Quantum emulation, what it is and how it works

Why quantum emulation is the sensible move right now

Published on 17/08/2026 · humanq.it

Table of contents

Quantum emulation lets you develop, test and validate quantum algorithms today, on classical high performance infrastructure, without waiting for the hardware to mature and without paying for access to real machines. The word "emulation" still circulates mostly among specialists, yet it describes a practice that companies and research groups have already adopted, because it is the most stable way to work on the quantum computer. Here you will find the definition, the difference from quantum simulation, the physical limits of the approach, the economic and operational advantages, the industrial use cases and how emulation platforms work.

What is quantum emulation?

Quantum emulation reproduces on classical computers the logical behaviour of a quantum processor, meaning the qubits, the gates, the measurement and, where required, the noise as well. The emulator runs the very same circuit that would run on real hardware and returns the same results, with one difference that changes the way you work, because the complete quantum state stays explicitly stored and can be inspected at any point during execution, whereas the hardware behaves like a black box and returns nothing but measurement statistics. On an emulator you watch the amplitudes evolve gate after gate, every run repeats identically, and an anomaly can be isolated precisely instead of drowning in statistical scatter.

What is the difference between quantum emulation and simulation?

The two terms are often used interchangeably, yet they describe different activities. Emulation reproduces the machine, that is the quantum processor with its logical architecture, and to do so it computes the evolution of state vectors and density matrices on a classical computer (CPU/GPU), so that a circuit written for real hardware runs on it unchanged. Simulation instead builds an abstract model of a system's behaviour, without binding itself to the architecture of the machine that executes it. The distinction has immediate practical consequences, because anyone developing an algorithm meant for a specific processor needs an emulator that guarantees compatibility and repeatability, whereas anyone studying a material or a molecule can turn to simulation methods that sit far away from today's circuit architecture. The second case is explored on the page devoted to quantum simulation.

Why emulate instead of using a real quantum computer?

The reasons are economic and operational, and every one of them can be measured.

- **Cost.** Access to real machines is billed by execution time, while an emulation infrastructure has predictable costs.
- **Availability.** There are no booking queues on shared systems, so the work proceeds when the team needs it.
- **Determinism.** Today's hardware is still very noisy and every run returns something slightly different, whereas the emulator repeats exactly the same result, a condition without which debugging an algorithm becomes impractical.
- **Inspectability.** On emulated states you can measure everything without destroying them, and this shortens the development time of quantum algorithms.
- **Noise control.** In emulation there is no physical hardware, so decoherence does not appear by itself as it does on real machines, and error models are introduced deliberately and calibrated at will, in order to anticipate how the circuit will behave on tomorrow's devices, including error correction.

Then there is the organisational factor, because an emulation infrastructure running on classical hardware fits into existing IT processes and is governed with the systems skills already present in the company, with no cryogenics and no binding remote access contracts.

How does a quantum emulator work technically?

A quantum emulator represents the state of the system as an array of numbers describing every possible combination of the qubits involved, and each logic gate applied to the circuit becomes a computation on that array, executable in parallel on CPUs and GPUs. The array, however, grows exponentially with the number of qubits, so beyond a certain threshold you move to alternative representations designed for circuits with particular features. Choosing the representation that suits the circuit is the most delicate piece of engineering in building an emulator. The most recent implementations distribute the computation across several nodes, and in that case interconnect bandwidth becomes the real bottleneck, well before available memory.

HumanQ s.r.l.

P.IVA e C.F.: 04790260980
Via delle Scuole 1/G, 25128 Brescia (BS)
030 64 90 099 – info@humanq.it
humanq@pec.it

The state vector representation

The most direct representation is the state vector, an array of 2^n complex amplitudes for n qubits, normally in double precision, that is 16 bytes per amplitude. Each gate is a structured matrix-vector multiplication that parallelises on CPUs and GPUs. Memory grows exponentially, because the 2^{30} amplitudes required for 30 qubits take up roughly 17 GB, while the 2^{40} for 40 qubits reach about 17 TB. Beyond that threshold alternative representations come into play, such as tensor networks that exploit the structure of the circuit, density matrix methods for noise, and stabiliser techniques for Clifford circuits.

