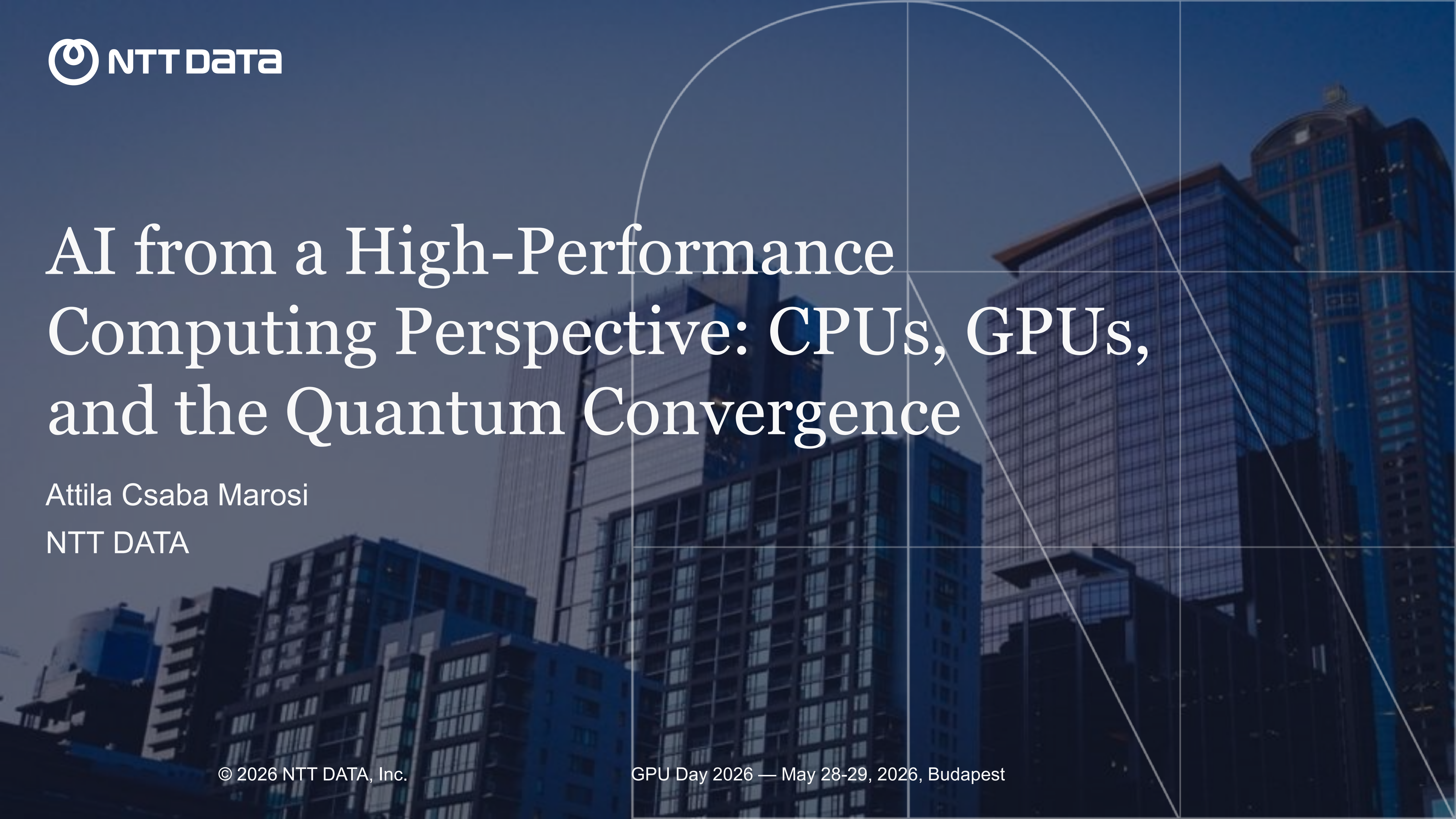


# AI from a High-Performance Computing Perspective: CPUs, GPUs, and the Quantum Convergence

Attila Csaba Marosi

NTT DATA

GPU Day 2026 — May 28-29, 2026, Budapest

# NTT DATA — Global at a glance

Global telecommunications and IT services provider, headquartered in Tokyo, Japan. NTT DATA delivers infrastructure, cloud and AI services across regulated industries worldwide.

**80+**
Countries

**~198K**
Employees

**Tokyo**
Global HQ

**IT & Telecom**
Services focus

## Key strategic priorities

### Data center growth

Hyperscale expansion across key regions

### AI & IoT integration

Embedded intelligence across services

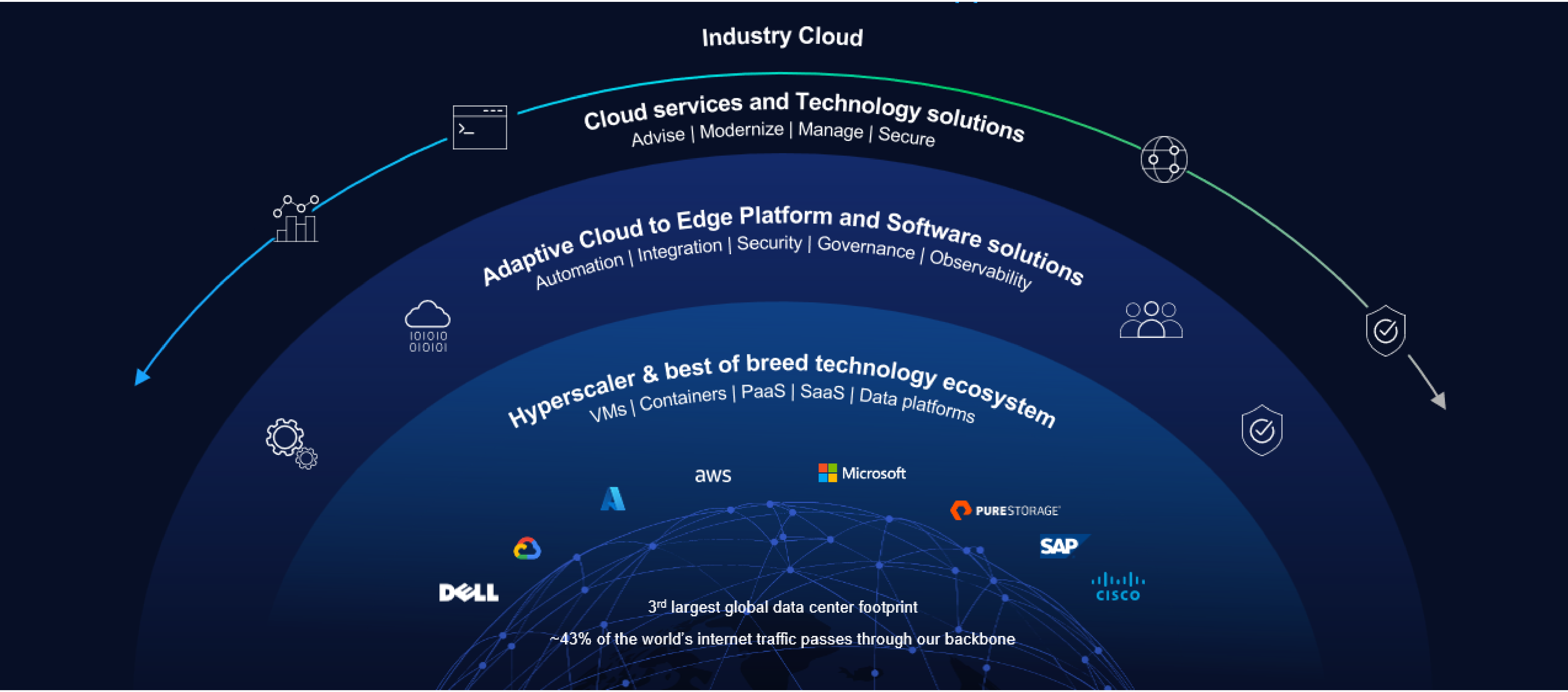
### Sustainability

Carbon neutrality goals and green operations

### 5G & next-gen networks

Advanced connectivity infrastructure

# Our technology stack and partner ecosystem



Industry Cloud

**Cloud services and Technology solutions**
Advise | Modernize | Manage | Secure

**Adaptive Cloud to Edge Platform and Software solutions**
Automation | Integration | Security | Governance | Observability

**Hyperscaler & best of breed technology ecosystem**
VMs | Containers | PaaS | SaaS | Data platforms

3rd largest global data center footprint

~43% of the world’s internet traffic passes through our backbone

# NTT Hungary

Hungarian subsidiary delivering enterprise technology services since 2001, backed by NTT DATA's global capabilities and partner ecosystem.

**2001**
Established

**180+**
Employees

**Budapest**
Local office

**ISO 27001**
Certified

## Core services delivered locally

**Data centers & networks**
Enterprise data center and network infrastructure

**Cloud & infrastructure**
Hybrid cloud, server, and storage solutions

**AI, GenAI & AIOps**
Generative AI solutions and AI-driven IT operations

**Cybersecurity**
Security advisory, monitoring, and response

# Supercomputers – High Performance Computing

**Definition:** *A computer system that can solve complex computational tasks far faster than average for the current technology level — within a timeframe that makes the result relevant.*

**Typical building blocks**

- Classical components: CPU, GPU, memory, interconnect, storage, network
- Massively parallel architecture, UNIX/Linux OS
- Batch processing with a job scheduler
- Parallel programming environments — MPI, OpenMP

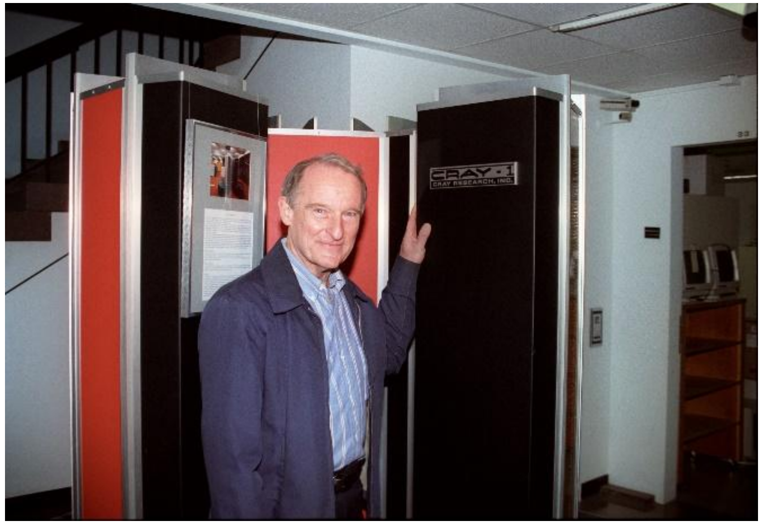
**Core use cases**

- Simulation and modeling
- Big-data analytics
- AI (training and inference)

