High-Performance Computing:

Why & How

An introductory guide for getting started with HPC

prepared by CaSToRC, the National Competence in HPC, as part of the EuroCC2 project

EURO²

THE CYPRUS INSTITUTE
RESEARCH•TECHNOLOGY•INNOVATION

EURO²

High-Performance Computing: Why & How

An introductory guide for getting started with HPC

Simone Bacchio[1], s.bacchio@cyi.ac.cy – on behalf of NCC Cyprus.

[1]Computation-based Science and Technology Research Center (CaSToRC), The Cyprus Institute, Nicosia, Cyprus.

High-Performance Computing (HPC) involves the use of powerful computers and clusters to address complex problems that go beyond the capabilities of standard desktops or laptops. It enables researchers and industry professionals to process massive datasets, conduct large-scale simulations, and accelerate computations in fields like physics, climate modeling, artificial intelligence (AI), and engineering.

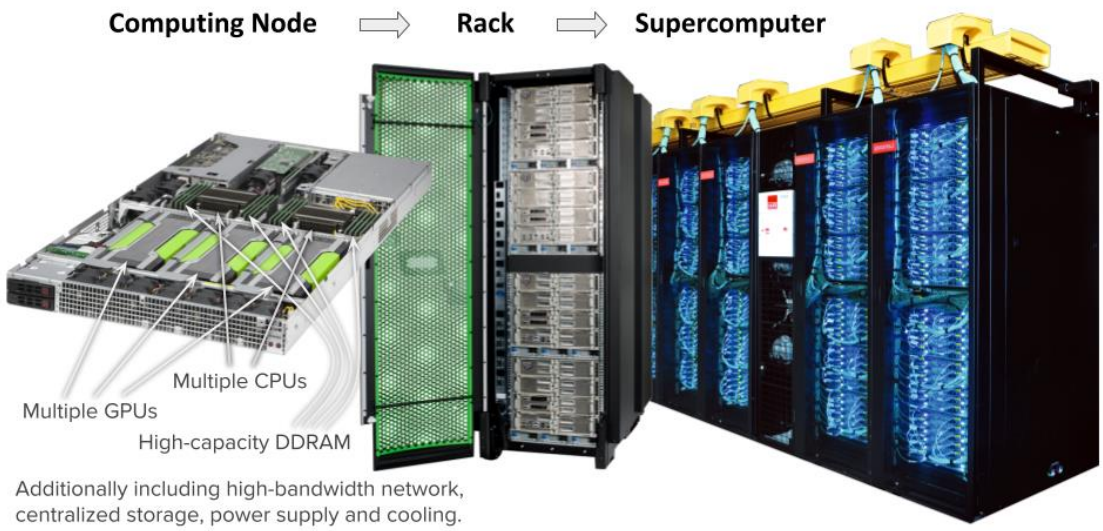
The **EuroHPC Joint Undertaking** **(JU)** is a major European initiative advancing HPC capabilities across Europe. It has funded the acquisition of multiple state-of-the-art supercomputers, including six petascale systems, three pre-exascale systems, and two exascale systems around Europe, each providing unprecedented computational power. To support the development of HPC expertise locally, EuroHPC JU has also established National Competence Centers (NCCs) in HPC across European countries, aimed at building local HPC skills and supporting academic and industry needs. Through competitive calls, EuroHPC JU offers free access to these supercomputing resources, enabling European researchers and industry users to complete computations that would take months or even years on standard systems in just hours or days. This is especially beneficial for tasks involving intensive simulations, AI-driven applications, and large-scale data processing. HPC not only enhances the scope and quality of research but also drives innovation and enables solutions to challenges that conventional computing cannot address.

This document serves as an introductory guide to help beginners understand and get started with HPC technologies. It covers:

BASIC CONCEPTS AND TERMINOLOGY

Understanding the core terminology used in High-Performance Computing (HPC) is essential for navigating and effectively utilizing these systems. Here are some key concepts used in this guide:

- **Computing Node:** A computing node is an individual computer within a supercomputing cluster. Each node contains one or more CPUs (central processing units) and often GPUs (graphics processing units), which perform the actual computations. HPC systems consist of many nodes working together in parallel to handle large-scale tasks. Other key components such as hard drives and filesystems are instead centralized such that all computing nodes can access the same data.



