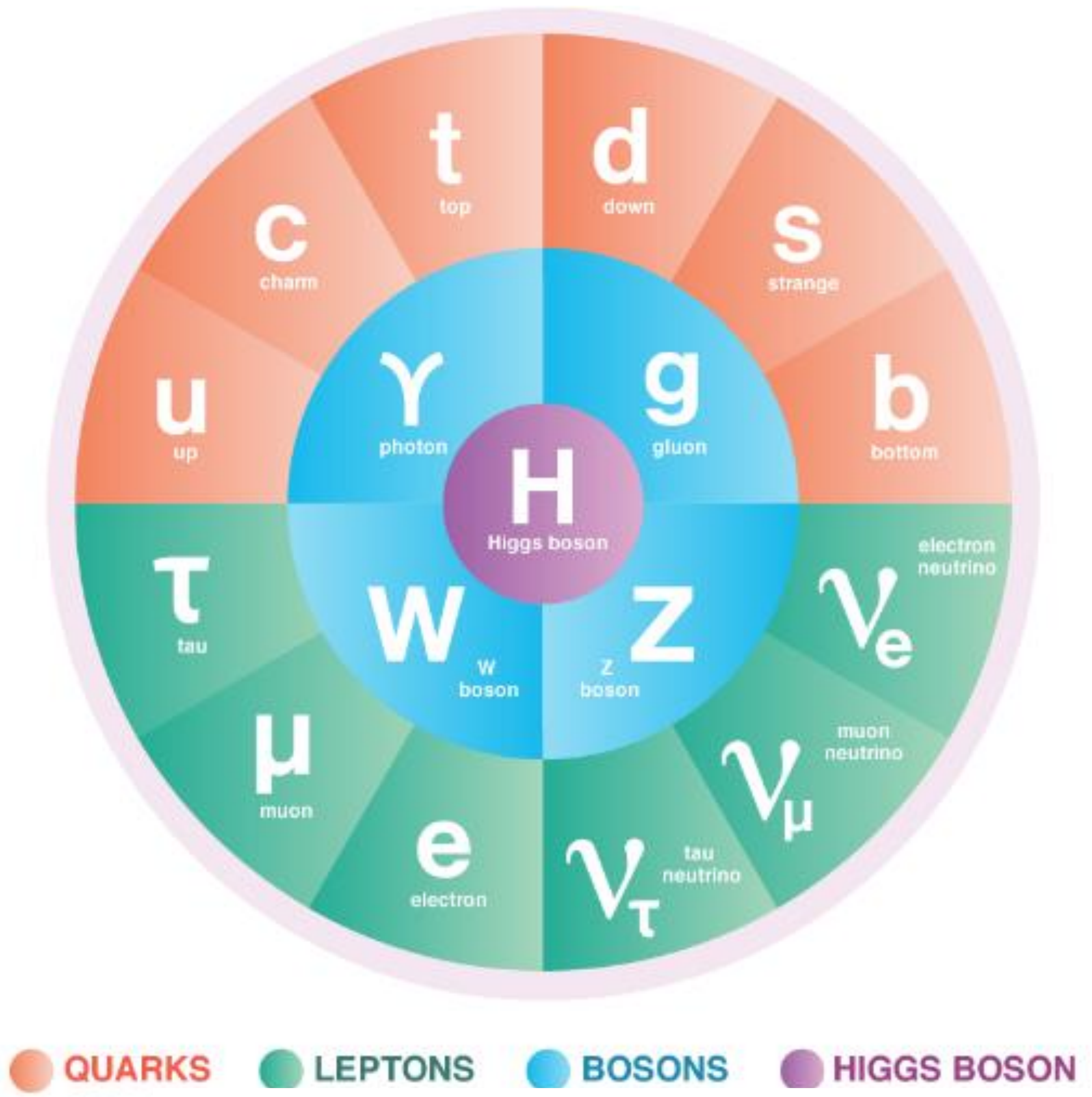
IBM Quantum Roadmap

Vincent R. Pascuzzi
Quantum Algorithm Engineer
IBM Quantum
T.J. Watson Research Center
Yorktown Heights, NY, USA

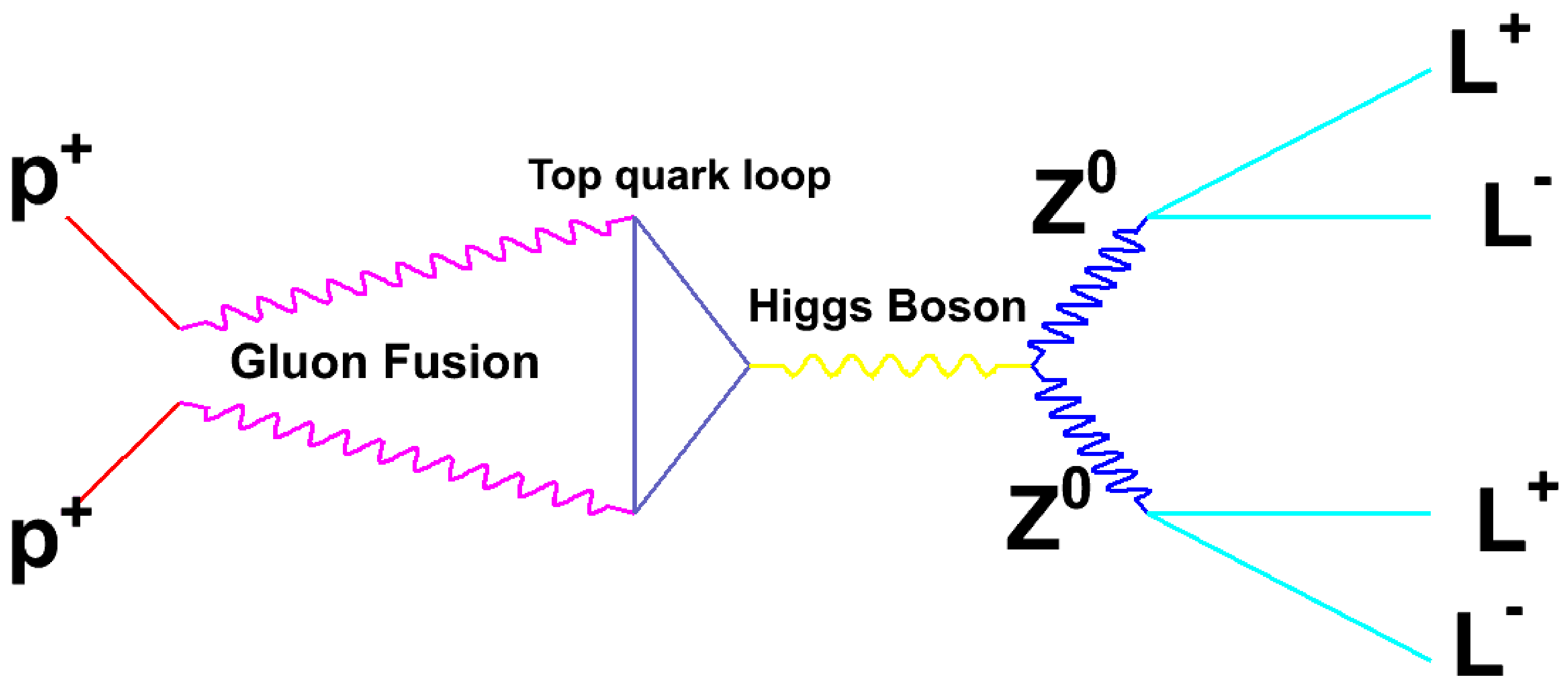
Quantum Technologies for High Energy Physics
CERN
Meyrin, Switzerland
23 January 2025