How many qubits can be emulated on classical hardware?

With the full state vector, a well equipped server handles up to 30-35 qubits, while dedicated HPC infrastructure goes beyond 40, the threshold past which the memory required doubles with every qubit added. Tensor networks push the limit much further when circuits have low depth or limited entanglement, to the point that instances of dozens of qubits once considered out of reach have already been emulated. For a company what matters most is a practical point, namely that the vast majority of today's industrial prototypes, from optimisation studies to quantum machine learning, sits comfortably within these limits.

What does quantum emulation offer research and training?

In research, access to the complete state makes it possible to measure exactly those quantities that on hardware are estimated through expensive statistical procedures, such as entanglement entropies, fidelity and spectra. In training the benefit is didactic, because the learner watches superposition build up and interference act, instead of having to infer them from noisy histograms. For corporate teams emulation is the gym in which to build skills before they become urgent, since developers train on real tools, with fast learning curves and without depending on foreign hardware vendors, and the circuits written today in emulation will run on future machines as they are. There is also a less obvious effect, since every experiment leaves behind a reproducible model, which in research makes peer review easier, while in a company it allows results to be kept and compared over time.

Which industrial use cases are developed in emulation?

Today emulation work concentrates on three families of problems.

- **Optimisation.** Fleet routing, production scheduling, portfolio allocation, problems where variational algorithms are prototyped entirely in emulation, with effects on finance and logistics.
- **Molecular simulation** for pharmaceuticals and materials, where the emulator validates the circuits before they move to real machines, with applications in healthcare.
- **Security.** Assessing the impact of quantum attacks and the transition to post-quantum cryptography, a matter of national strategic relevance connected to the Golden Power perimeter as well.

The full overview is in quantum computing applications. In all three cases the useful work lies in the systematic comparison with classical methods, because that is how you learn where quantum promises an economic margin and where it does not, before committing to an investment.

How to start a quantum emulation project?

An emulation project normally unfolds in four stages, over a timescale measured in months.

1. **Assessment.** The processes that present computationally critical problems are identified and their quantum potential is estimated.
2. **Training.** The team acquires the operational basics directly on the platform.
3. **POC - Proof of concept.** A single use case is prototyped in emulation, with metrics that compare it against the classical solutions already in use.
4. **Roadmap.** You plan how the prototype evolves, the criteria for moving to real hardware and the oversight of corporate quantum readiness, including the cryptographic front on which NIST has already set the standards.

To choose the first use case it pays to look for a problem small enough to be completed in a few weeks and real enough to have a measurable classical benchmark, because it is the comparison with the baseline that produces learning, whereas a demonstration for its own sake leaves the team with very little.

What is the EQM emulation platform?

EQM is a quantum emulation engine of the Italian Quantum Emulation Centre. It serves to support companies, universities and institutions in running quantum circuits on classical high performance infrastructure, with state inspection tools, configurable noise models and reproducible working environments. The platform accompanies the entire life cycle of a project, from training the team to prototyping the algorithm, from validation with realistic noise to preparing the code for execution on the real hardware available today and on whatever becomes ready in the future. Being a national infrastructure, governable and verifiable, the choice also carries a technological autonomy dividend for the country. Access is through remote working environments and programming interfaces compatible with the most widespread frameworks, so that code written elsewhere moves onto the platform without rewriting.

What is the Italian Quantum Emulation Centre?

The Italian Quantum Emulation Centre is the initiative that turns EQM into a system, a national centre with the emulator in Pavia and the server infrastructure in Genoa, presented publicly in July 2026 with application demonstrations in defence, healthcare and mobility. Two objectives are stated, because the Centre gives Italian industry and academia a sovereign point of access to quantum computing technologies, and at the same time builds a national skills supply chain in a domain classified among critical technologies. In a European context where the Quantum Flagship is pushing for strategic autonomy, this is Italy's answer on the emulation front.

Bring your use case into quantum emulation

HumanQ supports companies and institutions from assessment through to proof of concept. Assess your organisation's quantum readiness or contact the HumanQ team at info@humanq.it to design your first emulation experiment together.

Tags: qubit, quantum emulation · Article published on humanq.it/en/quantum-emulation