- **CPU and GPU:** The central processing unit (CPU) is the main processor responsible for general-purpose computing tasks, while the graphic processing unit (GPU) is specialized for parallel processing, making it ideal for tasks like machine learning and simulations that involve large datasets and repetitive calculations.
- **Performance (FLOPs):** HPC system performance is typically measured in **FLOPs** (Floating-Point Operations per Second), which represents the number of mathematical operations a computer can perform in one second. Modern HPC systems can achieve petaflops (10^{15} FLOPs) or even exaflops (10^{18} FLOPs) of performance. In comparison, a high-tier laptop operates at around one teraflop (10^{12} FLOPs), meaning that HPC systems perform between a thousand to a million times faster, enabling large-scale simulations and data-intensive tasks that are beyond the reach of conventional computers.

- **Parallel Computing**: This is the technique of dividing tasks into smaller subtasks that can be processed simultaneously across multiple CPUs or GPUs. It enables faster problem-solving by distributing workloads across several computing nodes.
- **Scalability**: This refers to the ability of a problem or code to maintain efficiency as the computational resources (nodes, cores) increase. Good scalability means that performance improves proportionally as more resources are added.
- **Cloud Computing vs High-Performance Computing**: Cloud computing provides flexible, scalable resources on demand, typically for smaller-scale projects or those requiring quick setup and teardown. In contrast, HPC systems are designed for sustained, large-scale computations, often involving multiple nodes working in parallel to achieve peak performance for tasks like simulations, AI, and data-intensive research. Cloud is ideal for early-stage work, while HPC is necessary for high-performance, large-scale computing.

MINIMAL REQUIREMENTS FOR USING HPC RESOURCES

Performance: Before considering HPC resources, it is essential to ensure that the software is optimized for basic systems such as personal computers or cloud services like Google Colaboratory. Often, slow performance stems from inefficiencies in the code—such as unnecessary loops, suboptimal I/O operations, or underutilized CPU cores—rather than a need for HPC. Optimizing code to fully exploit CPU resources using techniques like vectorization or multithreading should be the first step. This is particularly important when using Python, which is not inherently designed for high-performance computing. Instead of relying on explicit for-loops to process data, which can be inefficient, it's recommended to use high-performance Python libraries such as NumPy and CuPy (for Numpy on GPUs), which provide highly efficient array operations, or Pandas for data manipulation. For more advanced machine learning or deep learning tasks, PyTorch is a powerful framework that supports both CPUs and GPUs with highly optimized operations. Finally, understanding the specific computational needs of an application is also crucial for determining whether HPC resources are necessary and for justifying their use.

Scalability and Efficiency: HPC systems are designed to run applications across multiple computing nodes simultaneously, not just a single one. They feature centralized filesystems, high-bandwidth, high-connectivity networks between nodes, and scheduler systems to allocate supercomputer resources for a set period. *Scalability* is a critical factor when considering a move to HPC. It refers to a problem's ability to grow in size—whether by increasing data volume or enlarging simulations—while still maintaining good performance. For HPC to be effective, the code must scale efficiently, meaning it can utilize a large number of computing nodes in parallel without significant performance loss. This requires distributing computations across multiple cores or machines, ensuring efficient resource usage within the HPC environment. Efficiently scaling an application requires implementing communication protocols such as MPI (Message Passing Interface) to coordinate tasks and move data

across nodes. The goal is to achieve a similar level of performance across multiple nodes as would be achieved on a single node, ensuring that the application fully benefits from the HPC infrastructure.

Parallel Computing. To achieve scalability, an HPC application employs parallel processing, where tasks can be divided and executed simultaneously across multiple computing cores. There are three types of parallelism: *data-*, *task-parallelism*, and *SIMD vectorization*.

