

# < QC | HPC > Quantum for HPC

The impact of Quantum Computers on HPC applications  
and the integration of quantum computers in HPC centres

*White Paper*

Dr. Valeria Bartsch, Dr-Ing Guillaume Colin de Verdière,  
Dr-Ing Jean-Philippe Nominé, Dr. Daniele Ottaviani, Dr. Daniele Dragoni,  
Dr. Chayma Bouazza, Fabrizio Magugliani, David Bowden,  
Dr. Cyril Allouche, Dr. Mikael Johansson, Dr. Olivier Terzo,  
Dr. Andrea Scarabosio, Giacomo Vitali, Dr. Farida Shagieva,  
Prof. Dr. Kristel Michielsen

05/10/2021

[etp4hpc.eu](http://etp4hpc.eu)

[@etp4hpc](https://twitter.com/etp4hpc)



EUROPEAN  
TECHNOLOGY  
PLATFORM  
FOR HIGH  
PERFORMANCE  
COMPUTING

## Introduction

Quantum Computing (QC) describes a new way of computing based on the principles of quantum mechanics. From a High Performance Computing (HPC) perspective, QC needs to be integrated:

- at a system level, where quantum computer technologies need to be integrated in HPC clusters;
- at a programming level, where the new disruptive ways of programming devices call for a full hardware-software stack to be built;
- at an application level, where QC is bound to lead to disruptive changes in the complexity of some applications so that compute-intensive or intractable problems in the HPC domain might become tractable in the future.

The White Paper QC for HPC focuses on the technology integration of QC in HPC clusters, gives an overview of the full hardware-software stack and QC emulators, and highlights promising customised QC algorithms for near-term quantum computers and its impact on HPC applications. In addition to universal quantum computers, we will describe non-universal QC where appropriate. Recent research references will be used to cover the basic concepts. The target audience of this paper is the European HPC community: members of HPC centres, HPC algorithm developers, scientists interested in the co-design for quantum hardware, benchmarking, etc.

## Target audience

The white paper about Quantum Computing (QC) for High Performance Computing (HPC) is written in the context of the European Technology Platform ETP4HPC. Therefore the main focus will be on the integration of QC as a disruptive new technology in a wider HPC context. The target audience is the HPC community in Europe including:

- HPC centres interested in Quantum Computing in order to prepare its use in HPC facilities.
- HPC algorithm developers who would like to be informed if, how and when they need to familiarise themselves with quantum computing algorithms.
- Scientists interested in co-design for novel quantum hardware and the benchmarking of upcoming quantum systems.

The white paper also has the intention to set the scene for the upcoming strategic research agenda SRA5 of ETP4HPC and introduces some of its important topics in more detail.

## Table of Contents

Introduction .....	2
Target audience .....	2
Key insights .....	3
Key recommendations .....	3
Clarification of terms .....	4
QC hardware technologies .....	5
Algorithms and Applications: .....	6
Benchmarking .....	8
Software-Hardware stack .....	9
System integration .....	10
Emulation of Quantum Computers with HPC devices .....	12
Conclusions .....	13
References .....	14
Funded projects .....	15



## Key insights

- Currently, we are witnessing the emergence of quantum-enabled solutions while no quantum advantage for an industrially relevant application has yet been achieved.
- The race to QC hardware is still open and we will see multiple QC technologies in use, at least for an intermediate stage of development.
- Three application areas are of special interest: optimisation, quantum chemistry and quantum machine learning.
- Benchmarks (application-oriented) are needed to assess the properties and capabilities of different technologies .
- Many quantum algorithms, such as variational algorithms, are hybrid and divide the problem into a classical and a quantum part, therefore an effective integration of quantum and classical computers is important.
- There are several options for the system integration of quantum computers which may coexist in the medium term, until applications with quantum advantage and workflows between classical and quantum computers have been developed.
- New challenges concerning the hardware-software stack for the deployment and emulation of quantum computers will arise for larger quantum systems.



## Key recommendations

The recommendations are the building blocks for a quantum software portfolio that supports a European ecosystem for QC and aims at European sovereignty in this technology domain.

In today's era of multiple QC technologies, it is important not to lock into one too soon. This point is addressed in a series of recommendations:

- Promote the establishment of a well-defined vendor-independent European benchmark suite, covering a diverse set of quantum algorithms and real-world use cases. This is important for enabling a fair evaluation of the portfolio of quantum computers hosted in Europe.
- Promote the establishment of a European HPC-QC ecosystem to support open standards (interfaces/APIs). Especially here, a common standardised entry-point to QC implementations, due to its agnosticism towards the various competing quantum technologies, could be an important element of a European QC platform.
- To strengthen investment plans in quantum technologies, to reduce the risk of technology change (i.e. short technology update cycles), it might be advantageous to deploy QC initially in public institutions, e.g. HPC and research centres, and to involve industrial partners at an early stage, as it is currently the practice in HPC centres.

In addition, another important point in the integration of QC and HPC needs to be addressed: optimising the integration of QC in HPC. Variational algorithms with a high number of iterations between QC and HPC are likely to dominate on approximate quantum computers in the short term. Thus, the performance of quantum computing devices (computational power and different latencies) is of utmost importance. This can be (partially) optimised with an efficient runtime system, depending on the quantum technologies used.

In general, there is a need to build a workforce of designers, developers and users who are specifically familiar with quantum computing technologies and the integration of these technologies in HPC workflows. This can be facilitated by training curricula, courses and classes in universities and HPC centres, and projects and workshops aimed at connecting the HPC and QC communities.

## Clarification of terms

**Noisy Intermediate-Scale Quantum (NISQ) Era:** quantum computers with 50-100 qubits may be able to perform tasks that surpass the capabilities of today's classical digital computers, but noise in quantum gates will limit the size of quantum circuits that can be reliably executed (Preskill 2018).

**Universal Quantum Computer:** a fault-tolerant quantum computer that can (1) execute all classical algorithms and (2) in addition, execute algorithms that are beyond the capacity of any classical computer, by exploiting quantum parallelism and entanglement (Deutsch 1985). Universal QC approaches are relying on the concept of qubits that can be manipulated by gates or measurements. They can be realised with many different technologies, e.g. superconducting loops, cold atoms, trapped ions, diamond vacancies, spin qubits. In the NISQ era, only imperfect (or approximate) universal quantum computers exist that cannot yet run error-correction schemes.