**A short history**

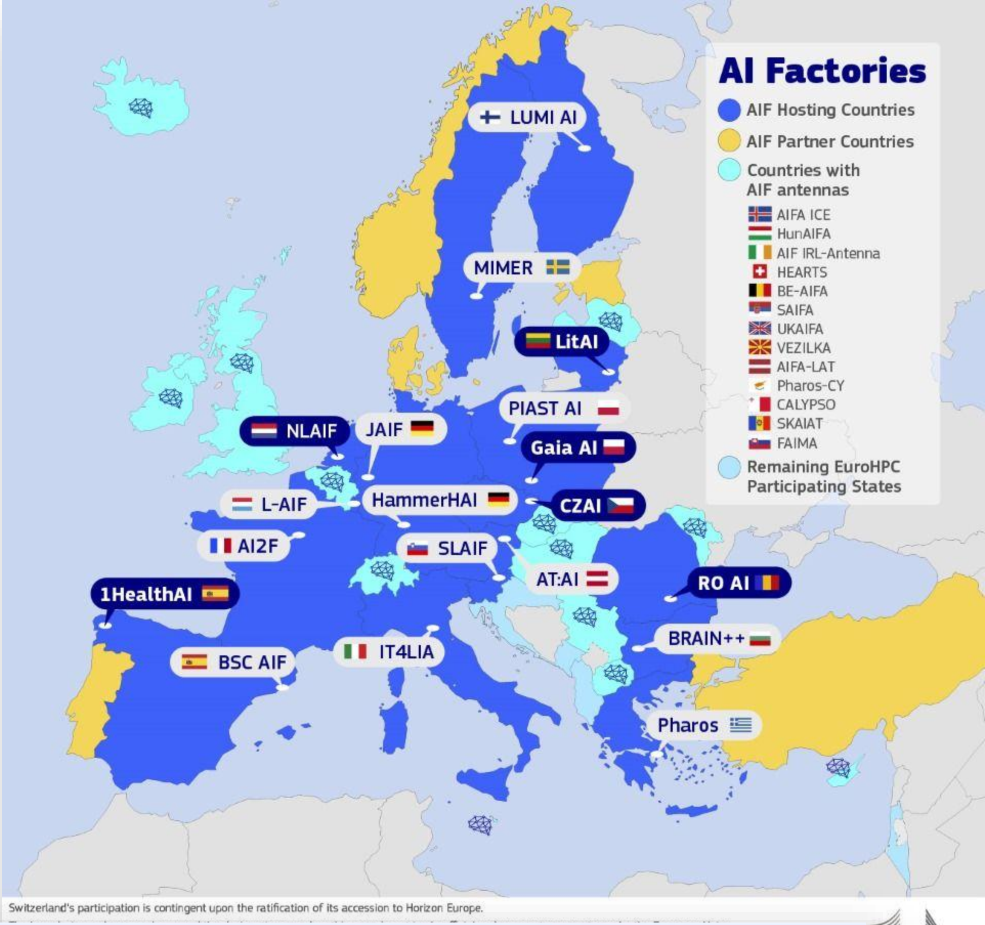
- Cray-1 (1976) → commodity clusters (1990s) → petascale (2008) → exascale (Frontier, 2022)
- GPUs and AI accelerators now dominate — paving the way for AI-first supercomputers: the EuroHPC AI Factories

TOP500 list: [http://top500.org](http://top500.org)

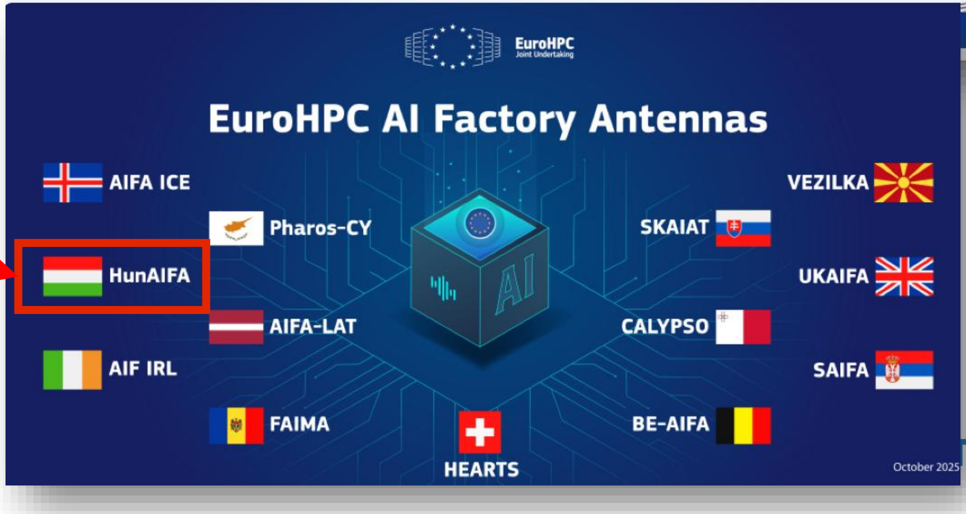


# AI Factories

- EU-backed network of **high-performance computing hubs** giving European startups, researchers, and industry sovereign access to AI supercomputing
- **19 AI Factories and 13 AI Factory Antennas** spanning the EU, anchored on AI-optimised supercomputers (LUMI, Leonardo, MareNostrum, and others)
- **3 AI access modes** tailored to users, with 40+ key industrial sectors covered across the network
- Provide **GPU clusters, data infrastructure, and expert support** to develop, train, and deploy trustworthy AI solutions
- Built around European SMEs, public-sector users, and strategic sectors
- Coordinated by **EuroHPC JU**



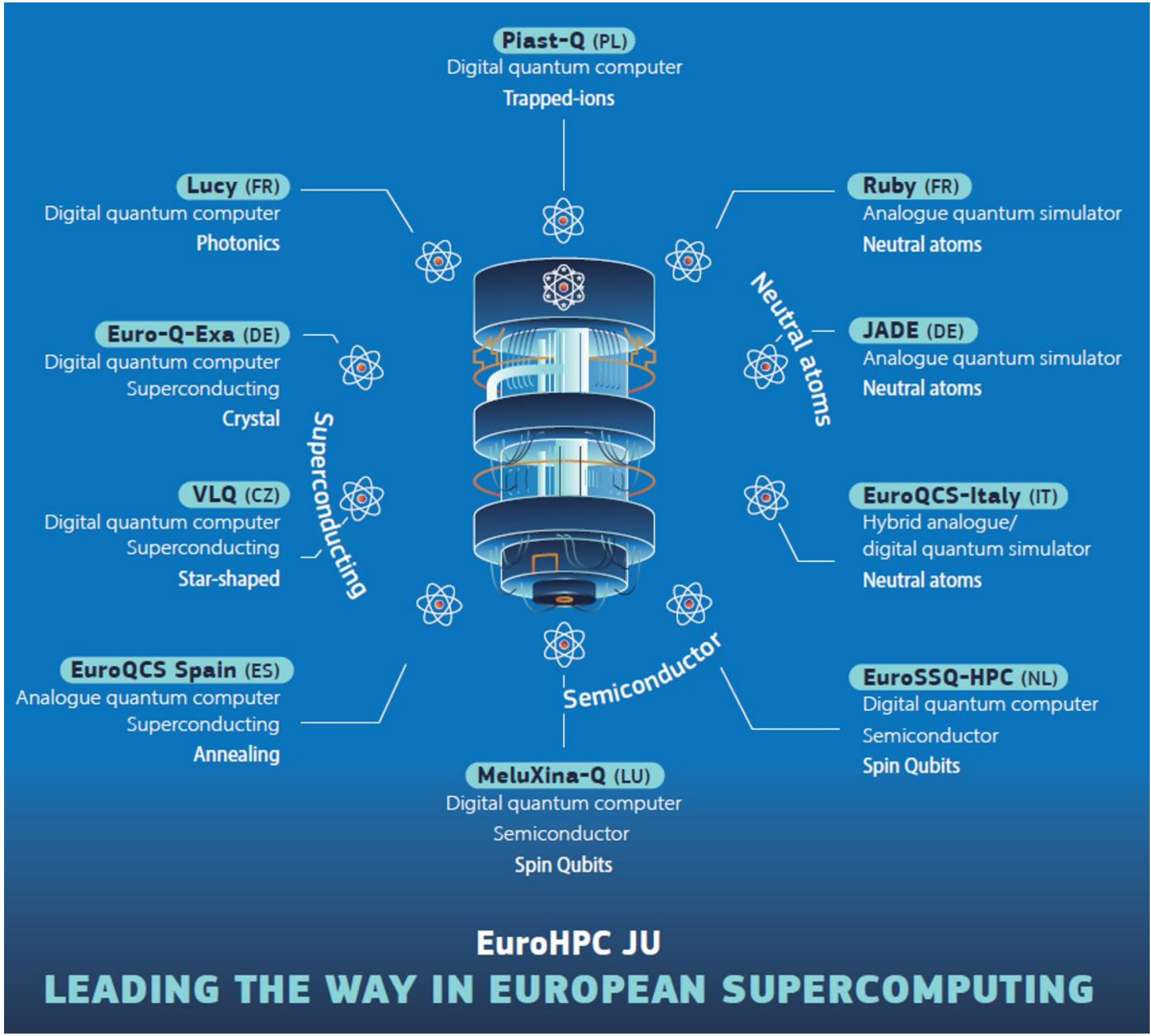
Switzerland's participation is contingent upon the ratification of its accession to Horizon Europe.



# EuroHPC: HPC – AI – QC Convergence 1.

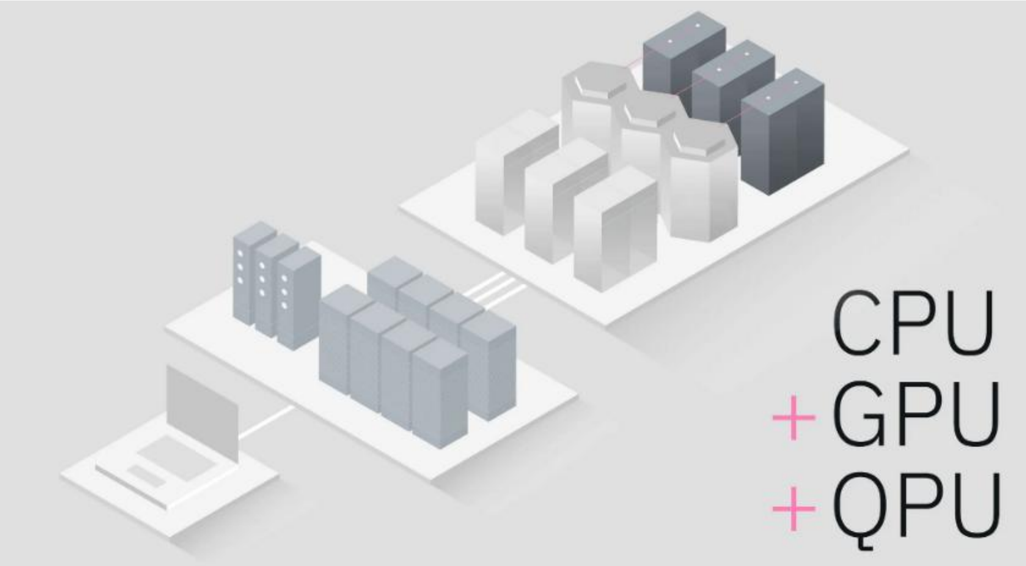
- 2022: **EuroHPC += QC**
- 2023: **EuroHPC += AI**
- 2026: **EuroHPC += Everything Quantum** (e.g., sensing, communication)

- Two closed calls for proposals: 8 + 2 = 10 quantum computers
- 120 million EUR
- European manufacturers
- 6 different qubit technologies
- 650 qubits in total
- 1 system is annealing; the others are gate based



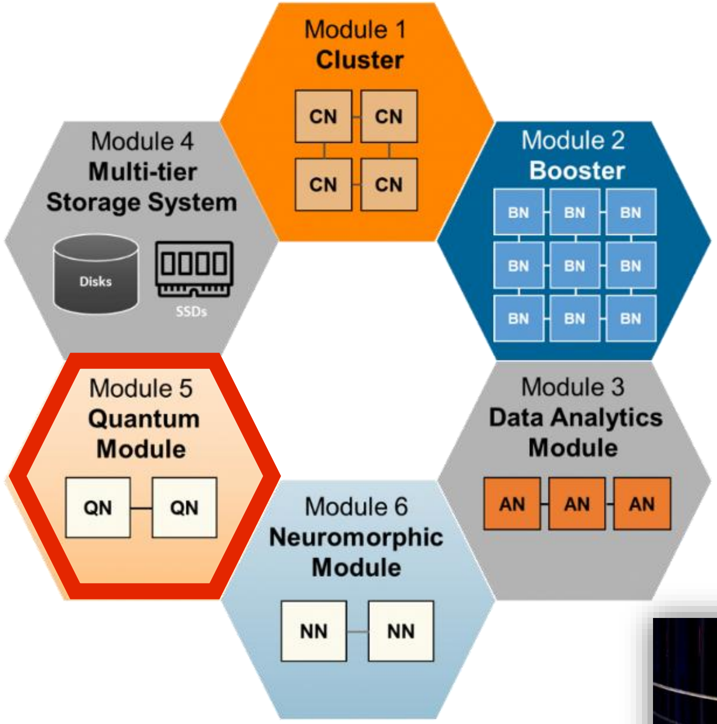
# Key Challenges of HPC – AI – QC Convergence 2.