- **Data parallelism** involves distributing the same computation across different data chunks, making it ideal for operations on large datasets (e.g., image processing or matrix multiplication). This approach is simpler but may be less effective if the data is not easily divisible, scalable, and localized, requiring minimal information from other nodes.
- **Task parallelism** divides different tasks across processors, which works well for applications with distinct, independent computations (e.g., multi-stage simulations). While powerful, task parallelism can be more complex to implement and requires careful management of dependencies and communication between tasks.
- **SIMD Vectorization** (Single Instruction, Multiple Data) is particularly critical for GPU computing. It allows the same operation to be applied simultaneously across multiple data points within arrays, optimizing the use of GPU cores. GPUs rely heavily on this type of parallelism to efficiently process large amounts of data at once. SIMD vectorization is especially useful for scientific computing, AI, and machine learning tasks, where performing the same operation on large datasets is common.

Each type of parallelism has its strengths and they can be combined to maximize the performance of an HPC application.

HPC Software. Thus, to fully leverage HPC resources, the software should support a variety of parallel computing approaches, such as *multithreading*, *distributed computing,* or *GPU programming*. The software should be optimized not only for general parallelism but also for the specific hardware architecture of the HPC system, such as Intel, AMD, or ARM CPUs, and NVIDIA, AMD, or Intel GPUs. Each hardware model is unique, and not all software supports every type of hardware uniformly. If third-party software is used, such as PyTorch for AI, it is essential to investigate how these features are supported. For example, while PyTorch supports most GPUs, installation and execution methods vary by hardware, as well as distributed computing is implemented in separate packages. If instead the software is proprietary and internally developed, these features need dedicated implementations or leverage HPC libraries and standards such as BLAS, ensuring that the code efficiently utilizes both CPU and GPU architectures. This often requires expertise in low-level programming and experienced developers in HPC.

Innovative Research and Development (R&D). Finally, accessing publicly funded resources like those provided by EuroHPC is an excellent opportunity for both non-

profit and for-profit entities engaged in high-impact, innovative projects. While applications generally require a research-driven proposal with clear computational objectives, private companies and industry partners are welcome to apply, as long as their projects align with EuroHPC's goals of advancing science, innovation, and technological capabilities. For-profit projects may be subject to specific conditions, such as access fees or co-financing requirements, depending on the call. Applicants are encouraged to demonstrate how EuroHPC resources will enable advanced data processing, simulations, or computational tasks that standard systems cannot handle, ensuring that the scale and power of HPC are leveraged to drive impactful advancements across academia and industry. Additionally, smaller resource allocations can sometimes be requested for development or benchmarking purposes, which can serve as a stepping stone for refining code and preparing larger projects. Access is in general granted competitively and innovative research that pushes the boundaries of science or technology typically stands a better chance of securing access to these highly sought-after resources.

INITIAL STEPS FOR BEGINNERS

1. **Get Experience with Cloud Computing:** Starting with cloud computing platforms like Google Colaboratory offers an accessible way to gain experience with small-scale experiments and GPU-enabled applications. These platforms provide a simple entry point into remote computing, allowing beginners to test their code and get accustomed to managing computational tasks in a cloud environment. This serves as a valuable foundation before moving on to more advanced HPC systems. However, cloud computing has limitations in terms of performance, scalability, and resource availability. Cloud platforms often impose restrictions on processing time, memory, and the number of concurrent tasks, making them less suitable for large-scale simulations or data-intensive applications. When projects outgrow these limits—such as requiring more sustained computing power or efficient parallel processing—transitioning to HPC becomes necessary to handle larger datasets, distributed tasks across multiple nodes, and high-performance workloads efficiently.

2. **Apply for Small Resource Allocations:** After gaining initial experience, consider applying for small allocations of resources on national supercomputers, such as Cyclone at The Cyprus Institute, or EuroHPC systems through benchmark and development access. This enables software testing, development, and optimization with a minimal resource commitment before scaling up to larger projects. While HPC resources are accessed remotely through a terminal and require familiarity with a Linux environment, command-line tools, and bash scripting, it is possible to begin by using interactive tools like Jupyter Notebooks to execute Python scripts. This approach provides access to superior GPUs and in-node performance, compared to cloud computing. However, it is essential to gradually build expertise in compiling applications, scheduling jobs, and minimizing runtime user input. Learning to run applications efficiently and

retrieving data, as well as collecting timing and performance benchmarks is crucial, as larger HPC allocations typically require demonstrated experience and benchmark results on the specific architecture being applied for.