**Specialised Quantum Computers:** include quantum annealing (which can be realised, for instance, with superconducting loops), quantum simulators (which can be realised, for instance, by trapping cold atoms or ions and manipulating them with laser excitations), and approaches based on boson sampling (which can be realised by photonic quantum computers). In the quantum annealing approach, the applications need to be formulated as a QUBO (Quadratic Unconstrained Binary Optimisation) problem.<sup>1</sup> In the quantum simulation approach, the problem needs to be formulated in the Hamiltonian evolution form.

**Quantum Computer Emulator:** a classical computer emulating the behaviour of a quantum computer.

**Quantum Simulation or Analog Quantum Computer:** The goal of quantum simulation is to solve important quantum problems by mapping them onto controlled quantum systems in an analogue or digital way.

**Quantum Hardware Technology:** the underlying hardware technology used to implement a quantum computer.

**Quantum Approach:** the approach using the underlying quantum technologies, i.e. how to programme them.

**Quantum Annealer:** finds the global minimum of a given objective function by evolving slowly from a controlled initial state to the ground state of the target Hamiltonian.

**Digital Quantum Computer:** time evolution of qubits is described by quantum gates or measurements.

**Quantum Processing Unit:** typically referred to as a QPU (Britt and Humble 2017), it is a similar model as other types of computational hardware accelerators, such as GPUs or FPGAs, in that the classical CPU can offload all or part of an algorithm's computation to the accelerator.

---

<sup>1</sup> However formulated QUBO problems can be also addressed by other types of quantum computers.

## QC hardware technologies

Currently, there exists a diverse range of QC hardware technologies with different Technology Readiness Levels (TRL). At present, no quantum advantage has been achieved for practical applications, so the aim is to ensure that quantum technologies are ready when a break-through is reached. The requirements for QC hardware technologies, for example to ensure a noise-free environment, have a clear impact on integration in a classical HPC system.

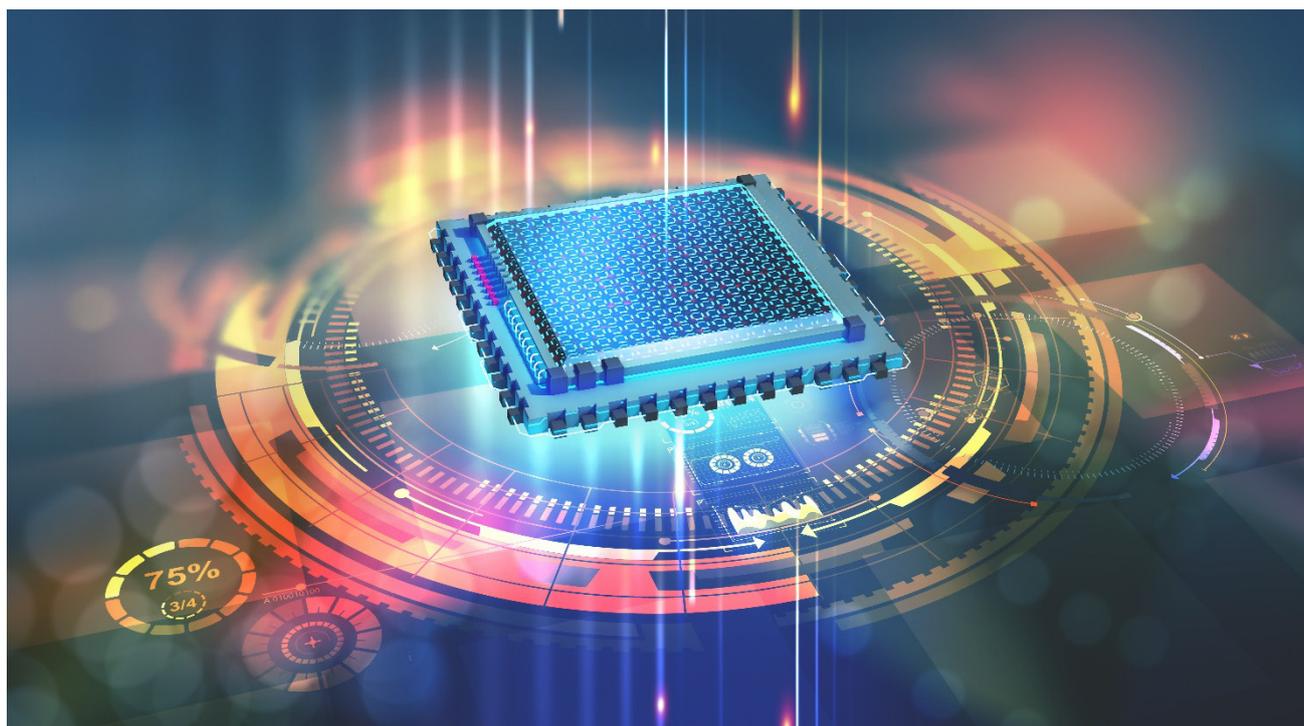
The race of QC hardware technologies is still on and we will see multiple QC technologies in use, at least for an intermediate stage of the development. These may have different characteristics, so that in the short to medium term some technologies may be best suited for specific applications, while others will be used for other application areas.

The TRLs of present quantum hardware technologies currently vary widely, and it is difficult to make a reliable prediction of future scalability, as this depends partly on the physical properties of the technologies (such as coherence times, error rates, etc.), but also on engineering challenges that are hard to predict. An overview of quantum technologies and manufacturers can be found at Bitkom (Bitkom n.d.), M. Kurek (Kurek 2020), BCG report (Bobier, et al. 2021), Quantum Computing Report (Quantum Computing Report n.d.).

All the current technologies (superconducting loops (Devoret, Wallraff and Martinis 2004), trapped ions (Häffner, Roos and Blatt 2008), Rydberg atoms / cold atoms (Henriet, et al. 2020), etc.) have one thing in common - they are relatively small systems (e.g., with a limited number of qubits) and none of them are fault-tolerant yet, which is why we are in the NISQ era.