IBM

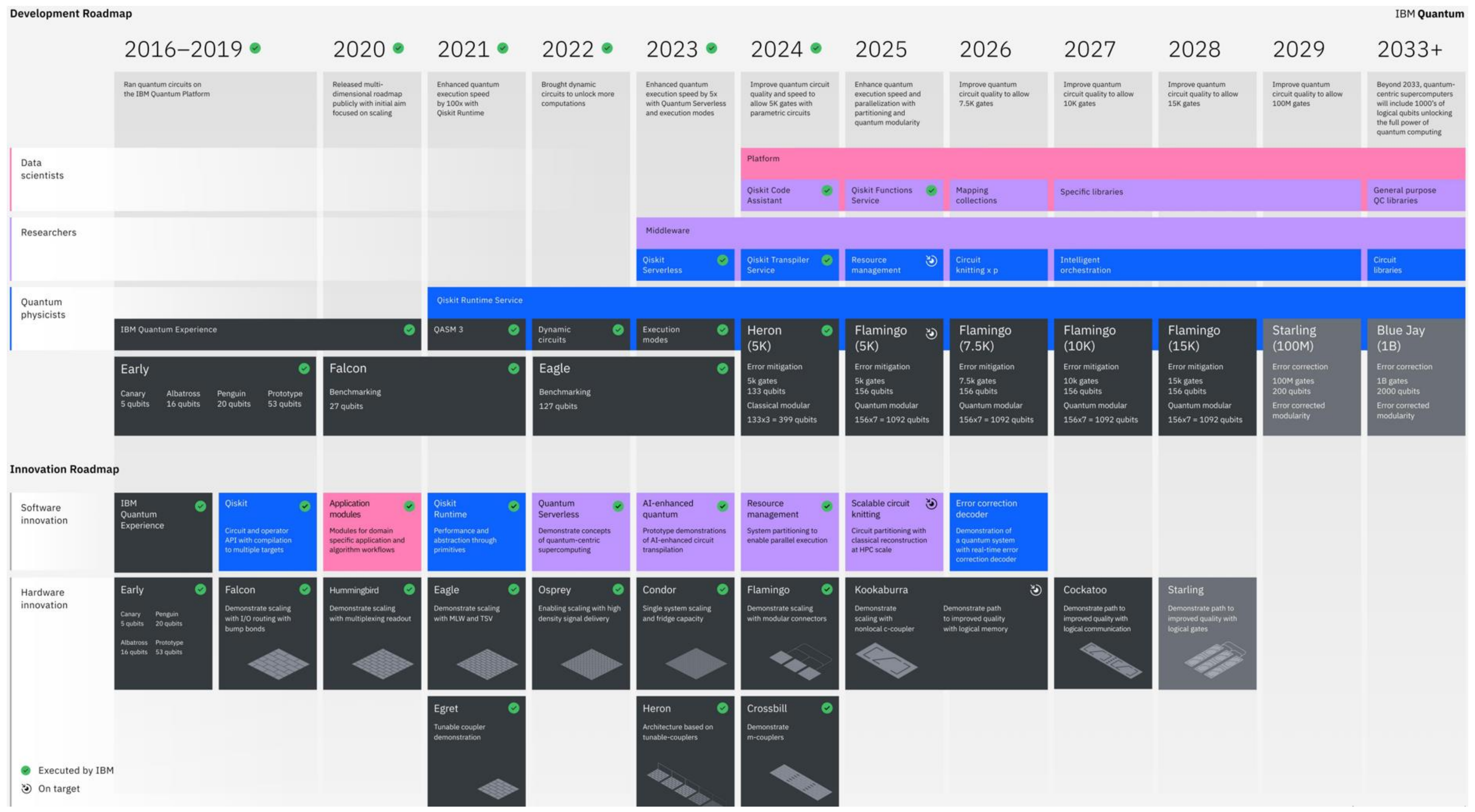
Physics

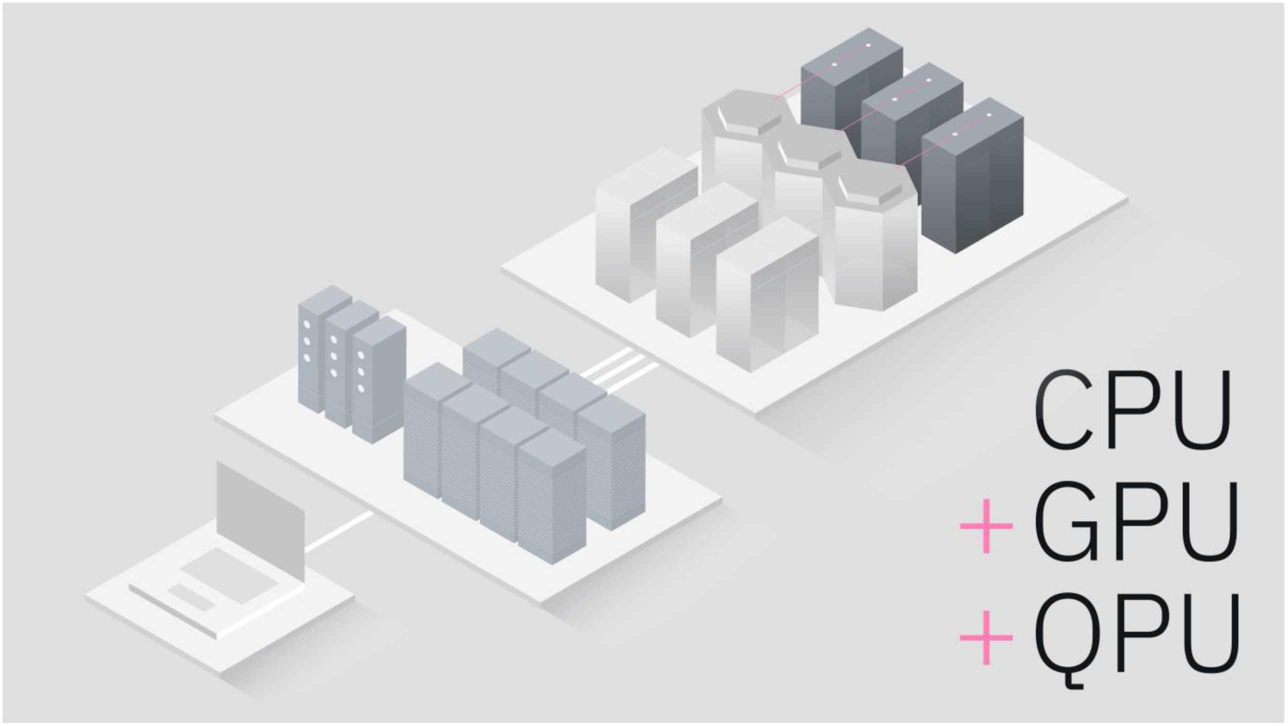


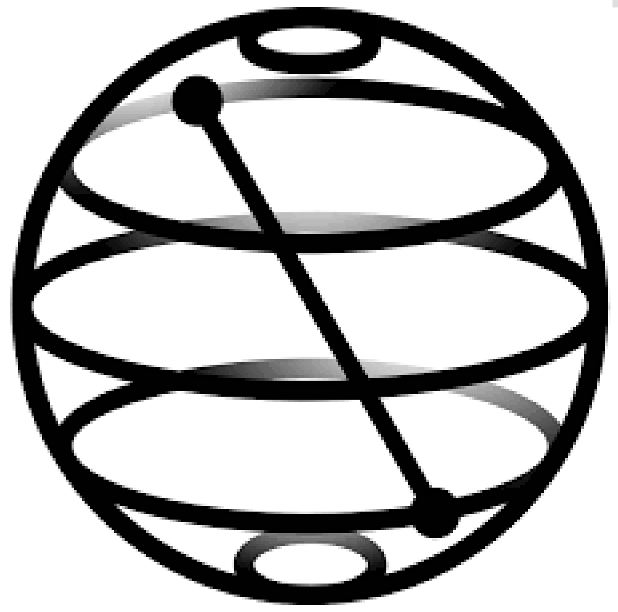
$$\mathcal{L} = -\frac{1}{4} F_{\mu\nu} F^{\mu\nu} + i \bar{\psi} \not{D} \psi + \psi_i y_{ij} \psi_j \phi + h.c. + |D_\mu \phi|^2 - V(\phi)$$