- **Hardware architecture**: integrating CPU, GPU, and QPU resources
- **Software stack:** unified programming across classical and quantum
- **Maturing portfolio of quantum algorithms**
- **Hybrid algorithms** orchestrating HPC, AI, and quantum together
- Existing and emerging **enabling technologies**:
  - Qubernetes
  - Slurm quantum support (SPANK)
  - MQSS — Munich Quantum Software Stack
  - NVQLink — quantum-GPU interconnect
  - NVIDIA CUDA-Q
  - QBridge — HPC–quantum integration

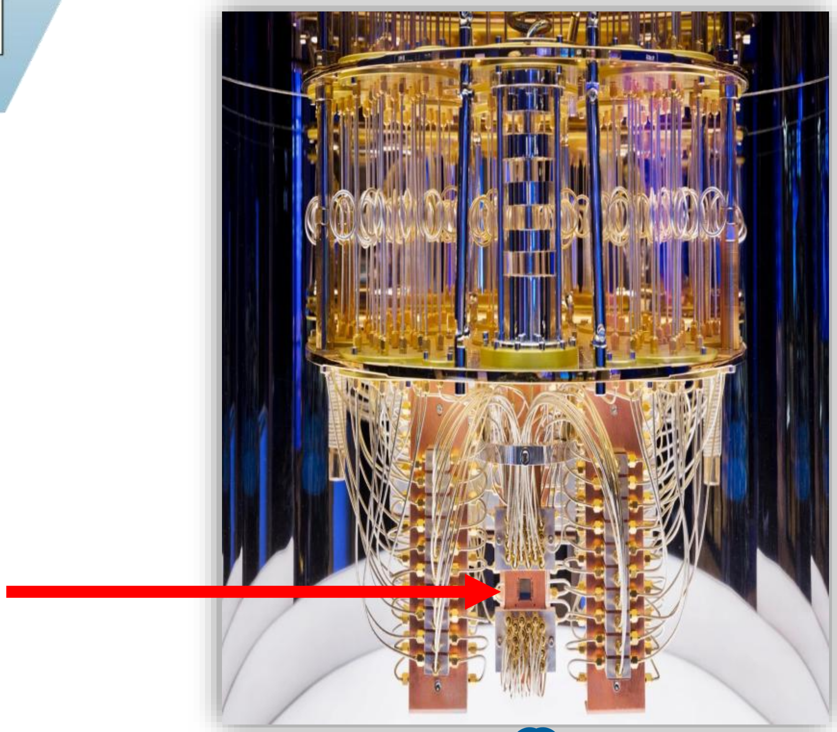


# EuroHPC: HPC – AI – QC Convergence 3.

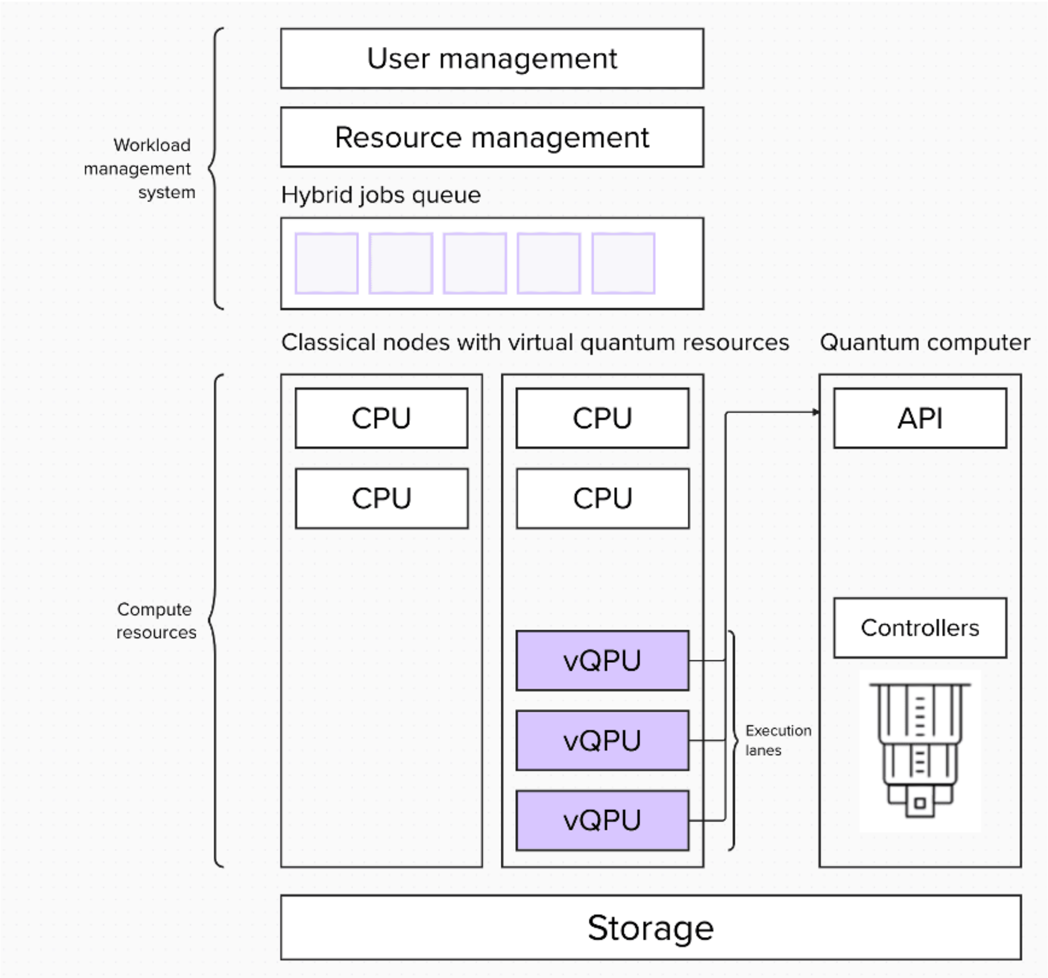
- Modular architecture **bringing classical HPC, AI, and quantum computing** into a single system
- Specialized modules for simulation, AI training, data analytics, and emerging quantum and neuromorphic workloads
- Shared storage and **high-speed interconnects** let workloads use the right resource at the right time
- **Enables hybrid pipelines**: HPC simulation feeding AI models, with quantum acceleration
- Energy **efficiency and sustainability are central design drivers** as performance scales
- **A blueprint for Europe's next generation of exascale and post-exascale systems**



Jülich Supercomputing Centre (JSC), ParTec



# Possible Architectures



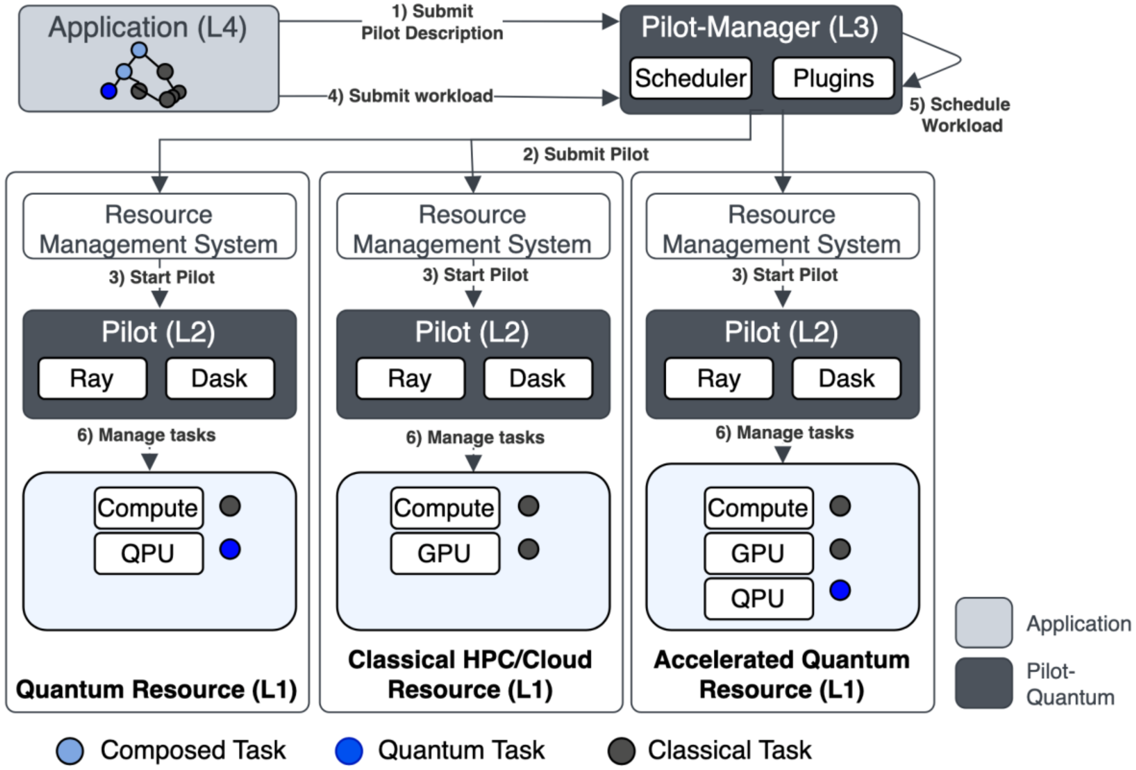
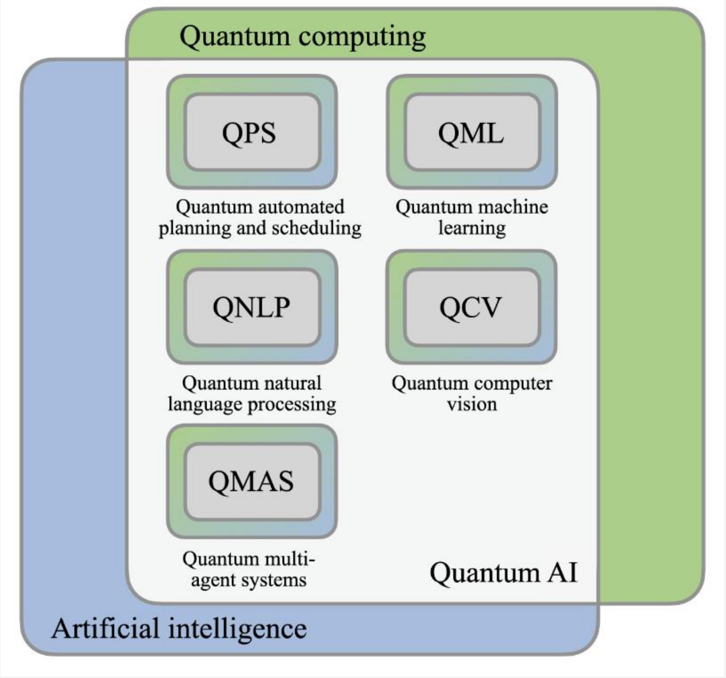


Fig. 2. **Pilot Quantum Architecture:** The system core is a *Pilot-Manager* that orchestrates and manages resources through pilots across both classical and quantum infrastructures, such as QPUs, GPUs, and CPUs. Pilots are responsible for reserving resources and managing task execution.