In the long term, scalability to millions of qubits must be achieved to reach the Large Scale Quantum (LSQ) era. Several roadmaps from different hardware providers exist to reach this goal (IBM n.d.), (Veldhorst, et al. 2017), (CEA LETI n.d.). The Boston Consulting Group foresees the emergence of fault tolerant QC by the next decade (Bobier, et al. 2021). To make progress, in improving quantum hardware technologies, the following research areas remain:

- I. improve the characteristics of the qubits (fast setup and [very] long coherence times, less or no sensitivity to the environment, lower cost manufacturing, etc.);
- II. better error correction (large number of qubits required);
- III. scalability to a larger number of qubits;
- IV. ease of use (room temperature operation, chip based form factor, etc.).



## Algorithms and Applications:

This section gives a short overview of the most promising algorithms and applications, their use cases, prospects and roadmaps. A more comprehensive list of algorithms known to be amenable to quantum devices is maintained here: <https://quantumalgorithmzoo.org>. Due to the technical limitations of currently available QC technologies, only small parts of applications, e.g. certain algorithms can be transferred to quantum computers. Thus, the current strategy is to exchange only the most computationally intensive parts of HPC applications with quantum algorithms that have proven or can be assumed to scale better on quantum computers.

Currently, four domains have been identified as being the most promising for porting to quantum computers, namely hybrid algorithms, quantum chemistry, optimisation problems and quantum machine learning.

- The main idea of the hybrid quantum algorithms is to divide the problem into classical and quantum parts and give a quantum computer a task that it can do better than a classical one (see Figure 1). Currently, there are many variational algorithms for solving different problems, e.g. Variational Quantum Eigensolver (Peruzzo, et al. 2014), Quantum Approximate Optimisation Algorithm (QAOA) (Farhi, Goldstone and Gutmann, A Quantum Approximate Optimization Algorithm 2014), Variational Quantum Linear Solver, which is a variational counterpart of the HHL algorithm (Bravo-Prieto, et al. 2020), and variational counterpart of Shor's algorithm (Anshuetz, et al. 2018). Although these algorithms prove to be robust in the NISQ era, it cannot be proven that these algorithms are faster than their classical counterparts.

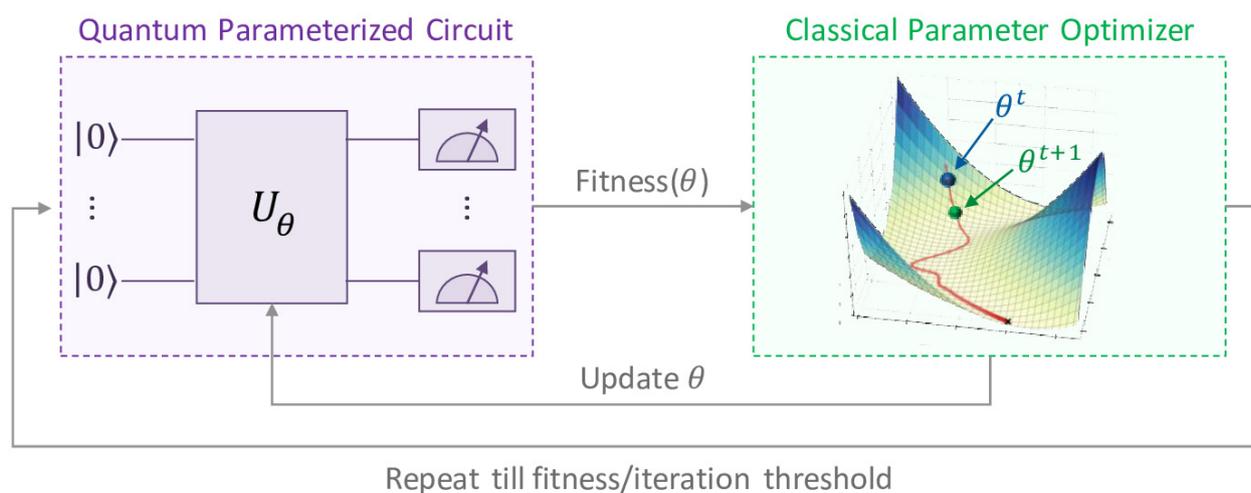


Figure 1: example of a hybrid classical / quantum algorithm from QuASER: Quantum Accelerated de novo DNA sequence reconstruction (Sarkar, Al-Ars and Bertels 2021)

- Quantum chemistry studies the ground state of individual atoms and molecules, and the excited states and transition states that occur during chemical reactions. Due to the quantum mechanical nature of molecules the scaling behaviour for the exact calculation is classically  $2^N$ , where  $N$  is the number of electron orbitals. Quantum computers can perform full configuration interaction calculations in polynomial time against the system size by adopting the Quantum Phase Estimation Algorithm (QPEA) (Kitaev 1995). However, the requirements on the error rates and the depth of a quantum circuit are too demanding to carry out the QPEA on current quantum computers. Therefore, the VQE (Peruzzo, et al. 2014) is used. Currently, the size of the total electronic system which can be simulated by QC is limited. However, it is widely believed that one of the first applications to show quantum advantage will be quantum chemistry. To achieve such a quantum advantage, it is important to focus on strongly correlated systems, where approximate methods such as DFT (Density Functional Theory) break down.

- Many industrial, financial and scientific optimisation problems can be formulated in combinatorial form with an objective, or goal, function of binary variables (the solution) that will be minimised. Typically, these problems can be mapped to well-known classical problems, such as the Traveling Salesman Problem, Graph Colouring, Scheduling, etc. Some examples are the cellular networks optimisation (Wang, Chen and Jonckheere 2016), job/task scheduling (Dalyac, et al. 2021), Portfolio Optimisation (Grant, Humble and Stump 2021). For some of these problems an optimal solution cannot be guaranteed in polynomial time by any known algorithm. To solve those problems, some example algorithms are the QAOA, a variational algorithm, which can be applied to both simulators and gate-based machines, or the Quadratic Unconstrained Binary Optimisation (QUBO) formulation (Lucas 2014).
- Quantum Machine Learning (QML) combines ideas of quantum computation with those developed in the field of machine learning to deliver alternative learning protocols which might yield a potential quantum advantage over classical counterparts, either in terms of speedup or solution quality. The most popular approach to QML currently rests on the idea of developing algorithmic tools that could accelerate or enhance specific parts of ML workflows fed with classical data on noisy near-term quantum hardware. Some of the most notable algorithms are provided by quantum k-Nearest Neighbors and k-means (Lloyd, Mohseni and Rebentrost 2013), as well as quantum Support Vector Machines (Anguita, et al. 2003) and Quantum Neural Networks based on quantum variational circuits (Farhi and Neven 2018). Most of these algorithms have already been successfully tested on academic small-sized problems. Leveraging the latest as well as the planned technological improvements in the field, it should be of strategic importance to support an early investigation of their potential application to real-world problems (computer vision, cybersecurity, natural language processing, etc.) that are relevant to European industry.