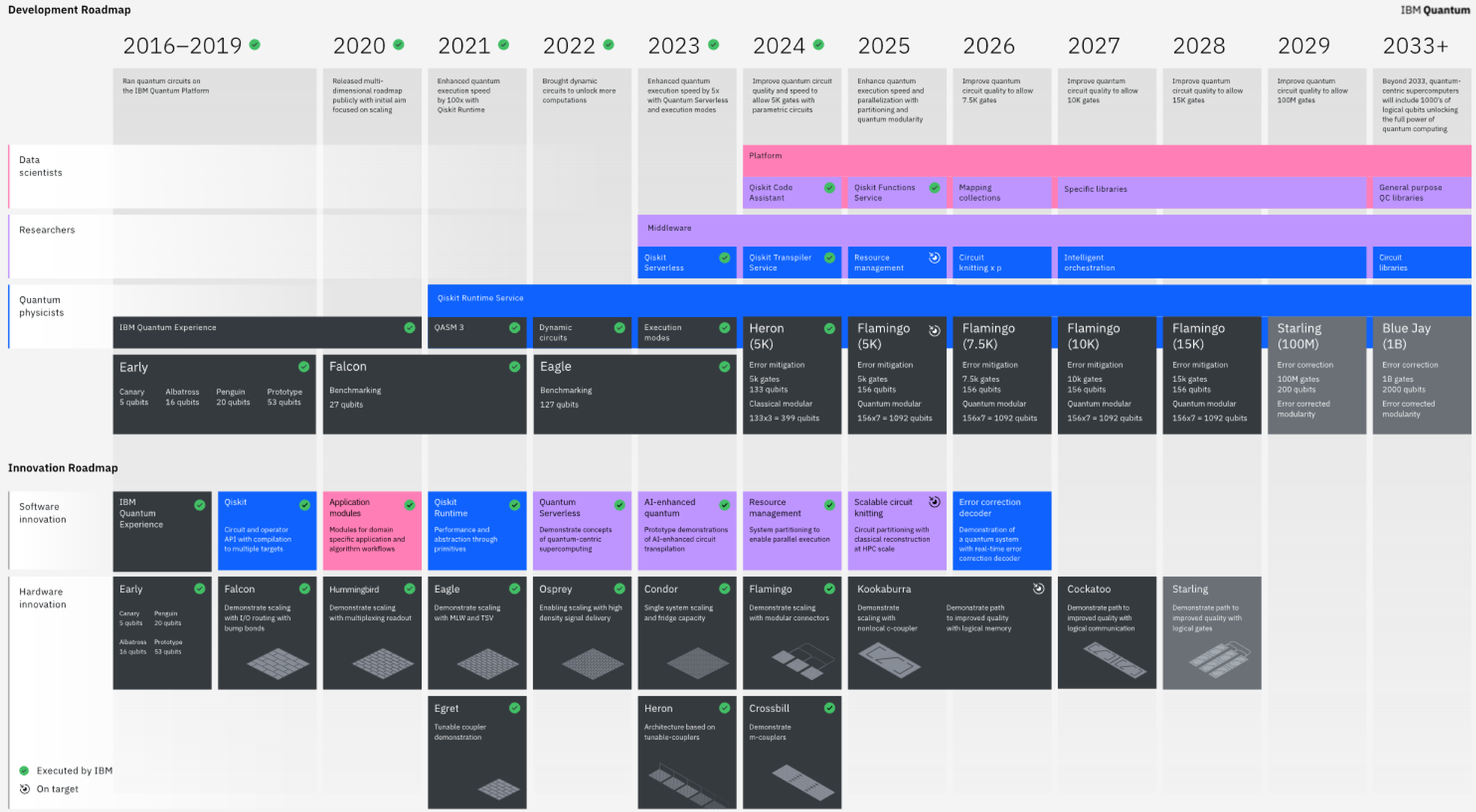


IBM Quantum





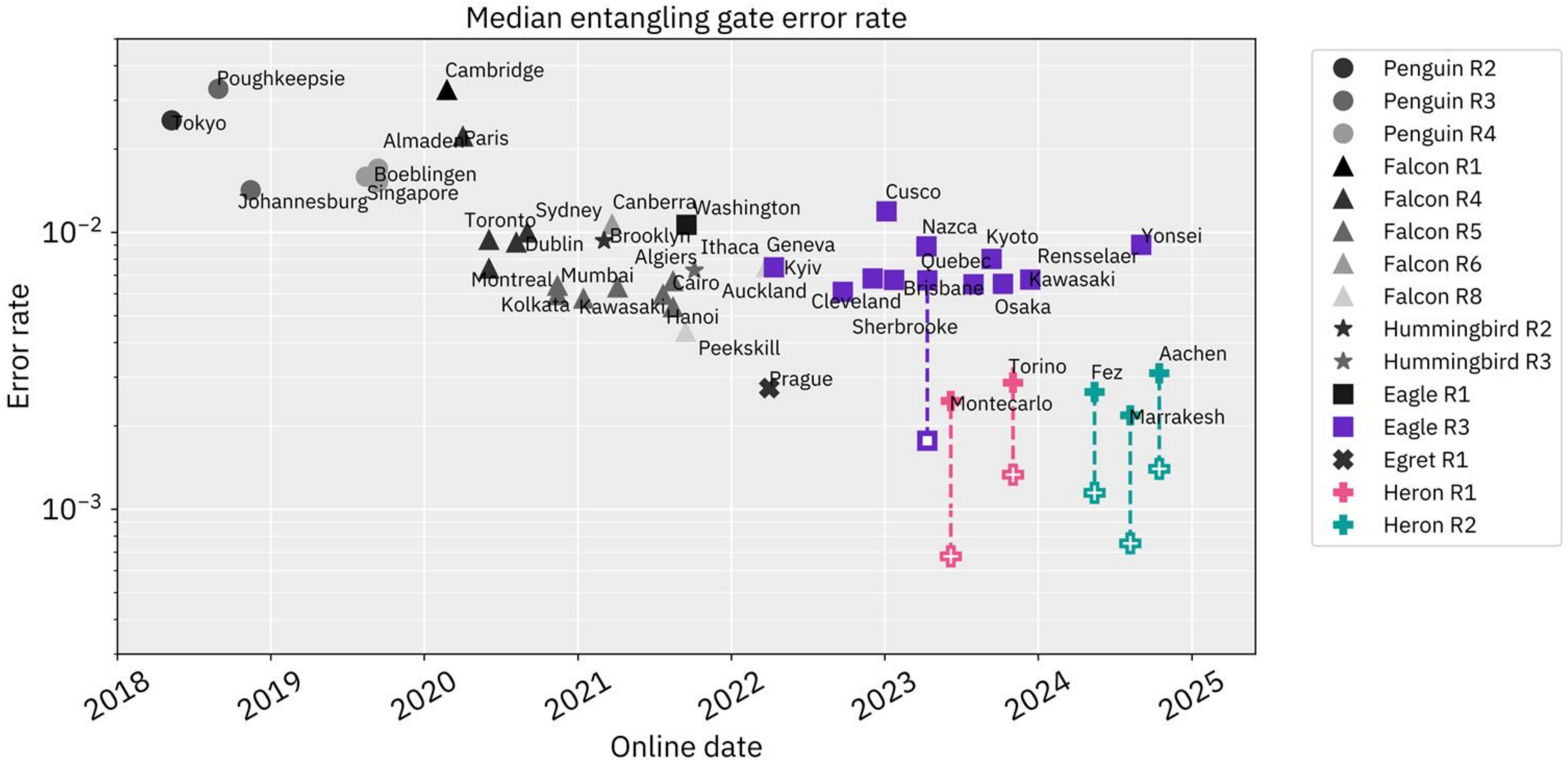


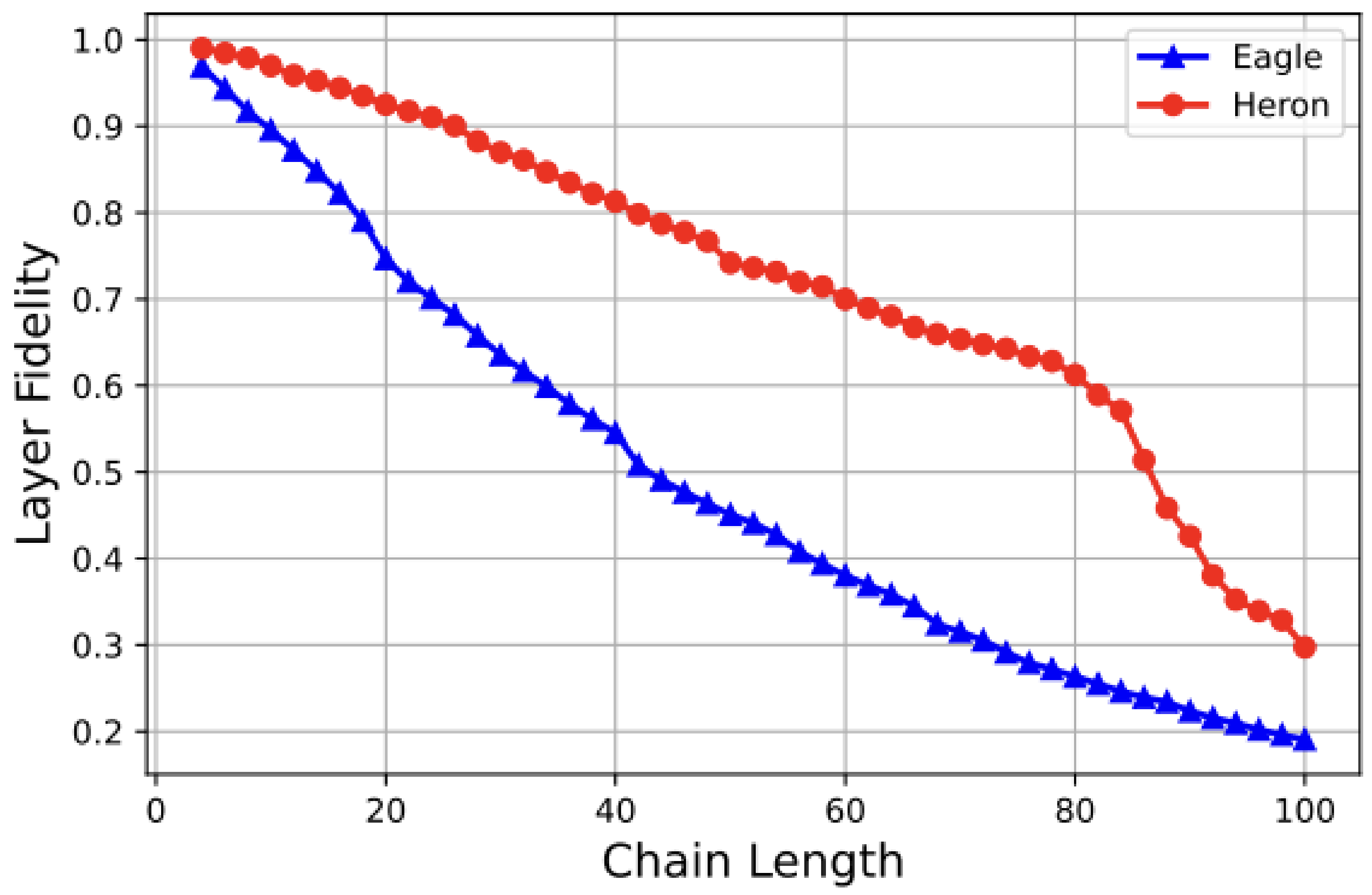


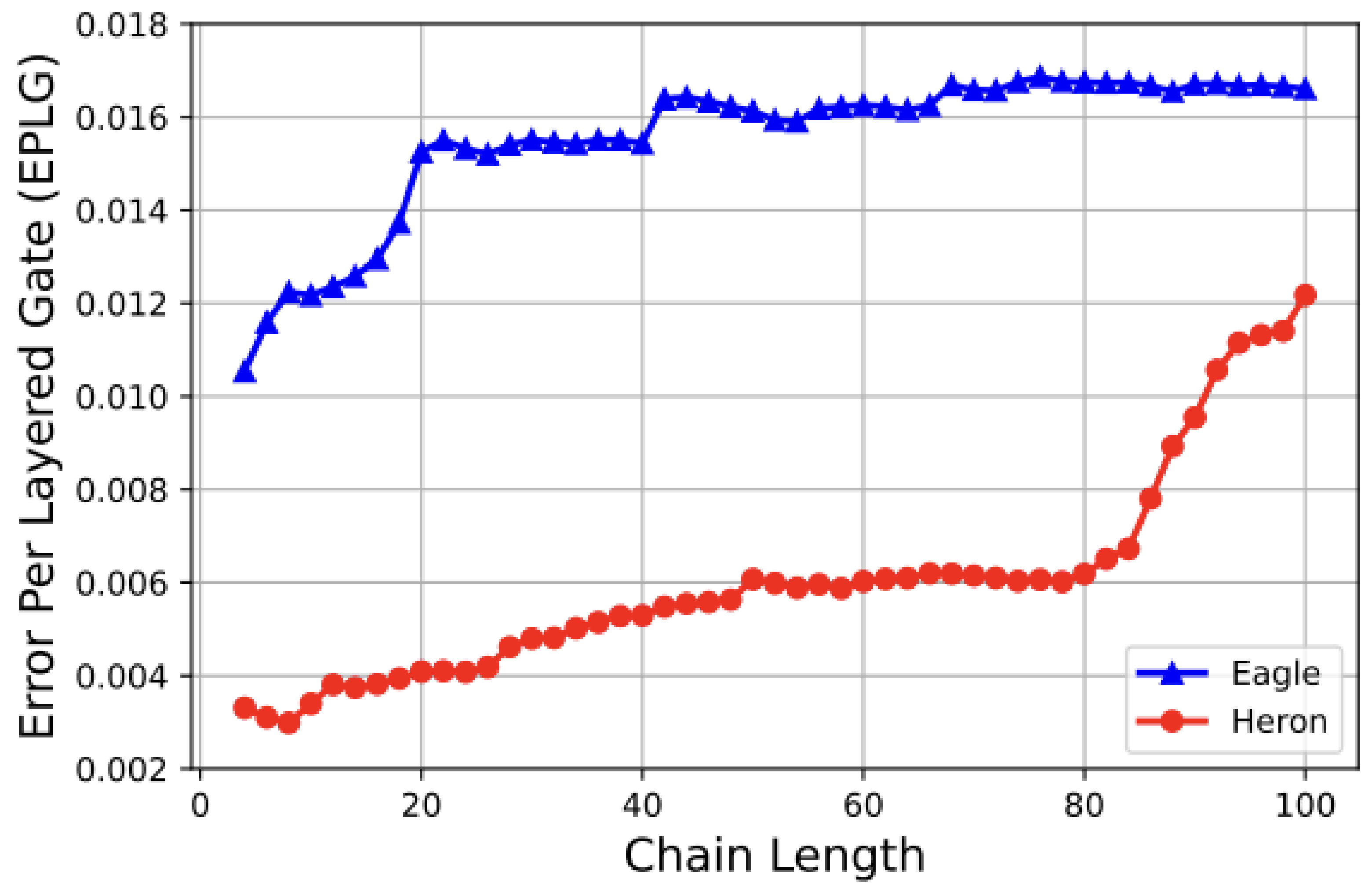
Development Roadmap
IBM Quantum
2016–2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2033+
Ran quantum circuits on the IBM Quantum Platform
Released multi-dimensional roadmap publicly with initial aim focused on scaling
Enhanced quantum execution speed by 100x with Qiskit Runtime
Brought dynamic circuits to unlock more computations
Enhanced quantum execution speed by 5x with Quantum Serverless and execution modes
Improve quantum circuit quality and speed to allow 5K gates with parametric circuits
Enhance quantum execution speed and parallelization with partitioning and quantum modularity
Improve quantum circuit quality to allow 7.5K gates
Improve quantum circuit quality to allow 10K gates
Improve quantum circuit quality to allow 15K gates
Improve quantum circuit quality to allow 100M gates
Beyond 2033, quantum-centric supercomputers will include 1000's of logical qubits unlocking the full power of quantum computing
Data scientists
Platform
Qiskit Code Assistant
Qiskit Functions Service
Mapping collections
Specific libraries
General purpose QC libraries
Researchers
Middleware
Qiskit Serverless
Qiskit Transpiler Service
Resource management
Circuit knitting x p
Intelligent orchestration
Circuit libraries
Quantum physicists
Qiskit Runtime Service
IBM Quantum Experience
QASM 3
Dynamic circuits
Execution modes
Heron (5K)
Error mitigation
5k gates
133 qubits
Classical modular
133x3 = 399 qubits
Flamingo (5K)
Error mitigation
5k gates
156 qubits
Quantum modular
156x7 = 1092 qubits
Flamingo (7.5K)
Error mitigation
7.5k gates
156 qubits
Quantum modular
156x7 = 1092 qubits
Flamingo (10K)
Error mitigation
10k gates
156 qubits
Quantum modular
156x7 = 1092 qubits
Flamingo (15K)