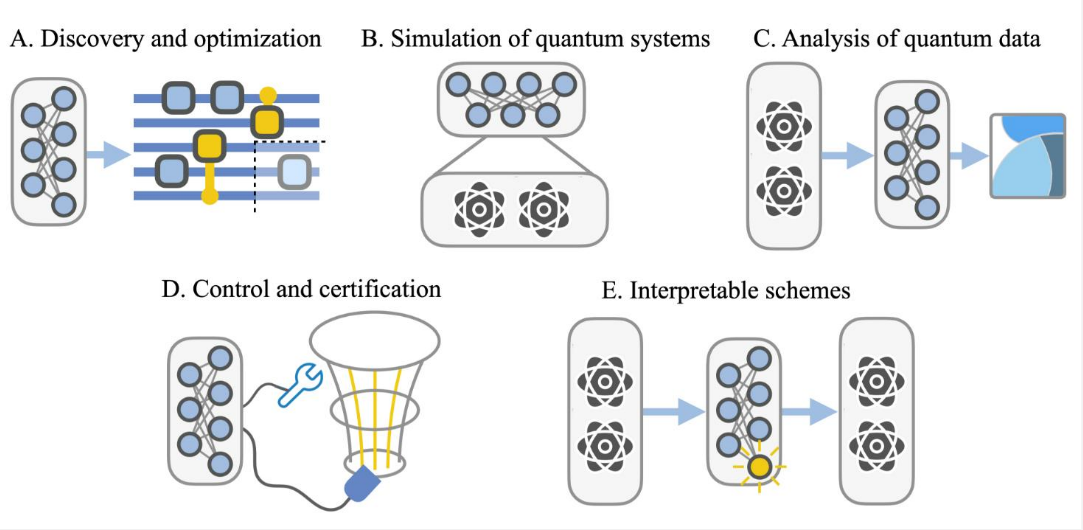
Mantha, Pradeep, Florian J. Kiwit, Nishant Saurabh, Shantenu Jha, and Andre Luckow. "Pilot-Quantum: A Quantum-HPC Middleware for Resource, Workload and Task Management." arXiv:2412.18519. Preprint, arXiv, May 28, 2025. https://doi.org/10.48550/arXiv.2412.18519.

# AI-for-Quantum or Quantum-for-AI?

**Quantum-for-AI:** *Utilizing quantum computing to accelerate AI tasks like optimization and data processing beyond classical computational limits.*



**AI-for-Quantum:** *Applying classical machine learning to optimize quantum hardware design, control, error correction, and numerical simulations.*

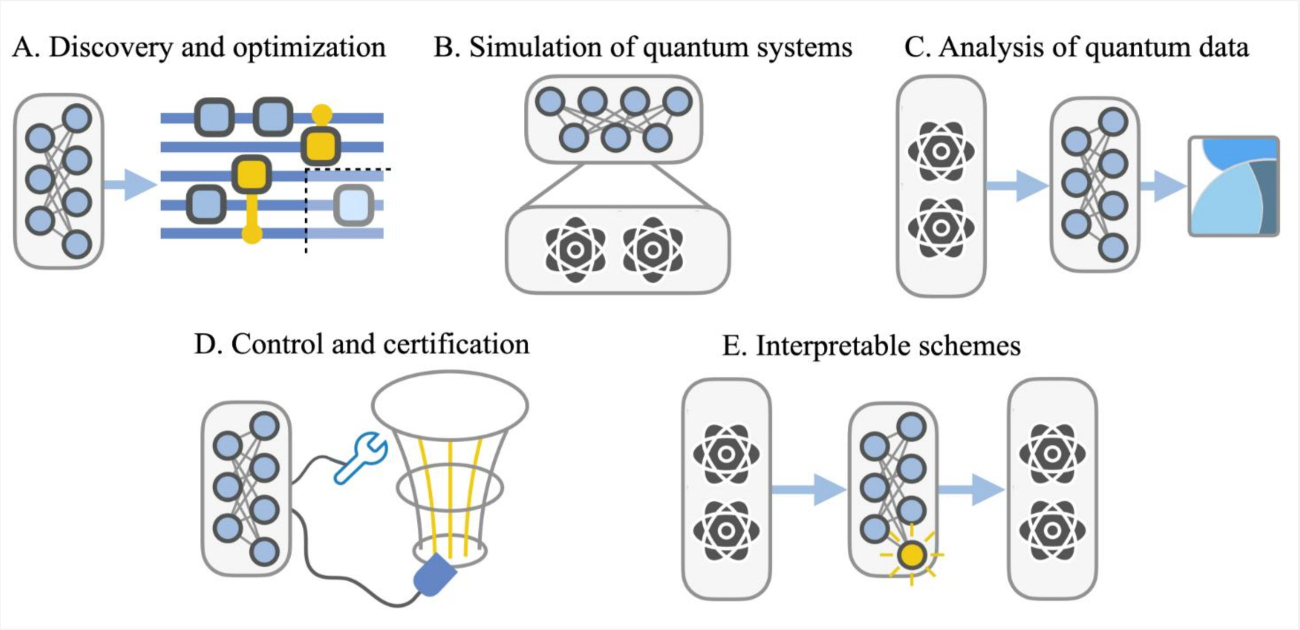


Acampora, Giovanni, Andris Ambainis, Natalia Ares, et al. “Quantum Computing and Artificial Intelligence: Status and Perspectives.” Version 3. Preprint, arXiv, 2025. https://doi.org/10.48550/ARXIV.2505.23860.

# AI-for-Quantum

**AI-for-Quantum:** *Applying classical machine learning to optimize quantum hardware design, control, error correction, and numerical simulations.*

- **Hardware & Algorithm Discovery:** Automates experimental design, circuit compilation, and the creation of new error-correction codes.
- **Automated Control & Calibration**: Applies reinforcement learning to precisely tune quantum devices and optimize real-time gate operations.
- **Analysis of Quantum Data**: Utilizes ML for quantum state and process tomography, extracting relevant physical features from high-dimensional or noisy experimental data, and performing quantum error mitigation through AI-driven post-processing.

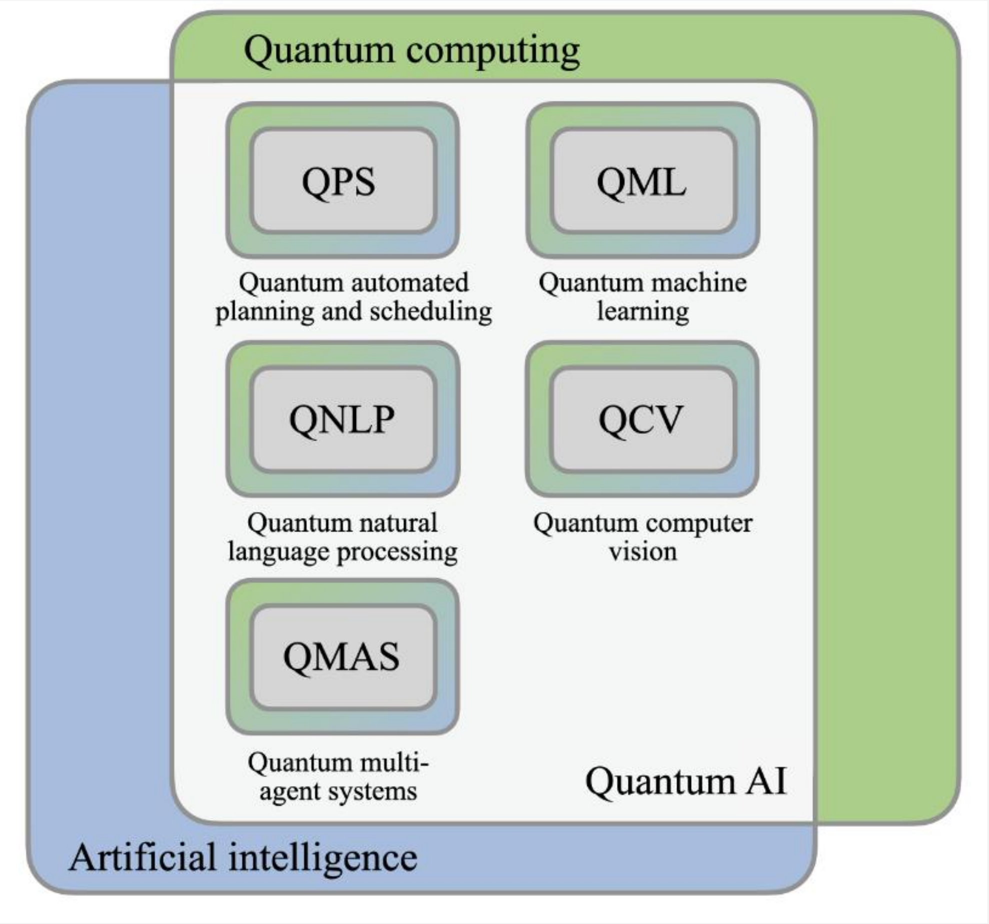


Acampora, Giovanni, Andris Ambainis, Natalia Ares, et al. "Quantum Computing and Artificial Intelligence: Status and Perspectives." Version 3. Preprint, arXiv, 2025. https://doi.org/10.48550/ARXIV.2505.23860.

# Quantum-for-AI (Quantum AI)

**Quantum-for-AI:** *Utilizing quantum computing to accelerate AI tasks like optimization and data processing beyond classical computational limits.*

- **Quantum Machine Learning (Supervised & Unsupervised**): For example, enhances classification, regression, and clustering tasks by using quantum algorithms to accelerate training.
- **Quantum Reasoning and NLP:** Uses quantum parallelism and entanglement to speed up logical inference and represent complex semantic relationships in natural language processing
- **Quantum Multi-Agent Systems (QMAS):** Improves coordination and resource allocation among multiple autonomous agents through quantum-supported negotiation and coalition protocols
- **Quantum Optimization and Planning**: Addresses combinatorial problems and industrial scheduling by exploring massive search spaces more effectively than classical solvers.