## Benchmarking

It is necessary to develop methods and tools to assess quantum computing technologies and their potential for scientific applications. To measure the performance of quantum computer benchmarks, on different levels of the system, have been suggested. All of these performance measurements have their pros and cons and are under discussion in the community.

Benchmarking on device and subsystem level (e.g. gate or circuit level protocols) is important to assess the properties and capacities of the different technologies. As we are currently in an era where these properties are difficult to reproduce and we are moving to pilot lines for bigger production, performing benchmarking at this level is important and might help with standardisation activities for the hardware technologies. Beyond gate or circuit level protocols application-centric benchmarks are necessary. However, at the current stage of technology and quantum algorithm development it is not clear the use of which algorithm will prove to be advantageous when applied to quantum and which algorithm should be developed further. Therefore, the characteristics of the most important benchmarks are not yet known and at the current stage it is important to promote a broad set of benchmarks, so that we do not remain constrained to a certain technology or algorithm too early.

Despite various initiatives, no general and widely adopted benchmark suite is currently available. A recent contribution is Atos Q-Score (Atos n.d.) (Martiel, Ayrat and Allouche 2021), with a certain level of generality and agnosticism, and which also allows for the comparison of quantum versions to classical implementations of well-known algorithms in a quantitative way. This kind of effort with additional value to Europe, should be continued and broadened.

Such extensive benchmarking on the level of algorithms and applications is still an open problem and further research in this area must be strongly supported, in close links with the application communities. It has to be noted that benchmarking at the algorithm level tests the combination of the algorithm, software stack and the technology. An evaluation of algorithms and applications needs to be done on the basis of previously defined criteria, such as time to solution or accuracy of the results.

The progress in application developments and their experimentation must feed the definition of benchmarks, i.e. benchmarks will play an important role in the hardware-software co-design that is currently necessary. It would also be important to develop a widely acknowledged ranking system, mature and flexible enough to accommodate future technology evolutions, and European players can significantly contribute to this effort.



## Software-Hardware stack

In the current era of developing and deploying quantum computers, most attention is drawn to the development of the underlying hardware technology. To address the hardware a dedicated software stack is necessary to bring up the quantum computer (including calibration, characterisation, optimal control, device simulation), to operate the system (including control electronics, scheduling, authentication, recalibration, re-characterisation) and to translate the user's code to run on the system (from high level languages and algorithms to quantum circuit tablature which can be executed on the hardware). All parts of the software stack exist today and are in many cases provided by the companies developing quantum computers (Pulser n.d.) (Google Quantum AI n.d.) (Qiskit n.d.), it is however clear that most software components can still be significantly improved. Here, standardisation can be key in order to allow a fixed interface design between the software components, e.g. an internal representation of QC circuits of a transpiler is key in order to generalise compiler construction. In addition, new challenges will occur with larger quantum computers, e.g. optimal mapping of an algorithm on a realistic quantum computer is an optimisation problem in itself.

The programming tools available to application developers are dominated by vendor-specific tools. There are examples of other technologies or vendor systems that are supported, such as Atos's myQLM, but most tools work at a low level of abstraction. An example of a higher-level programming language is Silq (Zurich, Switzerland) (Bichsel, Baader and Gehr 2020). Another possibility to abstract the programming from the hardware is the use of an intelligent compiler as pursued by Classiq (Israel) (Classiq n.d.).

From an HPC perspective the scheduler or runtime system which schedules the tasks from the (hybrid) applications and the (re-)calibration is most important. Especially using state-of-the-art iterative hybrid programs the total runtime depends on the latency between classical and the quantum system. In the loose integration model (see in System integration section), the interaction between the host computer and the coupled QPU unit may be performed in different ways: a 'standard' batch scheduler may run on the host node and activate the code running on the host node, which then activates the code running on the QPU. This model of operation requires that a common area (memory, HBM, NVMe) is shared between both units to enable the transfer of data between the two. A more complex but potentially more performing model may be that the QPU is considered as a thread, and it is activated by the program running on the host node via a 'wake-up' call that performs the data transfer and instructs the QPU to start the computation. This model may selectively use the common area (memory, HBM, NVMe) or the call for the transfer of the data, according to the size of the data to be transferred.

Due to the different underlying technologies including specifics of the quantum computers and different software stacks provided by the QC manufacturers, it is nowadays difficult to distinguish if a certain system performs better due to the underlying technology or a better optimised software stack.

## System integration

High-Performance Computing (HPC) has historically taken advantage of new specialised hardware. Graphical Processing Units (GPUs) and Field Programmable Gate Array (FPGAs) are examples of the trend in accelerator use for HPC development. A primary motivation for the accelerator paradigm is that low-level computational kernels can leverage specialised hardware while minimising changes to overall program structure: this approach isolates the need for program's or algorithm's refactoring efforts to those kernels specific to the hardware accelerator. A secondary motivation is that the accelerator model offers an opportunity to explore any emerging technologies, while also reducing the technical risk with respect to the overall system development. The opportunities offered by QC represent a challenge to the accelerator model, which has been up to now implemented exclusively within the framework of the classical, deterministic Turing model.

In addition to the challenges described above, it is not yet clear how the workflow of hybrid quantum applications will look like in the future, and different models of integration quantum and classical machines are being explored. Some large data centres such as CEA, CINECA and Jülich Supercomputing Centre (JSC) plan on integrating QC in their supercomputing facilities and workflows, while, for example, the LUMI consortium is also considering other distributed approaches.

We distinguish two main models of integration: loose and tight.