Error mitigation
15k gates
156 qubits
Quantum modular
156x7 = 1092 qubits
Starling (100M)
Error correction
100M gates
200 qubits
Error corrected modularity
Blue Jay (1B)
Error correction
1B gates
2000 qubits
Error corrected modularity
Early
Canary 5 qubits
Albatross 16 qubits
Penguin 20 qubits
Prototype 53 qubits
Falcon
Benchmarking
27 qubits
Eagle
Benchmarking
127 qubits
Innovation Roadmap
Software innovation
IBM Quantum Experience
Qiskit
Circuit and operator API with compilation to multiple targets
Application modules
Modules for domain specific application and algorithm workflows
Qiskit Runtime
Performance and abstraction through primitives
Quantum Serverless
Demonstrate concepts of quantum-centric supercomputing
AI-enhanced quantum
Prototype demonstrations of AI-enhanced circuit transpilation
Resource management
System partitioning to enable parallel execution
Scalable circuit knitting
Circuit partitioning with classical reconstruction at HPC scale
Error correction decoder
Demonstration of a quantum system with real-time error correction decoder
Hardware innovation
Early
Canary 5 qubits
Penguin 20 qubits
Albatross 16 qubits
Prototype 53 qubits
Falcon
Demonstrate scaling with I/O routing with bump bonds
Hummingbird
Demonstrate scaling with multiplexing readout
Eagle
Demonstrate scaling with MLW and TSV
Osprey
Enabling scaling with high density signal delivery
Condor
Single system scaling and fridge capacity
Flamingo
Demonstrate scaling with modular connectors
Kookaburra
Demonstrate scaling with nonlocal c-coupler
Demonstrate path to improved quality with logical memory
Cockatoo
Demonstrate path to improved quality with logical communication
Starling
Demonstrate path to improved quality with logical gates
Egret
Tunable coupler demonstration
Heron
Architecture based on tunable-couplers
Crossbill
Demonstrate m-couplers
Executed by IBM
On target