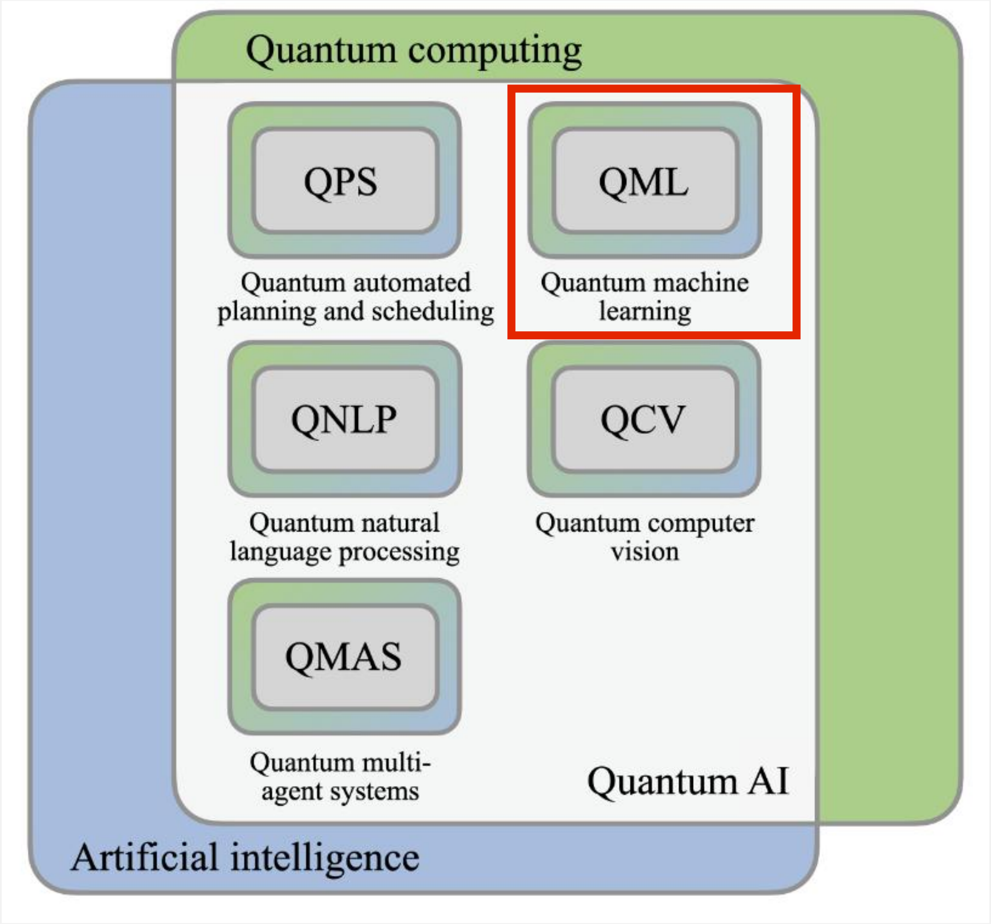


Acampora, Giovanni, Andris Ambainis, Natalia Ares, et al. “Quantum Computing and Artificial Intelligence: Status and Perspectives.” Version 3. Preprint, arXiv, 2025. https://doi.org/10.48550/ARXIV.2505.23860.

# Quantum Machine Learning

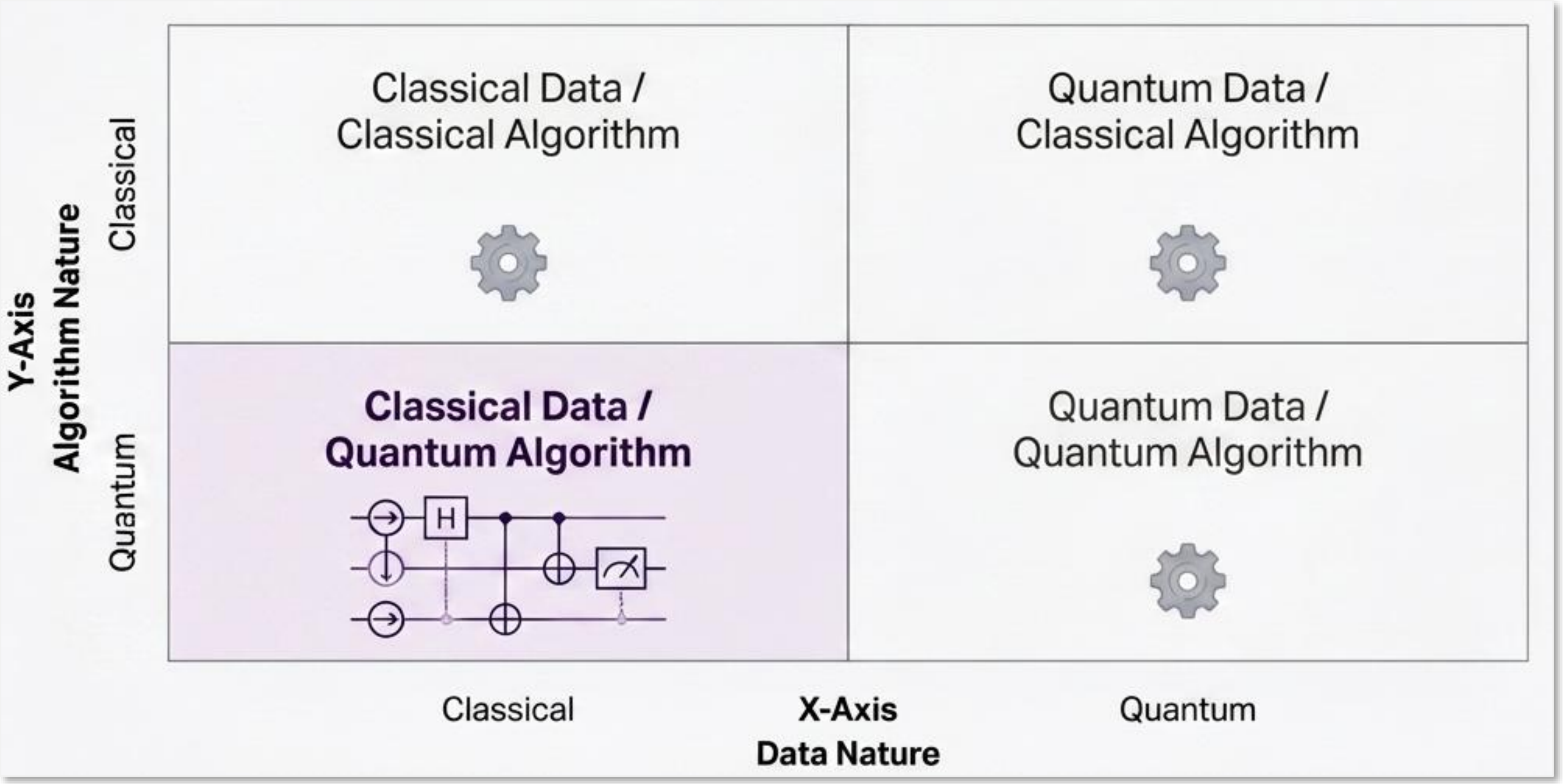
Combine concepts from **machine learning** and **quantum computing**.

1. *Learn patterns and relations from given training data.*
2. *Use quantum mechanical resources to solve computational problems - potentially more efficiently or accurately than classical computers.*



Acampora, Giovanni, Andris Ambainis, Natalia Ares, et al. “Quantum Computing and Artificial Intelligence: Status and Perspectives.” Version 3. Preprint, arXiv, 2025. https://doi.org/10.48550/ARXIV.2505.23860.

# Quantum Machine Learning



Y-Axis
Algorithm Nature

Classical

Quantum

Classical Data / Classical Algorithm

Quantum Data / Classical Algorithm

**Classical Data / Quantum Algorithm**

Quantum Data / Quantum Algorithm

Classical

X-Axis
Data Nature

Quantum

**Quantum Support Vector Machines (SVMs)**

**Quantum Neural Networks (QNN)**

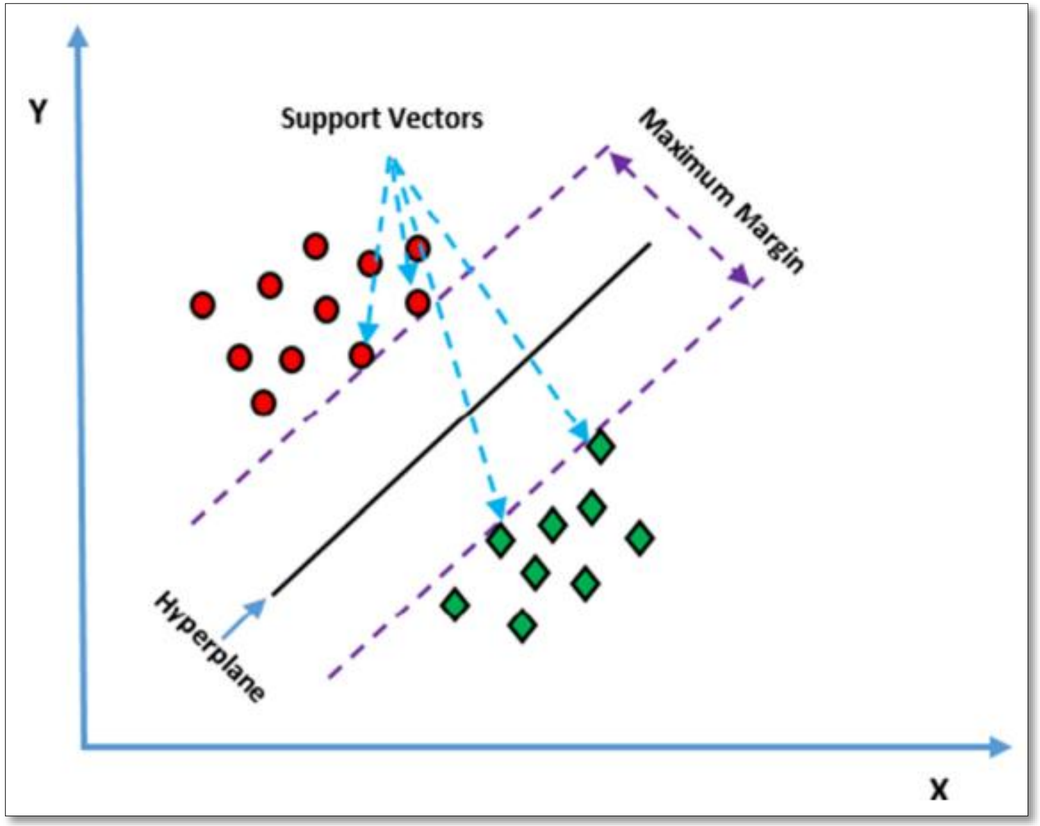
**Hybrid Architectures**

**Quantum Generative Adversarial Network (QGAN)**

…

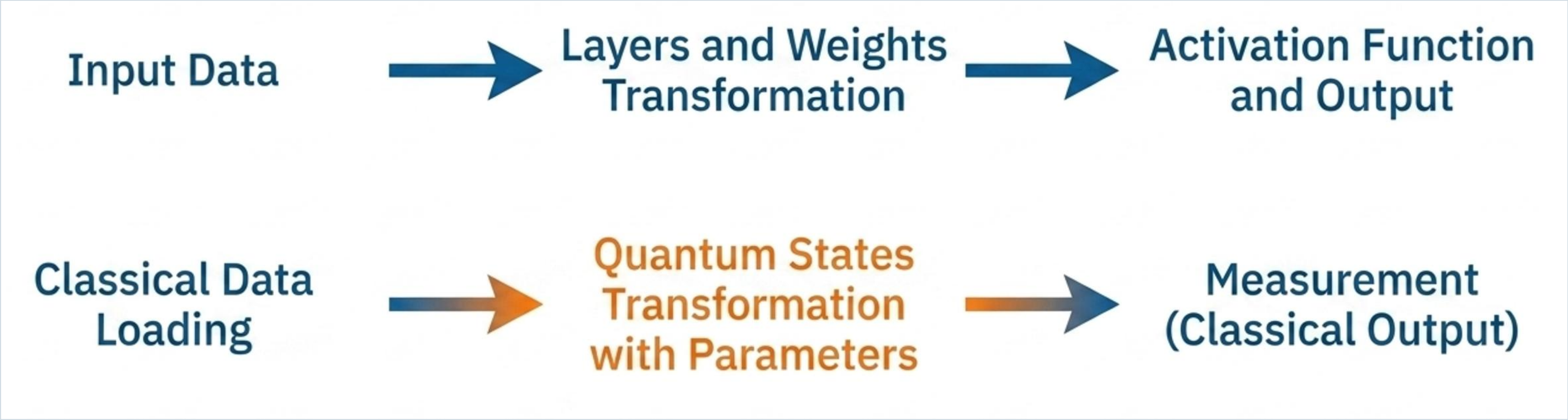
# Quantum Support Vector Machines

- **Supervised learning algorithm** used primarily for classification and regression tasks.
- **Finds the optimal hyperplane** that best separates data points of different classes.
- **Maximizes the margin** between support vectors — the points closest to the decision boundary.
- Classical data is mapped into quantum states via a quantum circuit, producing what is known as a **quantum kernel function**.
- **QSVM is a hybrid algorithm**, not purely quantum:
  - The quantum computer evaluates the inner product of the kernel function.
  - Mathematical optimization still runs on a classical computer.