### 1. Loose integration model:

It describes a case where the QPU remains physically separated from the host system, which can be an HPC or an intermediate server. The two systems can communicate through a network interface (e.g. Infiniband, CXL). This is the classical client-server model as shown in Figure 1 where the QC server may either be on a dedicated connection with the host node or part of a larger computational grid. In this model, the network takes care of communicating the requests between the host server system and the QC (client). Access to each QPU must still be provisioned by the control system and this control system may be a switch that forwards program and data to individual QPUs or an 'intelligent' routing algorithm dispatching the workload based on QPU usage and demand. The entry point into the system remains the primary bottleneck as well as the single point of failure.

The QPU can be located on-premise or on a remote server.

For the loose integration model, communication latency can be eliminated in both the co-located and remote server models, thus maximising the bandwidth and computation speed-up of the HPC+QC infrastructure. Further, as technology matures to allow deeper quantum algorithms, required for actual quantum advantage, the execution time on the QPU will grow, thus alleviating latency issues. Some practical issues do set the co-located and remote-server models apart. Arguments in favour of the co-located model include:

- Easier data security implementation, especially when it comes to dealing with sensitive (e.g. industrial trade secrets) or classified data (e.g. for governmental agencies).
- Synergies from having research and system administration staff of both HPC and QC at the same site.

Arguments in favour of the remote-server model include:

- In a distributed approach, both HPC and QC can be optimally accommodated. Data centres designed for HPC systems are suboptimal for quantum computers, and vice versa. QPUs need to be properly shielded from environmental noise, and might even need to be located underground. Further, HPC installations consume significantly more energy, and are best located in areas with lower cooling needs and cheap green energy sources; from an energy-consumption perspective, quantum computers can be located anywhere.
- Higher modularity, increasing inclusiveness of different quantum technological solutions developed in Europe.

As the technology is still emerging, hardware updates/improvements are frequent. This means that the QC hardware provider's engineers often have to intervene at the site of the quantum computer, to implement updates and/or maintenance. Hence, when selecting the hardware provider, its geographical proximity to the

site where the quantum computer is installed is to be preferred. For example, this is the case for an IBM Quantum System One installed at Ehningen in Germany as part of a strategic partnership between Fraunhofer and IBM. Thus, the remote-server model facilitates adding, for example, QPUs located in research labs to the HPC+QC ecosystem.

On-premise and distributed remote server-based schemes should be considered complementary. At this stage, Europe needs to hedge its bets, and explore both strategies. In the framework of the European project HPC-QS, two quantum simulators (QPU working in the analog regime) will be installed in two European HPC facilities: TGCC in France and JSC in Germany. This pilot project is a cornerstone towards a European platform, providing quantum capabilities across Europe. It can be seen as a testbed for on-premise access, to assess its advantages and drawbacks.

**2. Tight integration model:**

This model assumes that the hardware and software requirements for QPUs can be somehow combined within the hardware and software requirements for the host node, to achieve a single coherent tightly connected system. For some QC hardware technologies, it might be possible in the long run to integrate QPUs directly on the classical chips.

The tight integration model will require multiple and significant advances in the technology currently used.

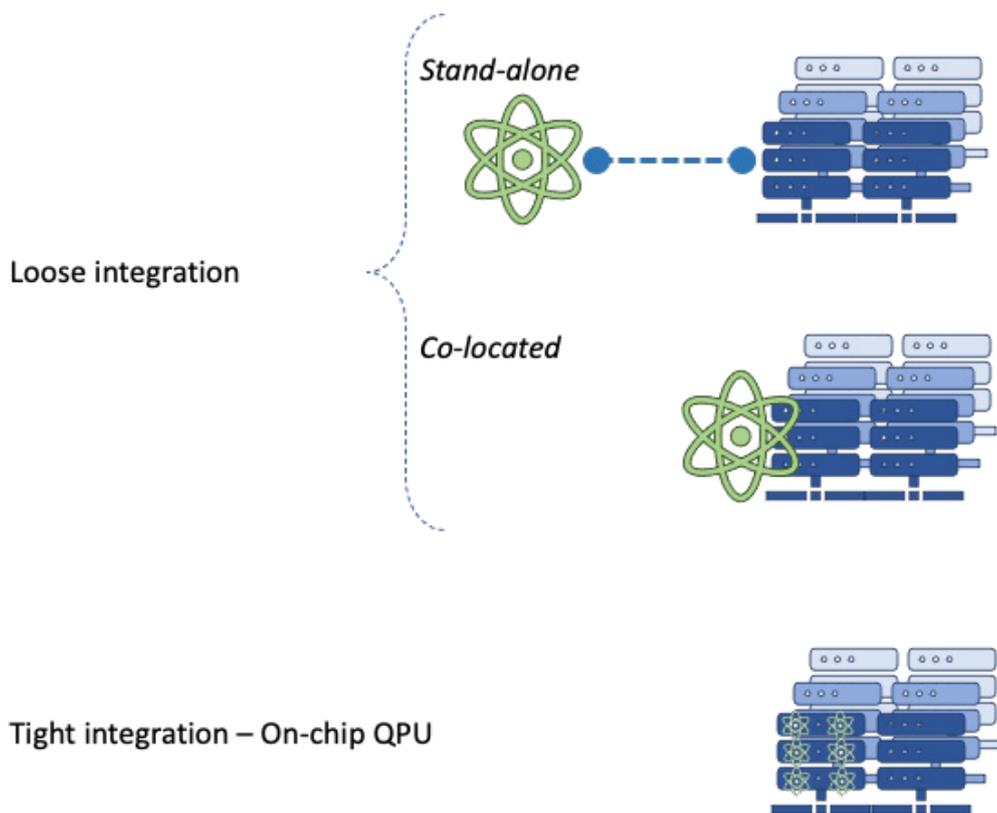


Figure 2. Representation of different levels of integration of QC systems with classical systems (tight or loosely coupled)

## Emulation of Quantum Computers with HPC devices

Although enormously expensive in cost of computational resources, the practice of emulating QPUs using current supercomputers turns out to be a very powerful tool for several reasons, one of which is the benchmarking of quantum algorithms. If on the one hand it is true that the current most developed quantum computers have a number of qubits that are prohibitively slow and costly to emulate with the supercomputers we have available today, on the other hand it is true that a classical computer is able to emulate perfect qubits, i.e. without all those defects that are common to quantum computers currently available. In particular, by emulating a QPU with a supercomputer it is possible to obtain a system of qubits with an infinite coherence time and a perfect fidelity, without errors in the application of quantum gates and with complete connectivity. This makes emulators an irreplaceable tool for designing, analysing and benchmarking quantum algorithms.