3. **Participate in Training:** Training programs offered by supercomputing centers and national competence centers in HPC provide essential skill-building opportunities. These sessions cover topics such as parallel programming, software optimization, and other key competencies needed to effectively use HPC resources.

4. **Contact the National Competence Center:** National competence centers in HPC are valuable resources for support and guidance. They offer assistance with technical issues, provide advice on accessing HPC resources, and can help navigate the application process, making the transition to HPC smoother for beginners.

NATIONAL COMPETENCE CENTER (NCC) IN HPC OF CYPRUS (https://eurocc.cyi.ac.cy)

The EuroHPC JU funds National Competence Centers (NCCs) across Europe to support HPC access and development, through the EuroCC2 project. In Cyprus, The Cyprus Institute's Computation-based Science and Technology Research Center (CaSToRC) serves as the NCC, providing crucial support for the Cypriot academic, industry, and government HPC community. Here's how the NCC can help you:

- **Regular Training in HPC and AI:** The NCC offers training programs at various levels, from beginner to advanced, targeting different communities, including academia, industry, and government.
- **Hackathons and Community Engagement:** To foster collaboration and engagement, the NCC organizes hackathons, workshops, and seminars, allowing participants to work on real-world problems and interact with experts in the field.
- **Dedicated Support:** The NCC is available for meetings and discussions tailored to your specific needs, providing personalized guidance on HPC-related queries.
- **Assessing Computational Needs:** The NCC can help evaluate the computational demands of your application, offering advice on how to scale up to HPC resources and optimize performance.
- **HPC Application Assistance:** Whether you're applying for national or EuroHPC resources, the NCC can assist you in preparing your application and guide you through the process of running on HPC systems for the first time.

Note: While the NCC offers valuable support, it is not responsible for actively developing software or performing specific computational tasks. Its role is to guide and advise, empowering you to succeed independently.

The full set of services offered by NCC Cyprus is available on the national website.

AVAILABLE HPC RESOURCES FOR THE CYPRIOT COMMUNITY

National Resources

- **Cyclone at the Cyprus Institute**: Cyclone is a high-performance computing system hosted at The Cyprus Institute, providing the national research community with substantial computational resources for scientific simulations and large-scale data processing. It supports a range of disciplines, including physics, climate modeling, bioinformatics, and more. Access to Cyclone also includes support from the National Competence Center for HPC and technical assistance from the High-Performance Computing Facility (HPCF), helping users make the most of the available resources.

EuroHPC Resources

EuroHPC resources are the main focus of this guide, offering vast computational power through competitive access for large-scale simulations, data analysis, and AI applications. These machines are classified into three categories, each offering different levels of performance:

1. **Petascale Systems:** Petascale machines can perform at least one petaflop (10^{15} floating-point operations per second). These systems are ideal for medium-to-large research projects requiring significant computational resources but not the extreme capabilities of larger systems.
   a. **Discoverer (Bulgaria):** Boasts over 1100 CPU-only nodes, including Intel Xeon processors, and a peak performance of 4.5 petaflops.
   b. **Vega (Slovenia):** Features 960 CPU nodes, powered by AMD EPYC processors, and 60 GPU nodes, powered by NVIDIA A100 GPUs, delivering a combined performance of over 6.9 petaflops.
   c. **Deucalion (Portugal):** Deucalion includes over 550 nodes with ARM architecture, supporting a computing power of 10 petaflops.
   d. **Karolina (Czech Republic):** Comprising 720 nodes with Intel Xeon processors and 72 nodes with NVIDIA A100 GPUs, Karolina offers a peak performance of 15.7 petaflops.
   e. **Meluxina (Luxembourg):** Equipped with 573 CPU-only nodes and 200 GPU-accelerated nodes, Meluxina delivers a total computing power of 18 petaflops.
   f. **Daedalus (Greece, Upcoming):** Details to be confirmed but expected to include a mix of CPUs and GPUs, offering around 7 petaflops of peak performance.

g. **Arrhenius (Sweden, Upcoming):** Details to be confirmed but expected to offer around 40 petaflops of peak performance.