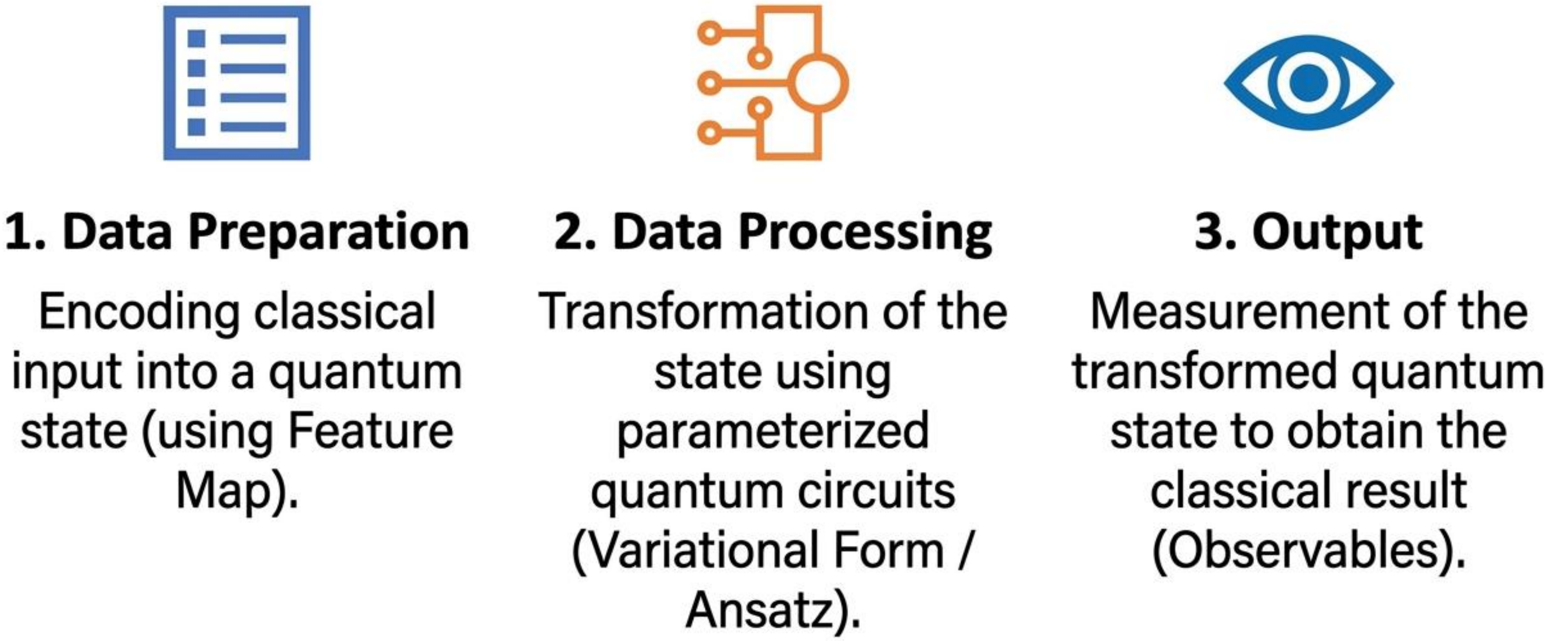
GPU Day 2026 — May 28-29, 2026, Budapest

# Quantum Neural Networks (QNN)



QNNs are CQ models — they have purely classical input and output with intermediate quantum processing.

# The Three Pillars of QNN Architecture



**1. Data Preparation**

Encoding classical input into a quantum state (using Feature Map).

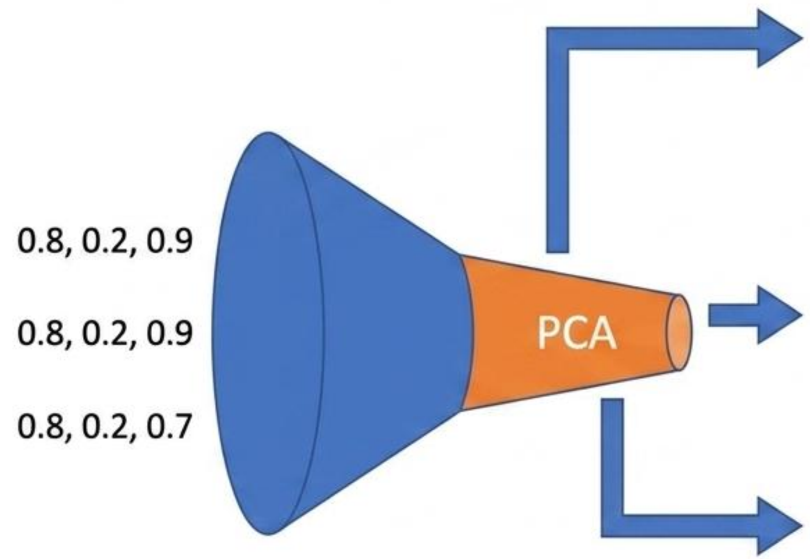
**2. Data Processing**

Transformation of the state using parameterized quantum circuits (Variational Form / Ansatz).

**3. Output**

Measurement of the transformed quantum state to obtain the classical result (Observables).

# QNN: Data Loading (Step 1)



## Core Concept

Quantum computers work with **quantum states**, not numbers. Classical data is encoded into the state via a specific **quantum circuit (Feature Map)** (e.g., **ZZFeatureMap, AmplitudeEmbedding**).

## Data Normalization

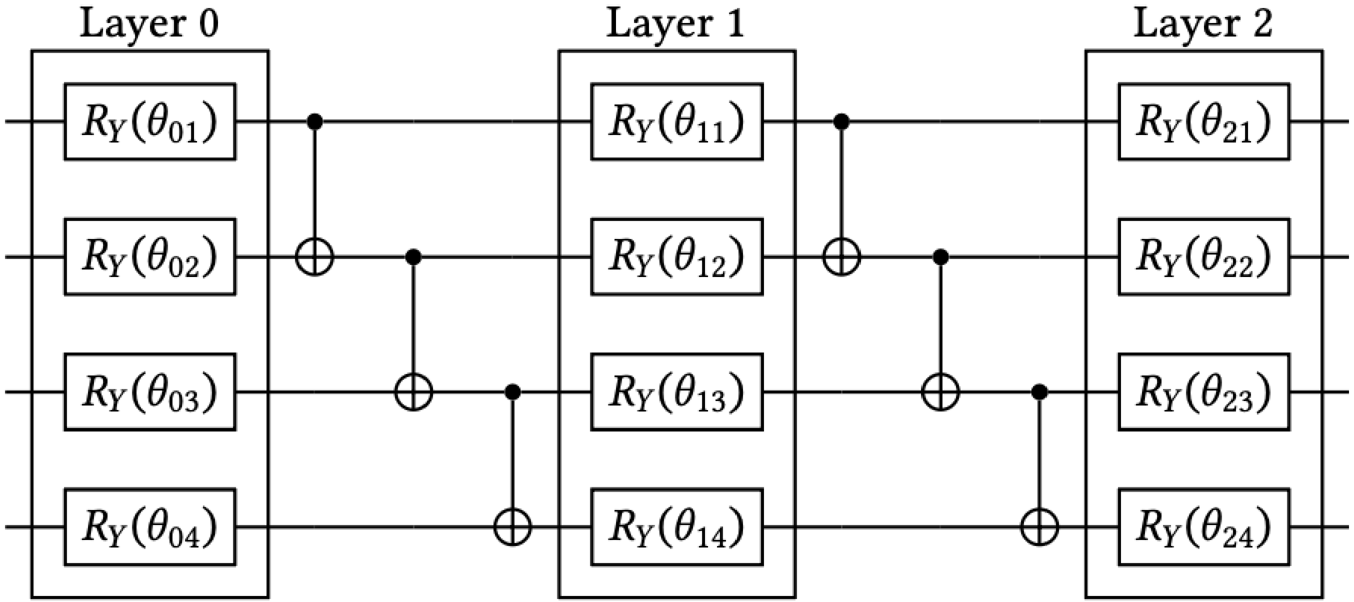
Essential to **normalize** data (e.g., using **MaxAbsScaler** between 0 and 1).

## Dimensionality Reduction

Current quantum hardware has limited qubits. Reducing input dimensions (e.g., with PCA from 30 variables to 4) is a critical step before data encoding.

# QNN: The Processing Engine (Variational Forms, Step 2)

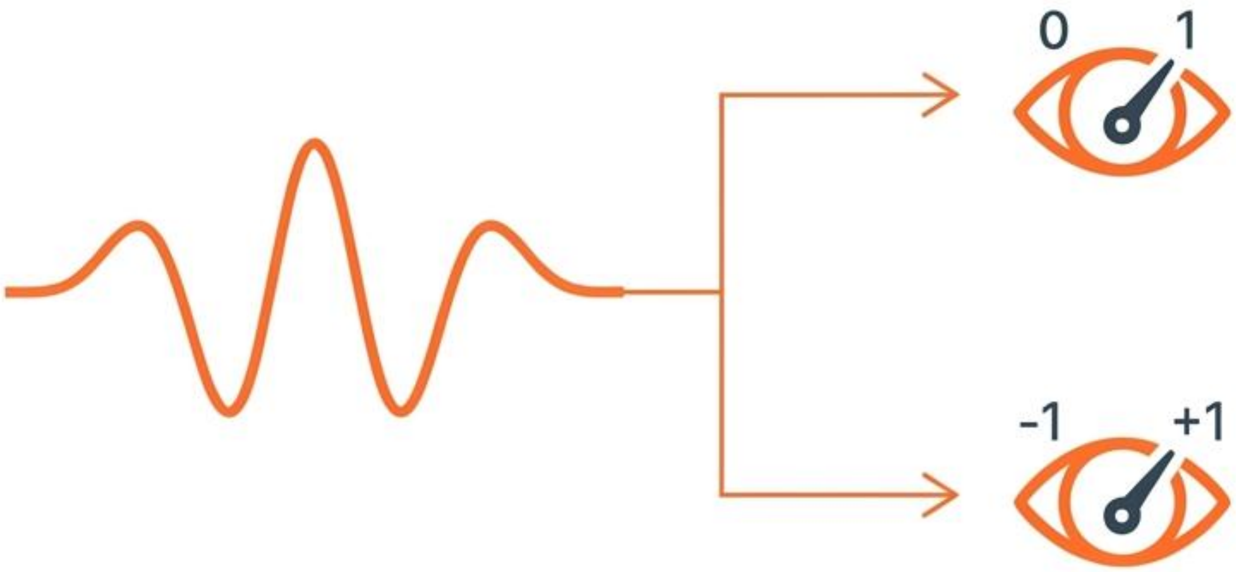
Instead of classical weights, we apply quantum circuits (**ansatz**) that depend on trainable parameters ($\theta$). The $\theta$ parameters are updated as the model is trained.