Why use HPC to emulate quantum computers? For two main reasons: storage and parallel computing. Storage because emulating a quantum state potentially occupies a lot of RAM: to be convinced of this, just think that to completely describe a quantum state formed by  $N$  qubits it is necessary to use an amount of memory proportional to  $2^N \cdot 128\text{bits}$ .

Parallel computing (both CPU and GPU), on the other hand, becomes necessary in case of very complex operations. In fact, emulating the evolution of a quantum state requires many linear algebra operations, to be carried out between elements with dimensions proportional to  $2^N$ , where  $N$  is the number of qubits involved.

For the perfect emulation of huge quantities of qubits, therefore, it is necessary to use HPC techniques and supercomputers as powerful as possible. In Europe we have several computing centres and industries capable of providing adequate computational resources. What is missing, right now, is a common vision that leads all European computing centres to work under a single unified access platform, a common language for all quantum developers who want to rely on the resources made available. Logging in at any of the EuroHPC sites should ideally give access to a large selection of QC resources.

An effort in this direction has been implemented within the EuroHPC "Pilot on Quantum Simulator" project, in which many of the most important European computing centre are the main players. The objective of the pilot action will be to develop, deploy and coordinate at European level a European quantum simulation (QS) infrastructure of circa 100+ interacting quantum units, which will be accessible via the cloud on a non-commercial basis to public and private European users.

In addition to the EuroHPC pilot project, what needs to be achieved is a convergence of vendor-independent tools for use by all European researchers. In fact, there are currently several quantum emulators on the market. The vast majority, even if of an open source nature, were developed by the quantum computer manufacturers themselves. Also, many of them do not support HPC implementations like distributed computing via Message programming services and/or GPU acceleration.

It is important to consider not only "perfect" emulators of general purpose systems, but also to extend to other promising algorithms capable of emulating quantum computers, first of all the tensor network method. The tensor network method is a factorisation of large tensors into a network of smaller tensors. This is useful for representing a quantum state, and is used to find particular properties of the system, such as its ground state energy. The most successful algorithm using tensor networks is the DMRG ("Density Matrix Renormalisation Group"), which allows us to find properties of systems of hundreds of qubits. To obtain these results, however, it is necessary to apply approximations, which inevitably introduce errors in the calculations, undermining the perfection otherwise guaranteed by state-vector emulators.

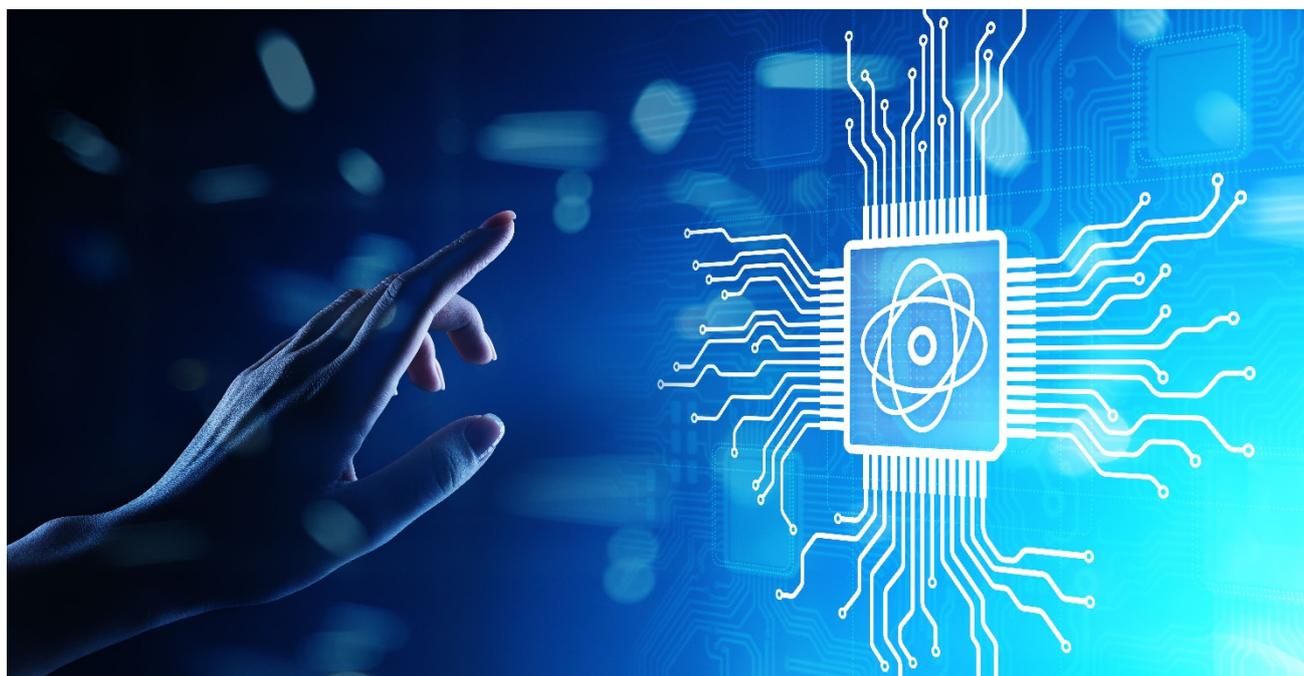
## Conclusions

From the HPC perspective, quantum computing is a novel accelerator with possible future advantages in certain application areas such as quantum chemistry, optimisation and quantum machine learning, to name but a few. The final successful hardware technology quantum computing will be built on is not yet known, with a number of technologies competing and possibly several technologies being the eventual winners. Therefore, it is important for Europe not to settle solely on one technology too early but to compare different technologies and build a software ecosystem that allows easy access to at least the most promising hardware technologies available. It also means that Europe must continue its R&D to obtain practical quantum devices. The different research avenues are known and the efforts must be relentless.