Evolution of IBM Quantum backends

arXiv:2311.05933



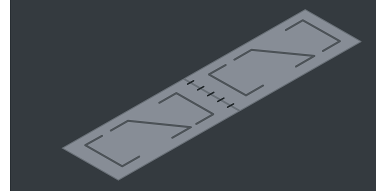




arXiv:2209.06841

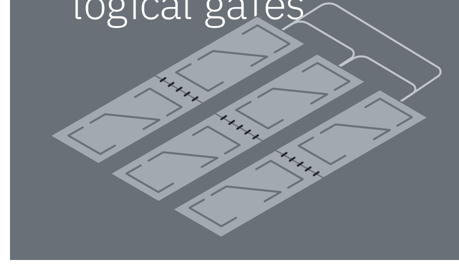
Cockatoo

Demonstrate path to improved quality with logical communication



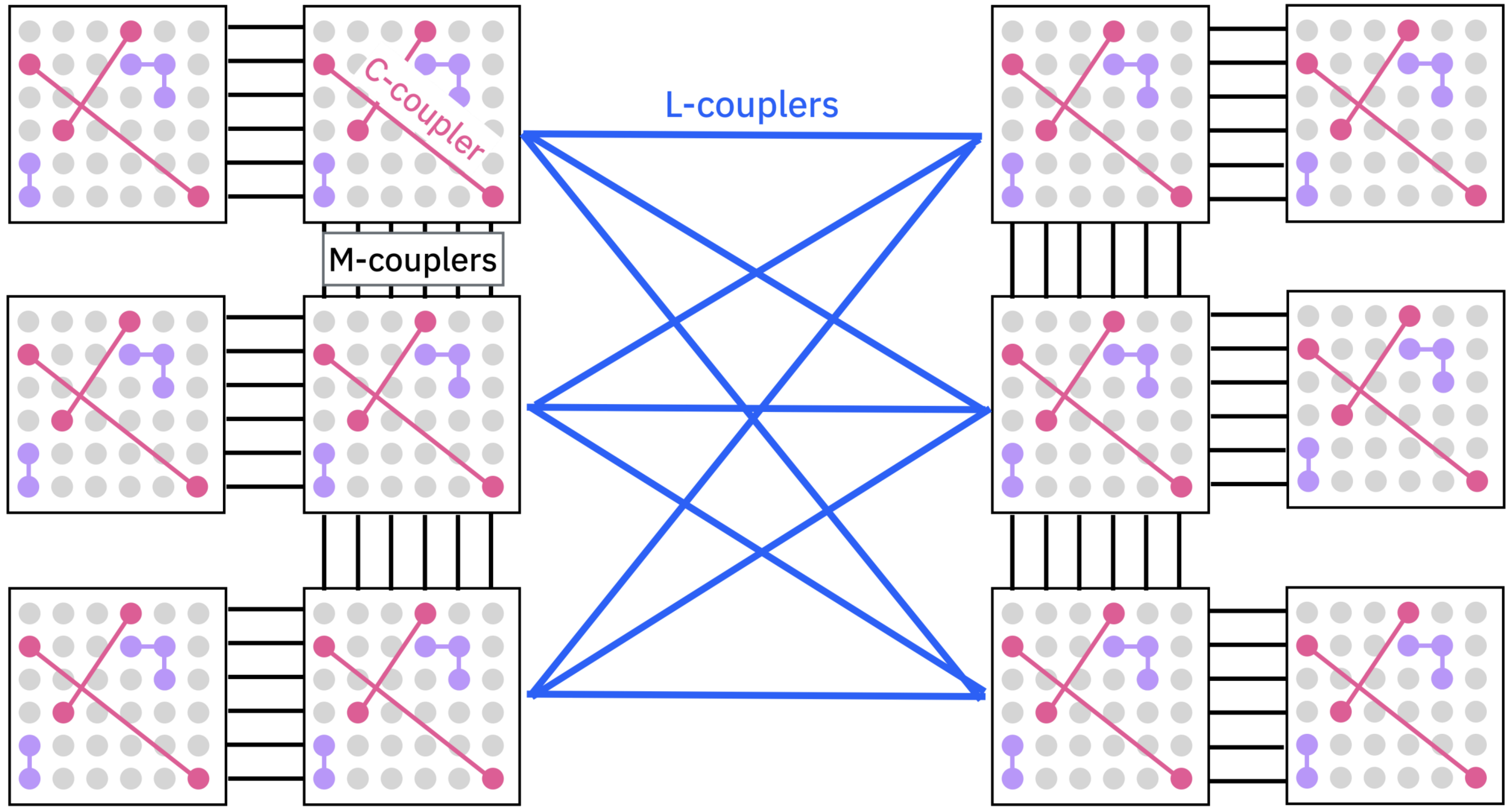
Starling

Demonstrate path to improved quality with logical gates



New innovations required

C-couplers, which are non-local connections, and 6-way connectivity

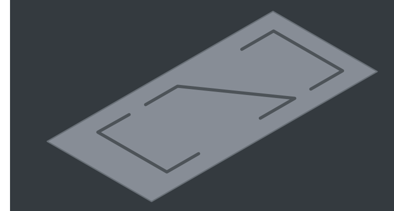


C-coupler

Kookaburra

Demonstrate scaling with non-local c-coupler

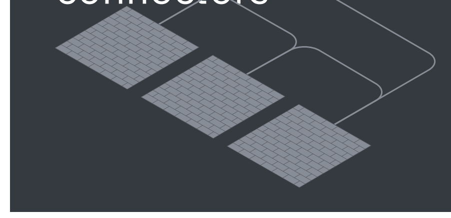
Demonstrate path to improved quality with logical memory