*Two-local variational form on four qubits and two repetitions.*

**Rotation gates**: adjust the state of individual qubits using independent parameters.

# QNN: Extracting the Classical Output (Step 3)



**Strategy A: Computational Basis Expected Value**

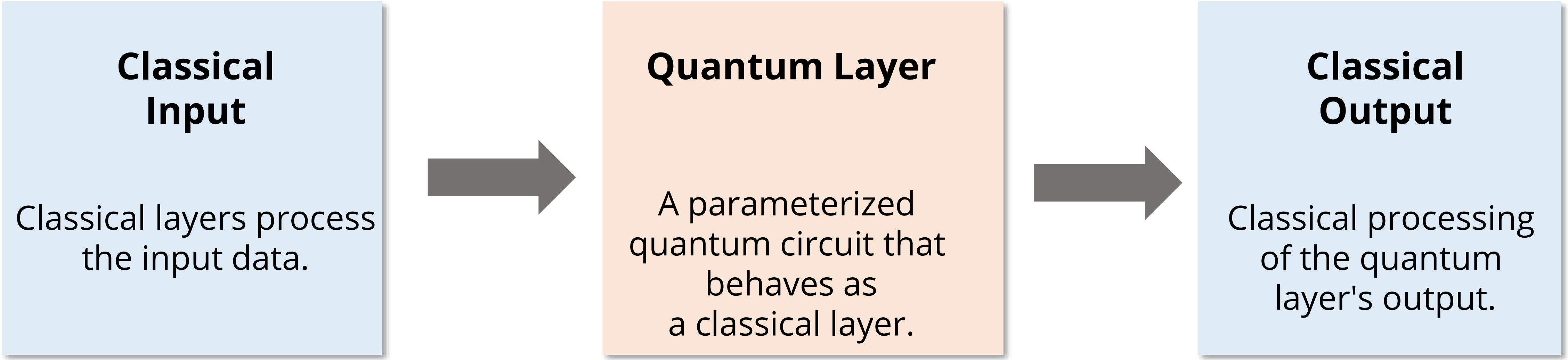
The simplest approach: calculating the measurement probability (e.g., the result is 0 or 1 on qubit 1). A common choice for binary classifiers.

**Strategy B: Parity Measurement**

Using more complex operators (e.g., Z tensor product Z). Simultaneously measures all qubits, the result is +1 (if we measure an even number of 0s) or -1 (if odd).

We need to extract a classical numerical value from the transformed quantum state by measuring an observable quantity (Hermitian operator).

# Hybrid Neural Networks



**Classical Input**

Classical layers process the input data.

**Quantum Layer**

A parameterized quantum circuit that behaves as a classical layer.

**Classical Output**

Classical processing of the quantum layer's output.

The full hybrid architecture is trained end-to-end as a single unit.

# Quantum Machine Learning – Speedup

1. Rigorous analysis of a quantum speedup **for supervised learning** [1].
2. Unsupervised case of **training generative models, quantum speedups** for carefully crafted learning tasks have likewise been established [2,3].

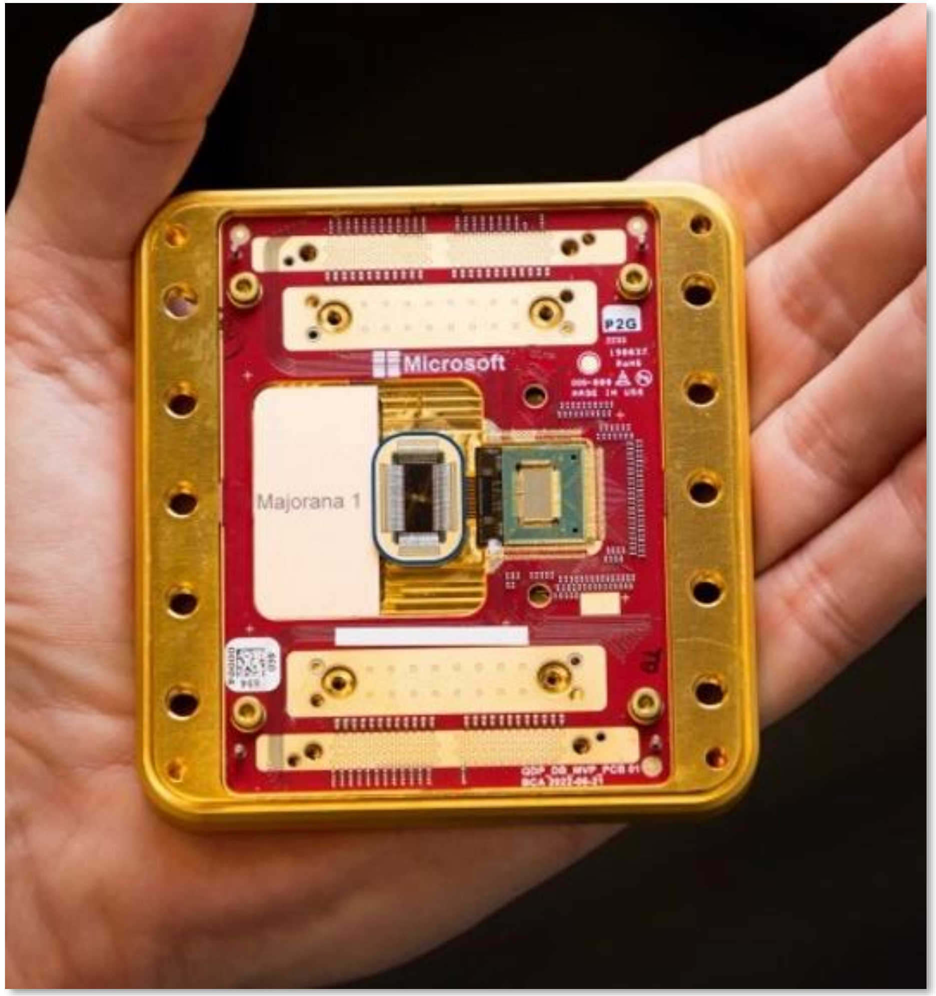
[1] Liu, Y., Arunachalam, S., Temme, K.: A rigorous and robust quantum speed-up in supervised machine learning (2020). arXiv preprint arXiv:2010.02174

[2] Gao, X., Zhang, Z.Y., Duan, L.M.: A quantum machine learning algorithm based on generative models. Sci. Adv. 4(12), eaat9004 (2018)

[3] Sweke, R., Seifert, J.P., Hangleiter, D., Eisert, J.: On the quantum versus classical learnability of discrete distributions. Quantum 5, 417 (2021)

# Outlook of Quantum Machine Learning – I.

1. **Still very early days.**
   A young field — researchers are still discovering what works and what doesn't.
2. **The big question: can it actually do better?**
   The aim is to show that quantum methods can outperform today's already powerful classical AI — but no clear practical advantage has been demonstrated yet.
3. **Machine learning is a particularly hard domain.**
   Unlike well-defined problems (e.g. shortest path), ML builds models that recognize patterns in messy real-world data — which makes any quantum advantage harder to prove.
4. **Strong open-source tools are accelerating progress.**
   Platforms like PennyLane and TensorFlow Quantum make it easy to prototype and simulate new ideas.



# Outlook of Quantum Machine Learning – II.

## 5. The focus is shifting from "faster" to "smarter."

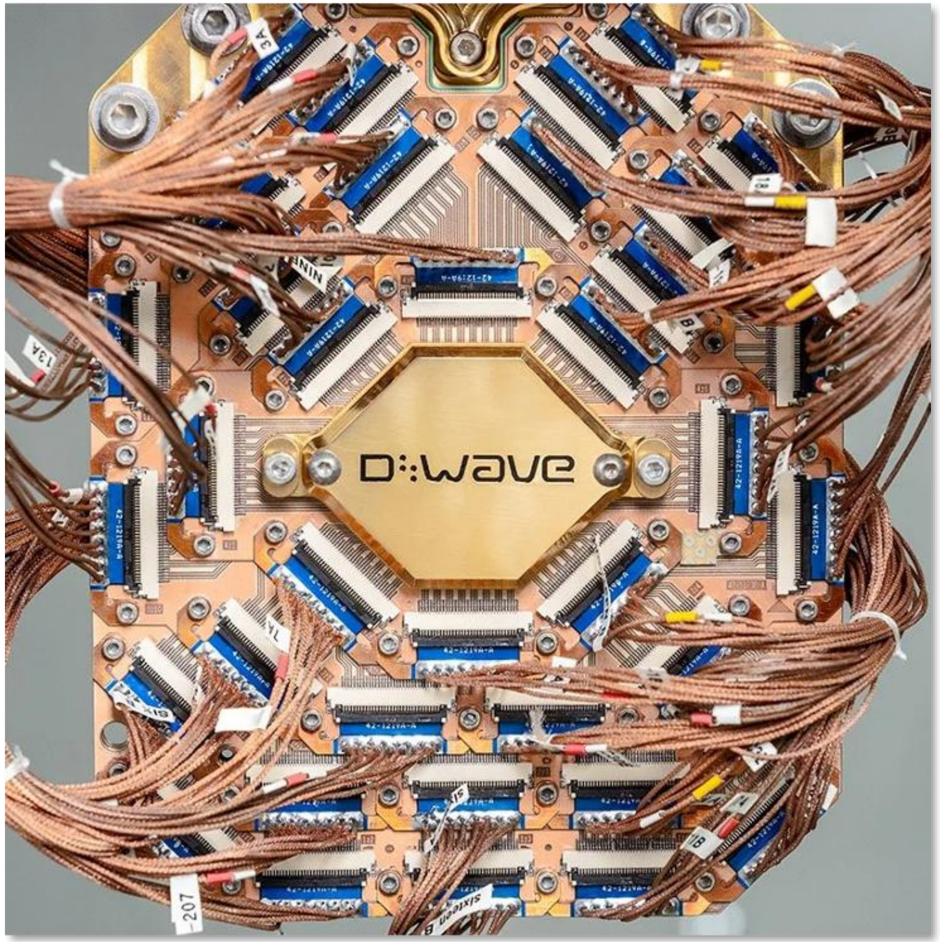
- Early research concentrated on speeding up existing algorithms.
- The question now is *what quantum models can actually learn, and how.*

## 6. The best model architecture is still unknown.

- Classical AI has proven structures.
- Quantum ML is *still searching for the foundational model type* that fits learning tasks naturally.

## 7. The impact will likely be in specialized problems.

Quantum computers won't replace classical ones — they'll be useful in *specific domains* such as complex network analysis, medical imaging, and physics simulation.



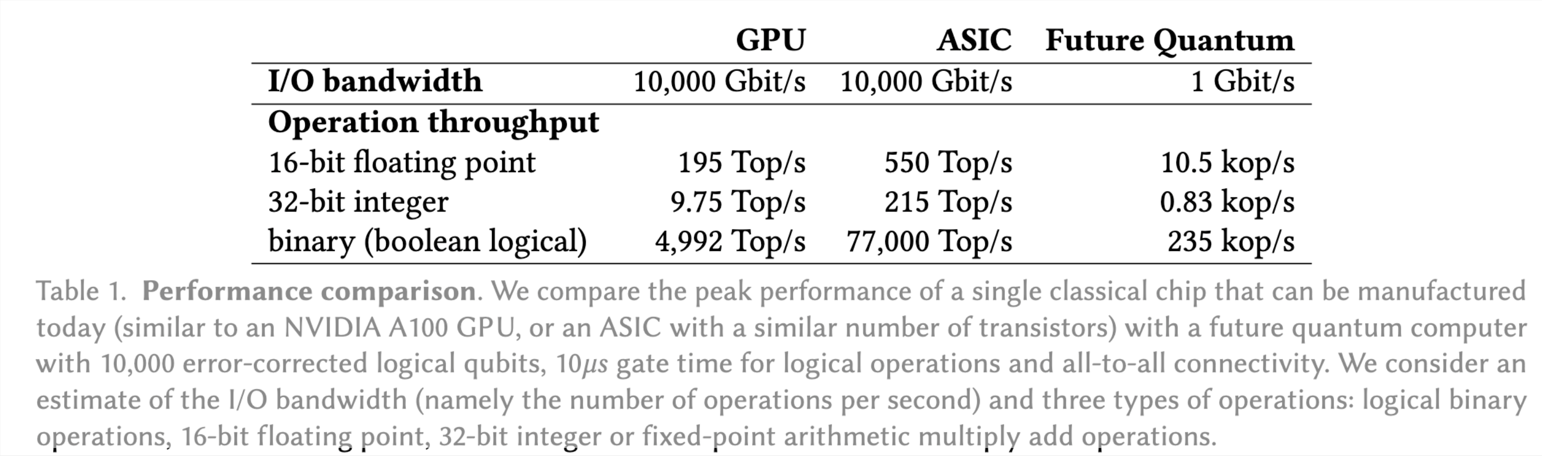
# Outlook of Quantum Machine Learning – III.