2. **Pre-Exascale Systems:** Pre-exascale machines are capable of performing up to hundreds of petaflops and are designed to tackle more complex and large-scale simulations.
   a. **LUMI (Finland):** One of Europe's largest supercomputers, LUMI has over 2,000 AMD EPYC CPU nodes and about 3,000 GPU nodes, with a peak performance of 550 petaflops.
   b. **Leonardo (Italy):** Comprises about 1,500 CPU nodes and 3,500 GPU nodes using NVIDIA A100 and Intel Xeon processors, delivering more than 240 petaflops.
   c. **MareNostrum 5 (Spain):** A heterogeneous system combining 3,500 nodes across ARM, x86, and GPU architectures, with a peak performance over 300 petaflops.
3. **Exascale Systems:** Exascale systems represent the cutting edge of computational power, capable of executing over one exaflop (10^{18} operations per second). These machines are designed for the most computationally demanding problems, such as large-scale climate simulations, advanced AI, and complex physical models.
   a. **JUPITER (Germany):** Expected to have over 6,000 GPU nodes with a combined exaflop performance, JUPITER will be Europe's first exascale machine.
   b. **Alice Recoque (France, Upcoming):** Detailed specifications are pending, but it is expected to exceed one exaflop with tens of thousands of nodes using a mix of CPU and GPU architectures.

These EuroHPC systems are accessible to both academic researchers and industry professionals through competitive application processes, providing the infrastructure necessary for cutting-edge research and development.

TYPES OF ACCESS MODES FOR REQUESTING RESOURCES

Understanding the various access modes to HPC resources is essential for maximizing your computing capabilities. Here's a breakdown of the access options available for both national and EuroHPC resources:

1. National Access: Cyclone at the Cyprus Institute (https://hpcf.cyi.ac.cy/apply)

To gain access to the Cyclone supercomputer, you need to submit a request that includes details about your research project and its computational needs. Access is primarily available to researchers within Cyprus, including academic institutions and industry partners. It typically involves the following steps:

- Submit a project proposal outlining your objectives, methodology, and the computational resources required.
- Your proposal will be reviewed for relevance, feasibility, and potential impact on the Cypriot research community.

- Successful applicants will receive access credentials and guidance on how to use the system effectively.

2. Benchmark and Development Access (https://access.eurohpc-ju.europa.eu/)

This access mode offers a fixed, small amount of resources on any HPC system, suitable for initial testing and development work.

- **Minimal Requirements for Proposals:** Proposals must outline the intended use of resources and demonstrate the need for benchmarking or development purposes.
- **Deadlines:** Applications are accepted on the 1st day of each month.

3. AI and Data-Intensive Applications Access (https://access.eurohpc-ju.europa.eu/)

This mode provides a fixed, medium amount of resources (approximately 10 times the benchmark access) specifically on GPU systems for projects focused on AI and data-intensive applications.

- **Proposal Length:** 5 pages, detailing the objectives, methodology, and expected outcomes of the project.
- **Deadlines:** Applications are due twice per year around April and June.

4. Regular Access (https://access.eurohpc-ju.europa.eu/)

This access mode allows users to request a medium-to-large amount of resources on any HPC system for broader research projects.

- **Proposal Length:** 12 pages, outlining the research background, objectives, methods, and anticipated results.
- **Deadlines:** Applications are due twice per year around March and September.

5. Extreme Scale Access (https://access.eurohpc-ju.europa.eu/)

For highly ambitious projects requiring large amounts of resources on pre-exascale systems, this access mode is designed for individual or collaborative research initiatives.

- **Proposal Length:** 12 pages, providing comprehensive details on the project scope, computational needs, and potential impact.
- **Deadlines:** Applications are due twice per year around April and October.
- **Access Types:** Includes individual user access for specific projects and collaborative access for interdisciplinary or multi-institutional initiatives.

Each access mode offers different tiers of support, allowing researchers and industry professionals to choose the level of resources that best fits their needs, whether through local resources, cloud computing, or EuroHPC systems.