L-coupler

Flamingo

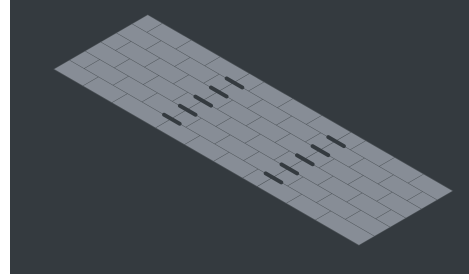
Demonstrate scaling with modular connectors



M-coupler

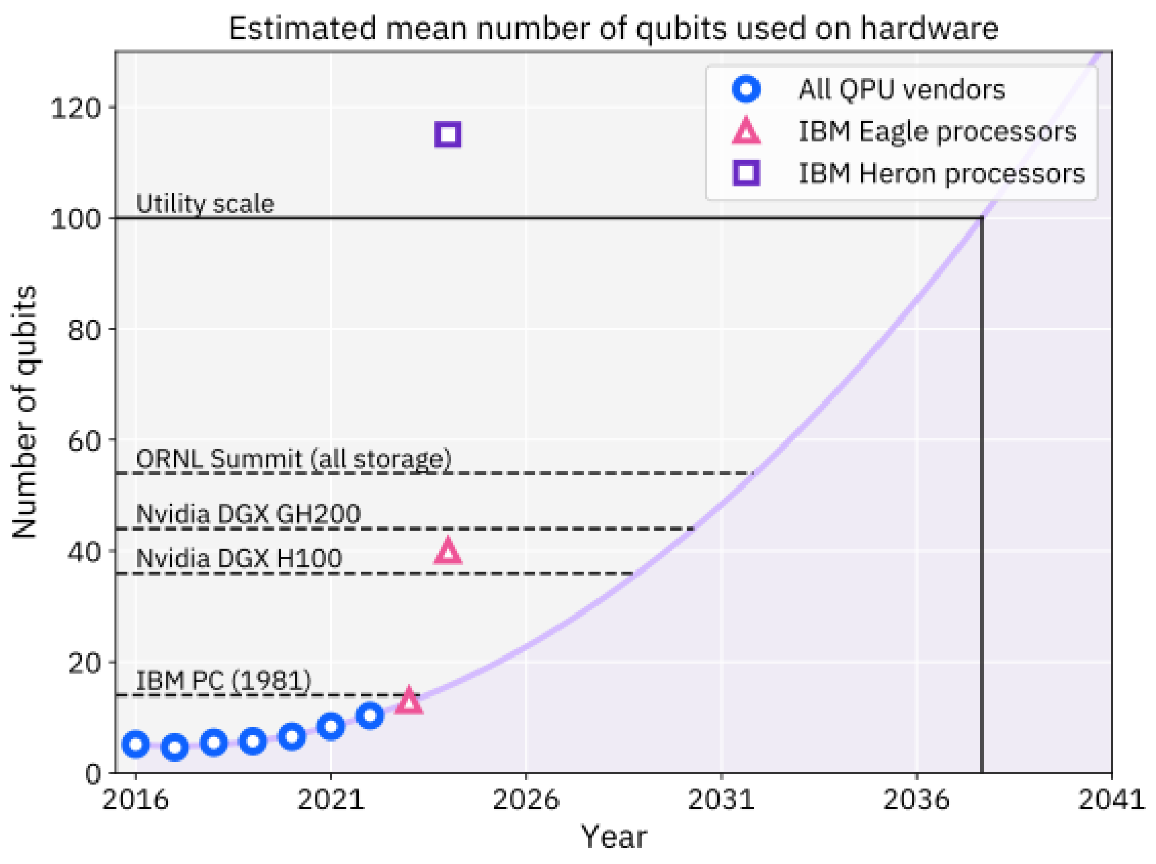
Crossbill

m- coupler



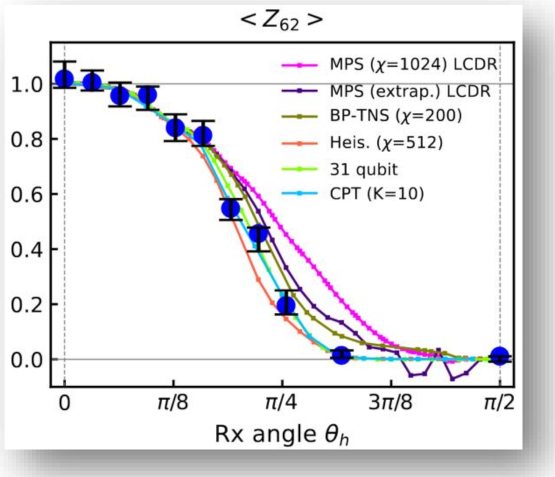
The era of quantum utility (c. 2022)

Quantum simulation beyond classical brute-force.



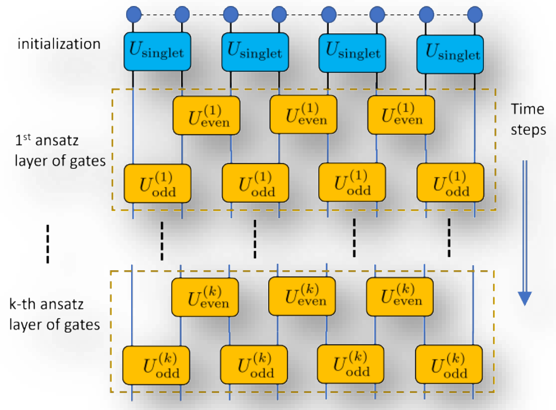
Evidence for the utility of quantum computing before fault tolerance [Nature, 618, 500 (2023)]

127 qubits / 2880 CX gates



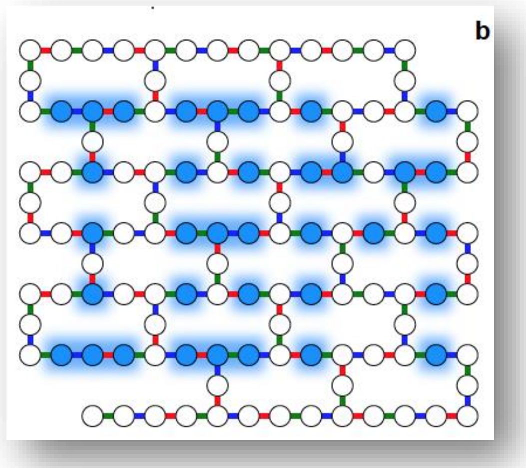
Simulating large-size quantum spin chains on cloud-based superconducting quantum computers [arXiv:2207.09994]