In the NISQ era, hybrid algorithms (e.g. variational algorithms) are an important strategy for dividing the algorithm into classical (e.g. HPC) and quantum (e.g. QC) parts, and then iterating between them to calculate the solution. The integration between the HPC and QC becomes important in order to facilitate the adoption of practical quantum solutions by a broad community in the short to medium term. However, the eventual design of the HPC/QC integration, the principal applications and the final workflow are not yet defined well. Thus, new generations of benchmarks will be essential to evaluate the various approaches in a consistent manner and provide quantifiable results in order to compare the various alternatives.

Finally, a comprehensive and efficient quantum software stack needs to be built to anticipate the advent of larger quantum systems. Using emulators will bootstrap its development, without waiting for actual large scale hardware. Additionally, the development and adoption of common APIs and entry points will help harmonise differing implementations and reduce the impact of the inevitable technology churn in quantum technologies.

Europe has the technologies and skills necessary to successfully meet all these challenges, while progressing into a leadership position in the future.



## References

- Anguita, Davide, Sandro Ridella, Fabio Riveccio, and Rodolfo Zunino. 2003. "Quantum optimization for training support vector machines." *Neural Networks* 16 (5-6): 763-770. doi:[https://doi.org/10.1016/S0893-6080\(03\)00087-X](https://doi.org/10.1016/S0893-6080(03)00087-X).
- Anschuetz, Eric R., Jonathan P. Olson, Alán Aspuru-Guzik, and Yudong Cao. 2018. "Variational Quantum Factoring." <https://arxiv.org/abs/1808.08927>.
- Atos. n.d. "Q-Score: measure what truly matters." Accessed 2021. <https://atos.net/en/solutions/q-score>.
- Bichsel, Benjamin, Maximilian Baader, and Timon Gehr. 2020. "Silq: a high-level quantum language with safe uncomputation and intuitive semantics." *PLDI 2020: Proceedings of the 41st ACM SIGPLAN Conference on Programming Language Design and Implementation*. 286-300. doi:<https://doi.org/10.1145/3385412.3386007>.
- Bitkom. n.d. "Bitkom-Übersicht Deutsches Quanten-Ökosystem v1.1." Accessed 2021. [https://www.bitkom.org/sites/default/files/2021-03/deutsches\\_quanten-okosystem\\_v1.1\\_public.pdf](https://www.bitkom.org/sites/default/files/2021-03/deutsches_quanten-okosystem_v1.1_public.pdf).
- Bobier, Jean-François, Matt Langione, Edward Tao, and Antoine Gourévitch. 2021. "What Happens When 'If' Turns to 'When' in Quantum Computing?" *BCG*. 21 07. <https://www.bcg.com/fr-fr/publications/2021/building-quantum-advantage>.
- Bravo-Prieto, Carlos, Ryan LaRose, M. Cerezo, Yigit Subasi, Lukasz Cincio, and Patrick J. Coles. 2020. "Variational Quantum Linear Solver." <https://arxiv.org/abs/1909.05820>.
- Britt, Keith A., and Travis S. Humble. 2017. "High-Performance Computing with Quantum Processing Units." *ACM Journal on Emerging Technologies in Computing Systems* 13 (3): 1-13. doi:<https://doi.org/10.1145/3007651>.
- CEA LETI. n.d. "Five advantages of silicon spin." Accessed 2021. <https://www.leti-cea.com/cea-tech/leti/english/Pages/Applied-Research/Strategic-Axes/Quantum-Computing/Fundamental-Advantage-of-Silicon-Spin/Five-advantages-of-silicon-spin.aspx>.
- Classiq. n.d. <https://www.classiq.io/>.
- Dalyac, Constantin, Loïc Henriët, Emmanuel Jeandel, Wolfgang Lechner, Simon Perdrix, Marc Porcheron, and Margarita Veshchezerova. 2021. "Qualifying quantum approaches for hard industrial optimization problems. A case study in the field of smart-charging of electric vehicles." *EPJ Quantum Technology* 8: 12. doi:<https://doi.org/10.1140/epjqt/s40507-021-00100-3>.
- Deutsch, David. 1985. "Quantum theory, the Church-Turing principle and the universal quantum computer." *Proceedings of the Royal Society A (Royal Society)* 400 (1818). doi:<https://doi.org/10.1098/rspa.1985.0070>.
- Devoret, M H, A Wallraff, and J M Martinis. 2004. "Superconducting Qubits: A Short Review." <https://arxiv.org/abs/cond-mat/0411174>.
- Farhi, Edward, and Hartmut Neven. 2018. "Classification with Quantum Neural Networks on Near Term Processors." <https://arxiv.org/abs/1802.06002>.
- Farhi, Edward, Jeffrey Goldstone, and Sam Gutmann. 2014. "A Quantum Approximate Optimization Algorithm." <https://arxiv.org/abs/1411.4028>.
- Google Quantum AI. n.d. "Cirq." Accessed 2021. <https://quantumai.google/cirq>.
- Grant, Erica, Travis S. Humble, and Benjamin Stump. 2021. "Benchmarking Quantum Annealing Controls with Portfolio Optimization." *Physical Review Applied* 15 (1): 014012. doi:<https://doi.org/10.1103/PhysRevApplied.15.014012>.
- Häffner, H., C.F. Roos, and R. Blatt. 2008. "Quantum computing with trapped ions." *Physics Reports (Elsevier)* 469 (4): 155-203. doi:<https://doi.org/10.1016/j.physrep.2008.09.003>.
- Henriët, Loïc, Lucas Beguin, Adrien Signoles, Thierry Lahaye, Antoine Browaeys, Georges-Olivier Reymond, and Christophe Jurczak. 2020. "Quantum computing with neutral atoms." *Quantum* 4: 327. doi:<https://doi.org/10.22331/q-2020-09-21-327>.
- IBM. n.d. "IBM's roadmap for scaling quantum technology." Accessed 2021. <https://research.ibm.com/blog/ibm-quantum-roadmap>.
- Kitaev, A. Yu. 1995. "Quantum measurements and the Abelian Stabilizer Problem." *Electronic Colloquium on Computational Complexity (ECCC)*. <https://arxiv.org/abs/quant-ph/9511026>.
- Kurek, Michel. 2020. "Technologies quantiques: vers la seconde révolution." [https://www.researchgate.net/publication/350521248\\_TECHNOLOGIES\\_QUANTIQUES\\_VERS\\_LA\\_SECONDE\\_REVOLUTION](https://www.researchgate.net/publication/350521248_TECHNOLOGIES_QUANTIQUES_VERS_LA_SECONDE_REVOLUTION).
- Lloyd, Seth, Masoud Mohseni, and Patrick Rebentrost. 2013. "Quantum algorithms for supervised and unsupervised machine learning." <https://arxiv.org/abs/1307.0411>.
- Lucas, Andrew. 2014. "Ising formulations of many NP problems." *Frontiers in Physics* 2: 5. doi:<https://doi.org/10.3389/fphy.2014.00005>.