USEFUL RESOURCES, LINKS AND CONTACTS

To facilitate your access to HPC resources and support your research endeavors, here are some valuable resources and links:

1. **EuroHPC Portal (https://eurohpc-ju.europa.eu/):** The central information on EuroHPC resources, funding opportunities, and access guidelines. Visit the portal to explore available systems and application procedures.
2. **Cyclone at the Cyprus Institute (https://hpcf.cyi.ac.cy):** Learn more about the Cyclone supercomputer, its specifications, and how to apply for access. The Cyprus Institute provides detailed information on its HPC capabilities.
3. **Top500 (https://www.top500.org):** A list of the top 500 supercomputers in the world, providing insights into the latest advancements in HPC technology and performance rankings.
4. **Google Colaboratory (https://colab.google/):** A cloud-based platform ideal for beginners to run small-scale experiments and gain experience with GPU-enabled applications in a user-friendly environment.
5. **HPC Training Materials and Courses:** Access a range of training materials, documentation, and tutorials designed to help users understand HPC systems, programming, and best practices.
   a. PRACE Training: https://events.prace-ri.eu/category/2
   b. NVIDIA Courses: https://www.nvidia.com/en-us/training/online
   c. NCSA Training: https://www.hpc-training.org/moodle
   d. Other courses: https://tinyurl.com/hpc-online-resources
6. **HPC Software:** Explore various HPC software options that support parallel computing, GPU utilization, and efficient data processing. Check compatibility with specific hardware architectures.
   a. Apache Spark - A powerful open-source framework for large-scale data processing and analytics. It supports in-memory computing, distributed processing, and integration with big data tools, making it a popular choice for HPC tasks.
   b. CUDA - A high-performance parallel computing platform and API model designed by NVIDIA for GPU acceleration. It enables developers to harness the power of NVIDIA GPUs for general-purpose processing.
   c. CuPy - A library that provides a NumPy-like API for GPU arrays, enabling fast computation on NVIDIA GPUs by utilizing CUDA. It is highly compatible with Python libraries for scientific computing.
   d. HIP - A C++ runtime API and kernel language developed by AMD for cross-platform GPU programming, compatible with AMD and NVIDIA GPUs.
   e. Horovod - A distributed deep learning training framework that works with TensorFlow, Keras, PyTorch, and Apache MXNet, making it easier to scale training across multiple nodes.
   f. JAX – High-performance library for numerical computing with GPU/TPU support, enabling automatic differentiation and parallelization.

g. MPI - A standardized protocol for inter-process communication used in parallel computing environments. The OpenMPI implementation is widely used in HPC applications.
h. mpi4jax - A library for zero-copy MPI communication for JAX arrays, enhancing distributed computing capabilities in JAX.
i. mpi4py - Provides Python bindings for MPI, enabling parallel computing capabilities in Python applications for distributed tasks.
j. NumPy - Core library for numerical computing in Python, supporting multi-dimen- sional arrays and mathematical operations. Often used with CuPy or Numba for acceleration.
k. Numba - A just-in-time (JIT) compiler that translates a subset of Python and NumPy code into fast machine code, significantly speeding up Python applications.
l. OpenMP - A multi-platform library supporting shared-memory parallel programming in C, C++, and Fortran. It simplifies parallel processing on multi-core CPUs.
m. PyTorch - A deep learning framework with GPU support that is ideal for scalable machine learning and data science tasks. Supports distributed training with Horovod.
n. Ray - A framework for scaling Python and AI workloads, from reinforcement learning to deep learning, across multiple nodes or clusters.
o. ROCm - The first open-source software development platform for high-performance computing on AMD GPUs, providing a powerful framework for GPU computing in HPC environments.
p. TensorFlow Distributed - TensorFlow's built-in support for distributed training, allows for large-scale deep learning model training on clusters of CPUs or GPUs

For any support on HPC and related technologies, contact NCC Cyprus.

https://eurocc.cyi.ac.cy/ eurocc-contact@cyi.ac.cy

ACKNOWLEDGMENTS

Funded by the European Union. This work has received funding from the European High-Performance Computing Joint Undertaking (JU) and Germany, Bulgaria, Austria, Croatia, **Cyprus (co-funded by the EU within the framework of the Cohesion Policy Programme "THALIA 2021-2027")**, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Türkiye, Republic of North Macedonia, Iceland, Montenegro, Serbia under grant agreement No 101101903.