102 qubits / 3186 CX gates



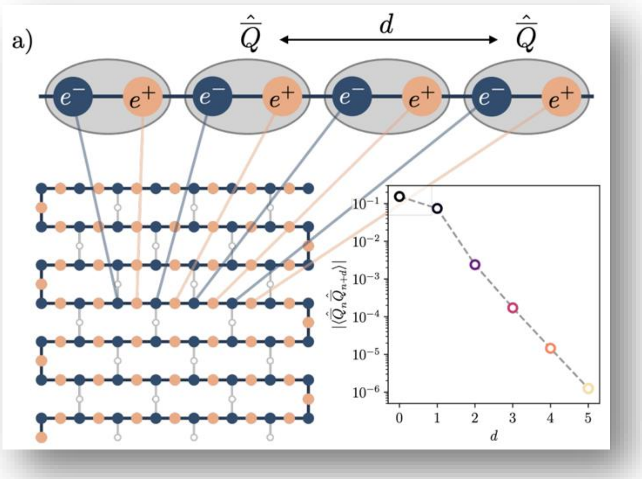
Uncovering local integrability in quantum many-body dynamics [arXiv:2307.07552]

124 qubits / 2641 CX gates



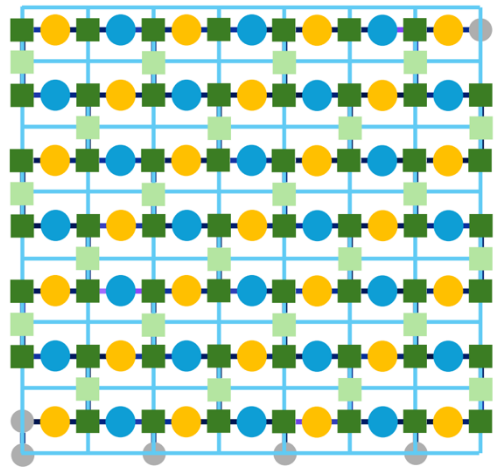
Quantum simulations of hadron dynamics in the Schwinger model using 112 qubits [arXiv:2401.08044]

112 qubits / 13858 CX gates



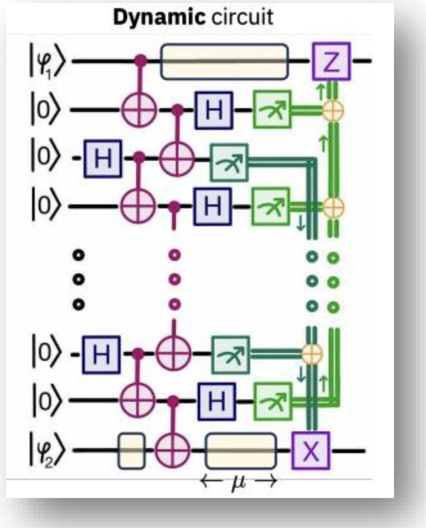
Quantum Simulation of SU(3) Lattice Yang Mills Theory at Leading Order in Large N [arXiv:2402.10265]

127 qubits / 113 CX layers



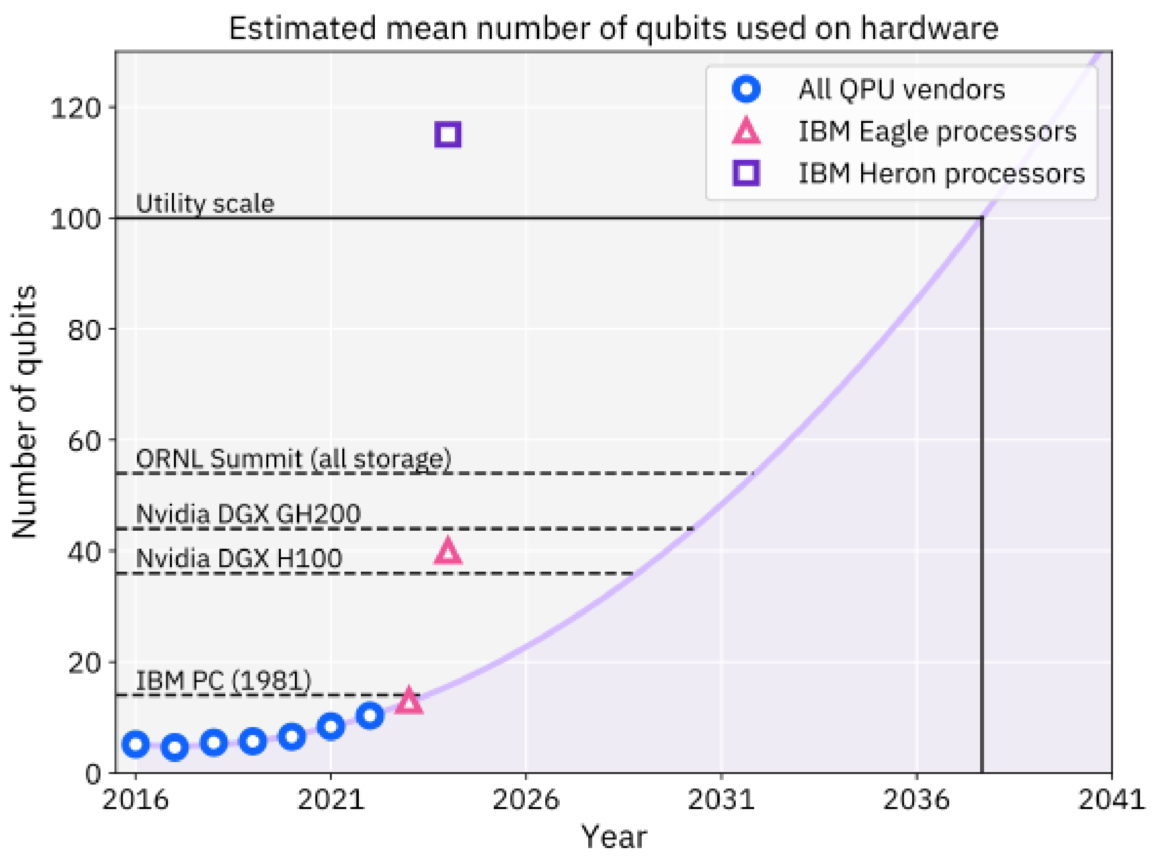
Efficient long-range entanglement using dynamic circuits [arXiv:2308.13065]

101 qubits / 504 gates + meas.



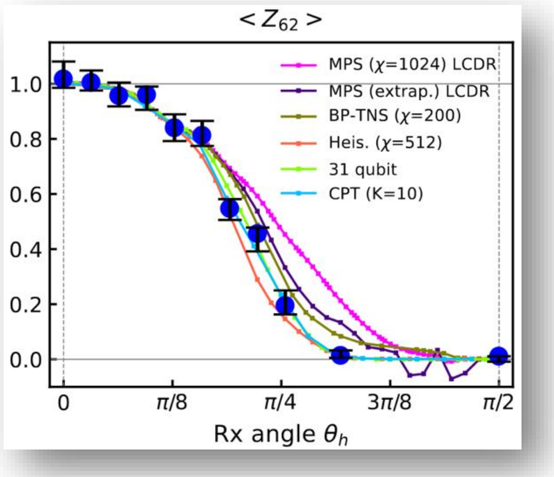
The era of quantum utility (c. 2022)

Quantum simulation beyond classical brute-force.