- Martiel, Simon, Thomas Ayrat, and Cyril Allouche. 2021. "Benchmarking Quantum Coprocessors in an Application-Centric, Hardware-Agnostic, and Scalable Way." *IEEE Transactions on Quantum Engineering* 2. doi:<https://doi.org/10.1109/TQE.2021.3090207>.
- Peruzzo, Alberto, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. 2014. "A variational eigenvalue solver on a photonic quantum processor." *Nature Communications* 5: 4213. doi:<https://doi.org/10.1038/ncomms5213>.
- Preskill, John. 2018. "Quantum Computing in the NISQ era and beyond." *Quantum* 2: 79. doi:<https://doi.org/10.22331/q-2018-08-06-79>.
- Pulser. n.d. Accessed 2021. <https://pulser.readthedocs.io>.
- Qiskit. n.d. Accessed 2021. <https://qiskit.org/>.
- Quantum Computing Report. n.d. "Players." Accessed 2021. <https://quantumcomputingreport.com/players/>.
- Sarkar, Aritra, Zaid Al-Ars, and Koen Bertels. 2021. "QuASer: Quantum Accelerated de novo DNA sequence reconstruction." *PLoS ONE* 16 (4). doi:<https://doi.org/10.1371/journal.pone.0249850>.
- Veldhorst, M., H. G. J. Eenink, C. H. Yang, and A. S. Dzurak. 2017. "Silicon CMOS architecture for a spin-based quantum computer." *Nature Communications* 8: 1766. doi:<https://doi.org/10.1038/s41467-017-01905-6>.
- Wang, Chi, Huo Chen, and Edmond Jonckheere. 2016. "Quantum versus simulated annealing in wireless interference network optimization." *Scientific Reports* 6: 25797. doi:<https://doi.org/10.1038/srep25797>.

## Funded projects

**HPCQS:** The project HPCQS (coordinated by FZJ) aims to build a European pilot infrastructure providing Pasqal quantum simulators integrated into modular HPC systems for quantum HPC hybrid simulations. The infrastructure is being developed in a co-design process together with selected use cases.

**European Quantum Flagship Projects:** (<https://qt.eu/>) The Quantum Technologies Flagship is a long-term research and innovation initiative that aims to put Europe at the forefront of the second quantum revolution.

Part of the European Flagship in Quantum Technologies, the NEASQC project (<https://www.neasqc.eu/>) is dedicated to the research in industrial applications of quantum computing. It brings together 12 academic and industrial players around 9 practical use cases.

Several projects within the flagship concentrate on building quantum computers or simulators based on different technologies:

- [AQTION](#): Advanced quantum computing with trapped ions.
- [PASQuanS](#): Programmable atomic large scale quantum simulation.
- [MicroQC](#): Microwave driven ion trap quantum computing.
- [OpenSuperQ](#): An open superconducting quantum computer.
- [PhoQuS](#): Photons for quantum simulation
- [Qombs](#): Quantum simulation and entanglement engineering in quantum cascade laser frequency combs.
- [QLSI](#): Quantum large scale integration in Silicon.
- [S2QUIP](#): Scalable two-dimensional quantum integrated photonics.
- [SQUARE](#): Scalable rare-earth ion quantum computing nodes.

---

## Authors:

Dr. Valeria Bartsch is a senior scientist at Fraunhofer ITWM, Germany, focusing on HPC programming models and the development of quantum computing software.

Dr-Ing Guillaume Colin de Verdière is an international expert in HPC at CEA, France, focusing on technologies for future supercomputers.

Dr-Ing Jean-Philippe Nominé is an international expert in HPC at CEA, France, in charge of HPC Strategic Collaborations.

Dr. Daniele Ottaviani is a mathematician, HPC expert and coordinator of the Quantum Computing Lab, a division of CINECA, the largest Italian supercomputing center.

Dr. Daniele Dragoni is a computational physicist and materials scientist, he is responsible for the quantum computing research activity at Leonardo company, Italy.

Dr. Chayma Bouazza is Business Development Manager at Pasqal, in charge of developing commercial activities and international expansion of the company, France.

Fabrizio Magugliani is Strategic Planning and Business Development at E4 Computer Engineering. Fabrizio's interests are in how innovative technologies can be integrated in the current infrastructures.

David Bowden is a Senior Principal Research Scientist at Dell Technologies, based at its OCTO research office in Ireland.

Dr. Cyril Allouche is VP & Fellow, head of Quantum Computing R&D at Atos, France.

Dr. Mikael Johansson is Quantum Strategist at CSC– IT Center for Science, Finland, and Vice Director of the Finnish Quantum Computing Infrastructure.

Dr. Olivier Terzo is a Computer Engineer and Head of the Advanced Computing, Photonics and Electromagnetics research domain at LINKS Foundation in Turin, Italy.

Dr. Andrea Scarabosio is a computational physicist and Senior Researcher in the Advanced Computing, Photonics and Electromagnetics research domain at LINKS Foundation in Turin, Italy.

Giacomo Vitali is a computational physicist and Senior Researcher in the Advanced Computing, Photonics and Electromagnetics research domain at LINKS Foundation in Turin, Italy.

Dr. Farida Shagieva is a research scientist at HLRS in Stuttgart, Germany, focusing on the development of quantum algorithms.

Prof. Dr. Kristel Michielsen is head of the research group Quantum Information Processing and the Jülich UNified Infrastructure for Quantum computing (JUNIQ) at the Jülich Supercomputing Centre, Germany

---

Cite as: V. Bartsch et al., < QC | HPC >: Quantum for HPC, ETP4HPC White Paper, 2021, doi 10.5281/zenodo.5555960

DOI: 10.5281/zenodo.5555960

© ETP4HPC 2021