysis/high-performance-computing-market

[INDUSTRY4.0, 2020] Why is Big Data the core of the 4.0 industry, https://nexusintegra.io/big-data-industry-4-0/

[INSIDE,2019] Inside HPC, FPGAs and the Road to Reprogrammable HPC, https://insidehpc.com/2019/07/fpgas-and-the-road-to-reprogrammable-hpc/

[INTEL, 2021] Intel® FPGA Add-on for oneAPI Base Toolkit, Accelerate Your Data-Centric Workloads with FPGAs , https://software.intel.com/content/www/us/en/develop/tools/oneapi/components/fpga.html

[INTEL 2021b] INTEL Products, Driving a New Era of Accelerated Computing, https://software.intel.com/content/www/us/en/develop/tools/oneapi.html

[INTEL, 2021c] INTEL Products, https://software.intel.com/content/www/us/en/develop/tools/oneapi/data-parallel-c-plus-plus.html

[INTEL, 2021d] Intel Products, Intel®-Optimized Math Library for Numerical Computing, https://software.intel.com/content/www/us/en/develop/tools/oneapi/components/onemkl.html

[JULIA, 2021] The Julia Programming Language, https://julialang.org/

[JUPYTER, 2021] JupyterHub, A multi-user version of the notebook designed for companies, classrooms and research labs, https://jupyter.org/hub

[KAMAL, 2021] Kamalavasan Kamalakkannan, Gihan R. Mudalige, Istvan Z. Reguly, Suhaib A. Fahmy, High-Level FPGA Accelerator Design for Structured-Mesh-Based Explicit Numerical Solvers, https://arxiv.org/abs/2101.01177

[KERYELL, 2018] Ronan Keryell & Lin-Ya YuXilinx Research Labs, Experimenting with SYCLsingle-source post-modern C++ on Xilinx FPGA, https://www.iwocl.org/wp-content/uploads/DHPCC_triSYCL.pdf

[KOBAYAS, 202] Kobayashi, Ryohei, et al. "Accelerating radiative transfer simulation with gpu-fpga cooperative computation." 2020 IEEE 31st International Conference on Application-specific Systems, Architectures and Processors (ASAP). IEEE, 2020

[MSV, 2021] Janakiram MSV, What Is A Data Processing Unit (DPU) And Why Is NVIDIA Betting On It?, https://www.forbes.com/sites/janakirammsv/2020/10/11/what-is-a-data-processing-unit-dpu-and-why-is-nvidia-betting-on-it/

[NVIDIA, 2021] NVIDIA Developer, CUDA Zone, https://developer.nvidia.com/cuda-zone

[NVIDIA, 2021b] NVIDIA Developer, NVIDIA Developer, NVIDIA DOCA Software, Data Center Infrastructure-on-a-Chip Architecture, https://developer.nvidia.com/networking/doca

[NVIDIA, 2021c] NVIDIA, Transform the Data Center with NVIDIA DPUs, https://www.nvidia.com/en-us/networking/products/data-processing-unit/

[OPENCL, 2020] OpenCL, OPEN STANDARD FOR PARALLEL PROGRAMMING OF HETEROGENEOUS SYSTEMS https://www.khronos.org/opencl/

[PU, 2021] A Whole New Dimension of Processing Power": Paderborn University Commissions Atos to Build a New Supercomputer, https://www.uni-paderborn.de/en/news-item/95952

[PWC,2019] PWC, 2019 AI Predictions https://www.pwc.com/us/en/services/consulting/library/artificial-intelligence-predictions-2019.html

[SC, 2020] Artificial Intelligence is a Supercomputing problem https://towardsdatascience.com/artificial-intelligence-is-a-supercomputing-problem-4b0edbc2888d

[SC, 2021] https://sc21.supercomputing.org/

[Stone, 2018] Interactive HPC Requirements, Challenges, and Solutions for Cutting Edge Molecular Simulation Science Campaigns. John, E Stone. s.l. : First Workshop on Interactive High-Performance Computing, ISC 2018

[SYCL, 2020] Khronos Group, SYCL 2020, https://www.khronos.org/sycl/

[SYCL, 2021b] SYCL Projects, https://sycl.tech/projects/#frameworks

[TOP500, 2021] The TOP500 List, 2021 Release, https://www.top500.org/news/intel-ships-xeon-skylake-processor-with-integrated-fpga/

[TOP500, 2021b] TOP500, June 2021 edition, https://www.top500.org/lists/top500/2021/06/

[TFLOW, 2021] TENSOR FLOW, TensorFlow is an end-to-end open source platform for machine learning, https://www.tensorflow.org/overview

[QUANTUM, 2021] Mikael P. Johansson, Ezhilmathi Krishnasamy, Norbert Meyer, Christelle Piechurski, Quantum Computing – A European Perspective, PRACE-6IP TR, 2021

List of acronyms

ACID	Atomicity, Consistency, Isolation, Durability
AI	Artificial Intelligence
AOCL	Optimizing CPU Libraries
ASIC	Application-specific Integrated Circuit
BLAS	Basic linear algebra subprograms
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture
DAOS	Distributed Asynchronous Object Storage
DL	Deep Learning
DNN	Deep neural network
DPC++	Data Parallel C++
DPU	Data Processing Unit
EU	European Union
FPGA	Field Programmable Gate Array
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HBM	High Bandwidth Memory
HIP	Heterogeneous-Computing Interface for Portability
HPC	High Performance Computing
HPCC	High Performance Computing Cluster
IC	Interactive Computing
ICEI	Interactive Computing E-Infrastructure for the Human Brain Project
IOPS	I/O operations per second
IoT	Internet of Things
IPU	Intelligence Processing Unit
MKL	Math Kernel Library
ML	Machine Learning
NVMe	Non-Volatile Memory Express
OpenCL	Open Computing Language
PRACE	Partnership for Advanced Computing in Europe
RAPIDS	Suite of open-source libraries on top of CUDA
ROCm	Radeon Open Compute
SIMD	Single Instruction Multiply Data
SME	Small and Medium Size Enterprises
SoC	System-on-a-Chip
UPI	Ultra Path Interconnect
EuroHPC	European HPC / European HPC Joint Undertaking
VHDL	Very High Speed Integrated Hardware Description Language
VNNI	Vector Neural Network Instructions
VRLA	Valve Regulated Lead Acid

Acknowledgements

This work was financially supported by the PRACE project funded in part by the EU's Horizon 2020 Research and Innovation programme (2014-2020) under grant agreement 823767.