8. **There may not be a single "killer app."**
   Rather than one breakthrough application, quantum ML is more likely to serve as a specialized accelerator within targeted domains.
9. **Business value is still uncertain.**
   Classical AI is already extremely powerful, so it's not yet clear when or how quantum ML will deliver a real commercial advantage.
10. **At heart, it's a deep scientific question.**
    Quantum ML asks what "learning" really means when information is processed under the laws of quantum physics — a fascinating research frontier in its own right.

# Classical Throughput vs. Quantum Gates

*A fundamental performance gap exists between classical parallel processors and projected quantum hardware, necessitating a clear understanding of where each excels.*

- The continued evolution of classical heuristics and accelerated simulation means the bar for "advantage" is a rapidly moving target.
- Classical chips operate with cycle times around `0.7ns`, while logical gate times for future error-corrected quantum computers are optimistically estimated at `10μs`.
- Because of this gap, **classical systems are the engines for high-throughput, data-intensive tasks.**
- Quantum processors are fundamentally better suited for "big compute" on small data sets rather than high-volume data processing.
- Quantum Random Access Memory (qRAM) is an essential but currently under-resourced enabling technology for data-intensive quantum machine learning.
- Practical utility requires that the quantum computer be "verifiable," so that its output can be efficiently checked by classical or other quantum systems.



| | GPU | ASIC | Future Quantum |
|---|---|---|---|
| **I/O bandwidth** | 10,000 Gbit/s | 10,000 Gbit/s | 1 Gbit/s |
| **Operation throughput** | | | |
| 16-bit floating point | 195 Top/s | 550 Top/s | 10.5 kop/s |
| 32-bit integer | 9.75 Top/s | 215 Top/s | 0.83 kop/s |
| binary (boolean logical) | 4,992 Top/s | 77,000 Top/s | 235 kop/s |

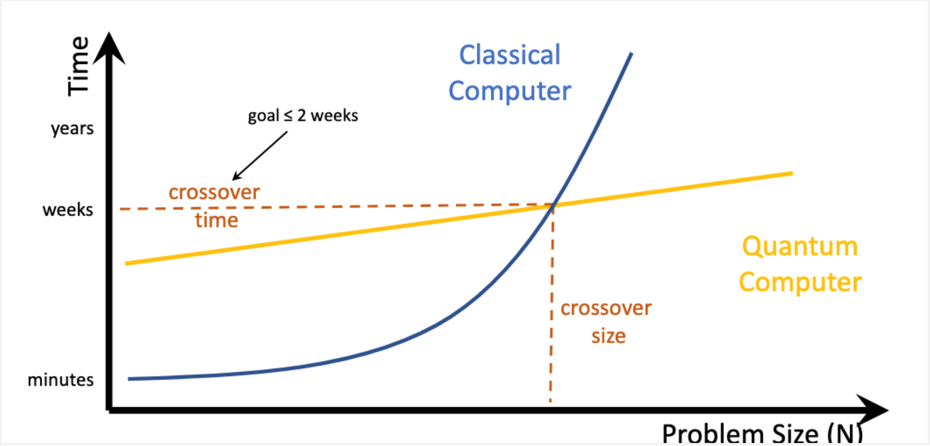
Table 1. **Performance comparison**. We compare the peak performance of a single classical chip that can be manufactured today (similar to an NVIDIA A100 GPU, or an ASIC with a similar number of transistors) with a future quantum computer with 10,000 error-corrected logical qubits, $10\mu s$ gate time for logical operations and all-to-all connectivity. We consider an estimate of the I/O bandwidth (namely the number of operations per second) and three types of operations: logical binary operations, 16-bit floating point, 32-bit integer or fixed-point arithmetic multiply add operations.

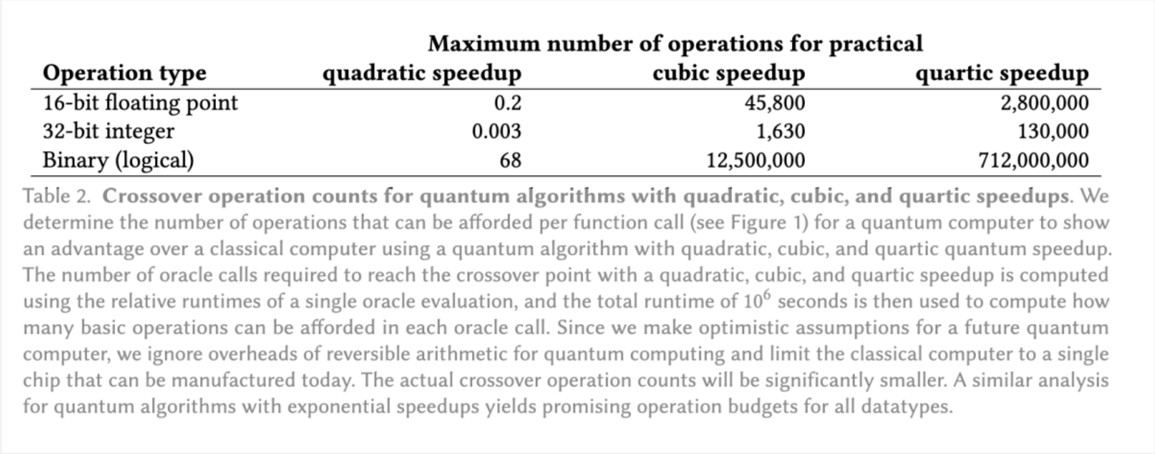
Hoefler, Torsten, Thomas Haener, and Matthias Troyer. “Disentangling Hype from Practicality: On Realistically Achieving Quantum Advantage.” arXiv:2307.00523. Preprint, arXiv, July 2, 2023. https://doi.org/10.48550/arXiv.2307.00523.

# Defining Practical Advantage: Beyond Quadratic Speedups

*To overcome the massive throughput lead of modern parallel hardware, quantum algorithms must provide super-quadratic speedups to be commercially viable.*

- Quadratic speedups (e.g., Grover's algorithm) are generally insufficient to achieve a practical crossover point within reasonable timeframes (e.g., two weeks).
- A quantum algorithm providing only quadratic speedup would require a problem size so large that even one floating-point operation would take months to show an advantage.
- Practical quantum advantage likely requires **super-quadratic (cubic, quartic) or exponential speedups**.
- Identifying "data-hard" problem instances that are classically intractable on average is a critical bottleneck in current research.
- Asymptotic scaling is less important for industrial utility than performance on specific, finite problem instances, such as chemical active spaces.





| Operation type | Maximum number of operations for practical quadratic speedup | cubic speedup | quartic speedup |
|---|---|---|---|
| 16-bit floating point | 0.2 | 45,800 | 2,800,000 |
| 32-bit integer | 0.003 | 1,630 | 130,000 |
| Binary (logical) | 68 | 12,500,000 | 712,000,000 |

Table 2. **Crossover operation counts for quantum algorithms with quadratic, cubic, and quartic speedups.** We determine the number of operations that can be afforded per function call (see Figure 1) for a quantum computer to show an advantage over a classical computer using a quantum algorithm with quadratic, cubic, and quartic quantum speedup. The number of oracle calls required to reach the crossover point with a quadratic, cubic, and quartic speedup is computed using the relative runtimes of a single oracle evaluation, and the total runtime of $10^6$ seconds is then used to compute how many basic operations can be afforded in each oracle call. Since we make optimistic assumptions for a future quantum computer, we ignore overheads of reversible arithmetic for quantum computing and limit the classical computer to a single chip that can be manufactured today. The actual crossover operation counts will be significantly smaller. A similar analysis for quantum algorithms with exponential speedups yields promising operation budgets for all datatypes.

Hoefler, Torsten, Thomas Haener, and Matthias Troyer. "Disentangling Hype from Practicality: On Realistically Achieving Quantum Advantage." arXiv:2307.00523. Preprint, arXiv, July 2, 2023. https://doi.org/10.48550/arXiv.2307.00523.

# Summary

1. **HPC: AI, and quantum are converging into one platform.**
   - EuroHPC and the AI Factories network are positioning Europe to offer CPUs, GPUs, and QPUs through a single sovereign infrastructure.
2. **Quantum and AI reinforce each other in two directions.**
   - **Quantum-for-AI** uses quantum computing to accelerate AI tasks such as optimization and data processing beyond classical computational limits.
   - **AI-for-Quantum** applies classical machine learning to optimize quantum hardware design, control, error correction, and numerical simulations.
3. **Hybrid architectures are the practical path forward.**
   - For example, classical layers handle data preparation and output, while parameterized quantum circuits act as trainable components inside the same end-to-end model.
4. **Convergence still has open challenges.**
   - Unified hardware integration, a common software stack across classical and quantum, and orchestration of hybrid algorithms remain active areas of work.
5. **Value will come from specialized, domain-specific use cases.**
   - Quantum will complement classical computing in targeted domains, not replace it.

# References

1. M. A. Nielsen and I. L. Chuang. Quantum Computation and Quantum Information. Cambridge University Press, 2010.
2. S. Aaronson and A. Arkhipov. The computational complexity of linear optics. 2011.
3. D. Shepherd and M. J. Bremner. Temporally unstructured quantum computation. 2009.
4. V. Havlíček, et al. Supervised learning with quantum-enhanced feature spaces. 2019.
5. *S. Aaronson. Read the fine print. 2015.*
6. J. McClean, et al. Barren plateaus in quantum neural network training landscapes. 2018.
7. Schuld, Maria, and Francesco Petruccione. *Machine Learning with Quantum Computers*. Quantum Science and Technology. Springer International Publishing, 2021. https://doi.org/10.1007/978-3-030-83098-4.
8. Alvarez, Elias Fernandez-Combarro, and Samuel Gonzalez Castillo. *PRACTICAL GUIDE TO QUANTUM MACHINE LEARNING AND QUANTUM OPTIMISATION Hands-on Approach to Modern Quantum Algorithms*. First edition. PACKT PUBLISHING LIMITED, 2023.
9. Hoefler, Torsten, Thomas Haener, and Matthias Troyer. “Disentangling Hype from Practicality: On Realistically Achieving Quantum Advantage.” arXiv:2307.00523. Preprint, arXiv, July 2, 2023. https://doi.org/10.48550/arXiv.2307.00523.
10. Babbush, Ryan, Robbie King, Sergio Boixo, et al. “The Grand Challenge of Quantum Applications.” arXiv:2511.09124. Preprint, arXiv, December 4, 2025. https://doi.org/10.48550/arXiv.2511.09124.

NTT DATA

# Thank You!