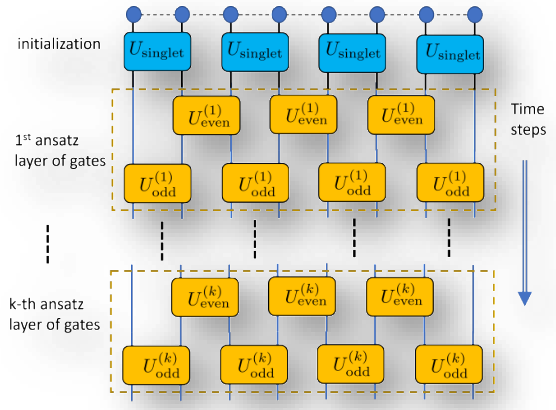
Evidence for the utility of quantum computing before fault tolerance [Nature, 618, 500 (2023)]

127 qubits / 2880 CX gates



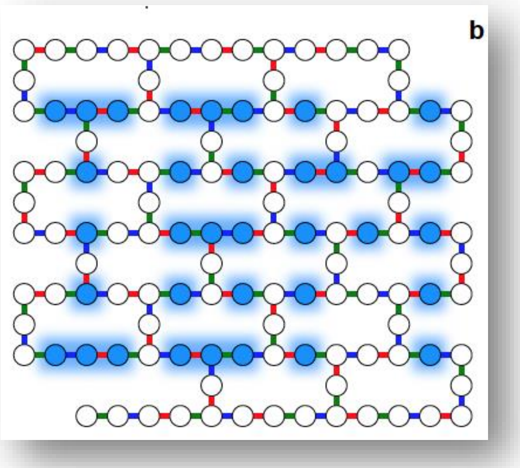
Simulating large-size quantum spin chains on cloud-based superconducting quantum computers [arXiv:2207.09994]

102 qubits / 3186 CX gates



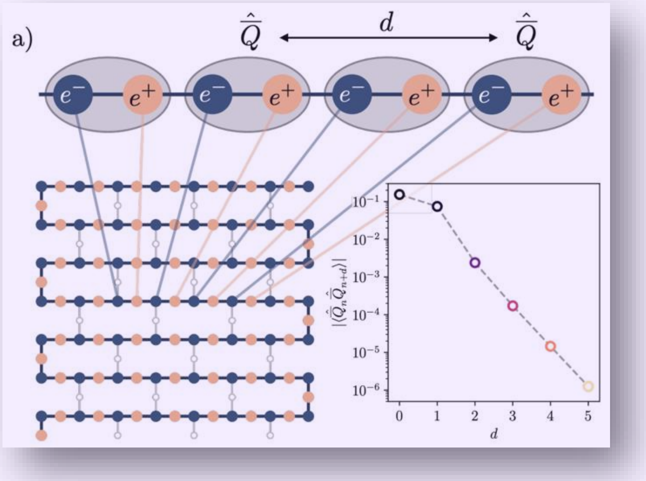
Uncovering local integrability in quantum many-body dynamics [arXiv:2307.07552]

124 qubits / 2641 CX gates



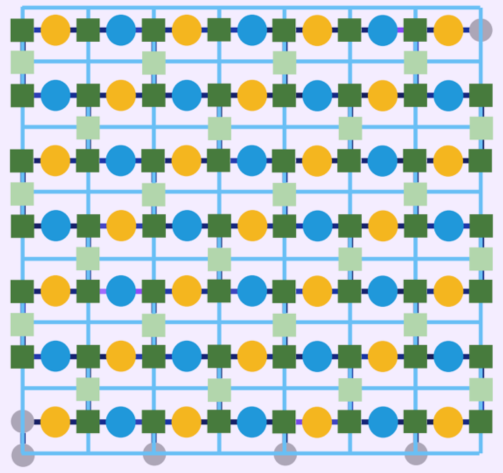
Quantum simulations of hadron dynamics in the Schwinger model using 112 qubits [arXiv:2401.08044]

112 qubits / 13858 CX gates



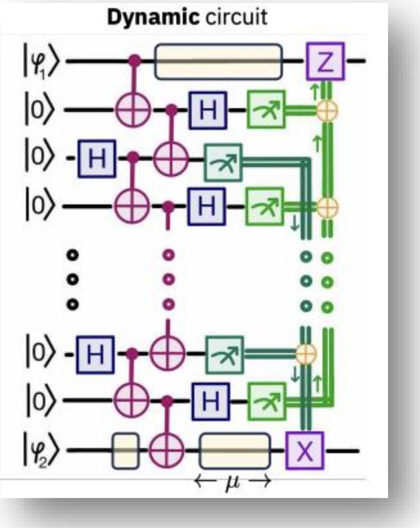
Quantum Simulation of SU(3) Lattice Yang Mills Theory at Leading Order in Large N [arXiv:2402.10265]

127 qubits / 113 CX layers



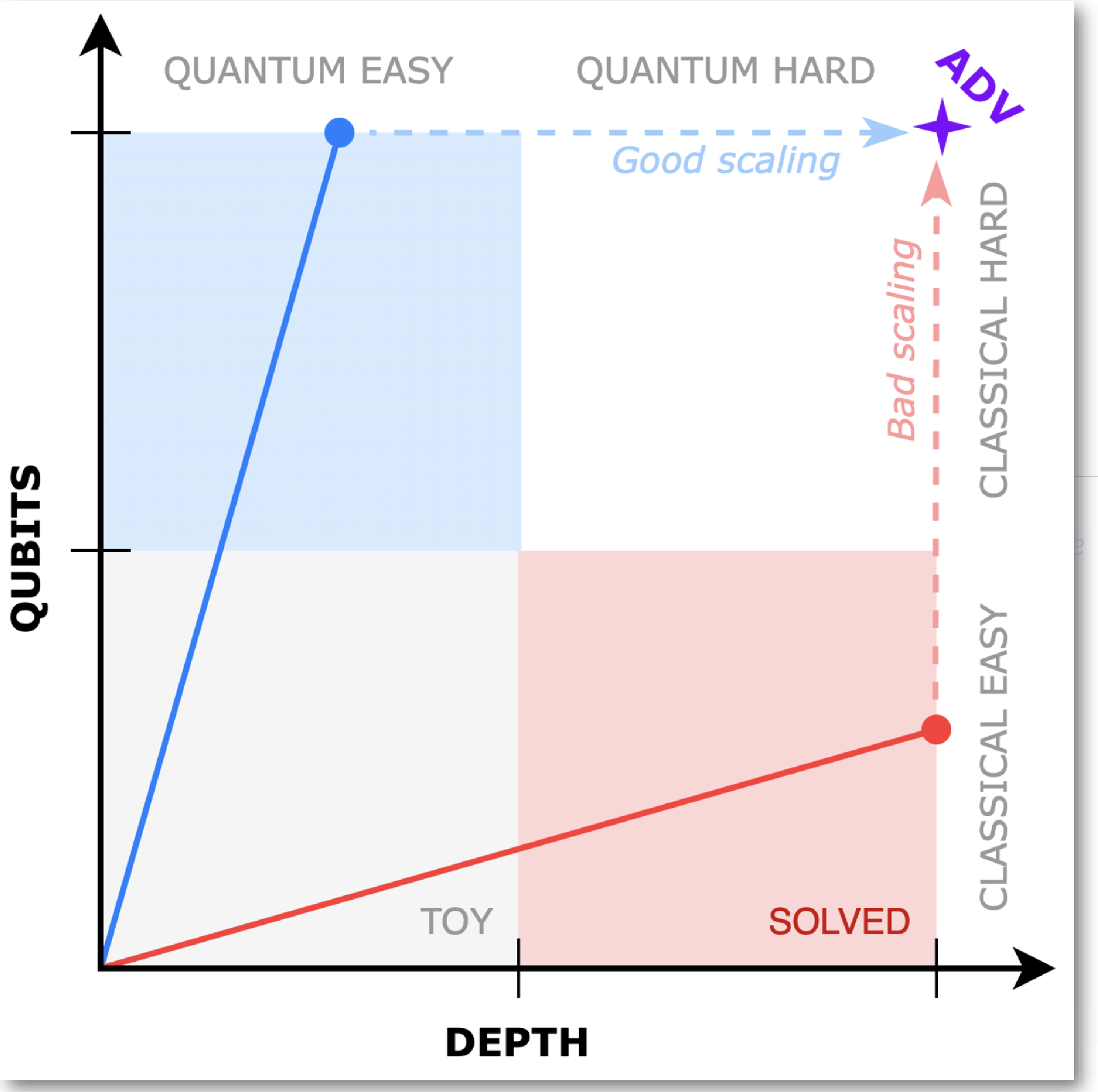
Efficient long-range entanglement using dynamic circuits [arXiv:2308.13065]

101 qubits / 504 gates + meas.



The era of quantum utility (c. 2022)

Quantum simulation beyond classical brute-force.



Uncovering local integrability in quantum many-body dynamics
[arXiv:2307.07552]

124 qubits / 2641 CX gates

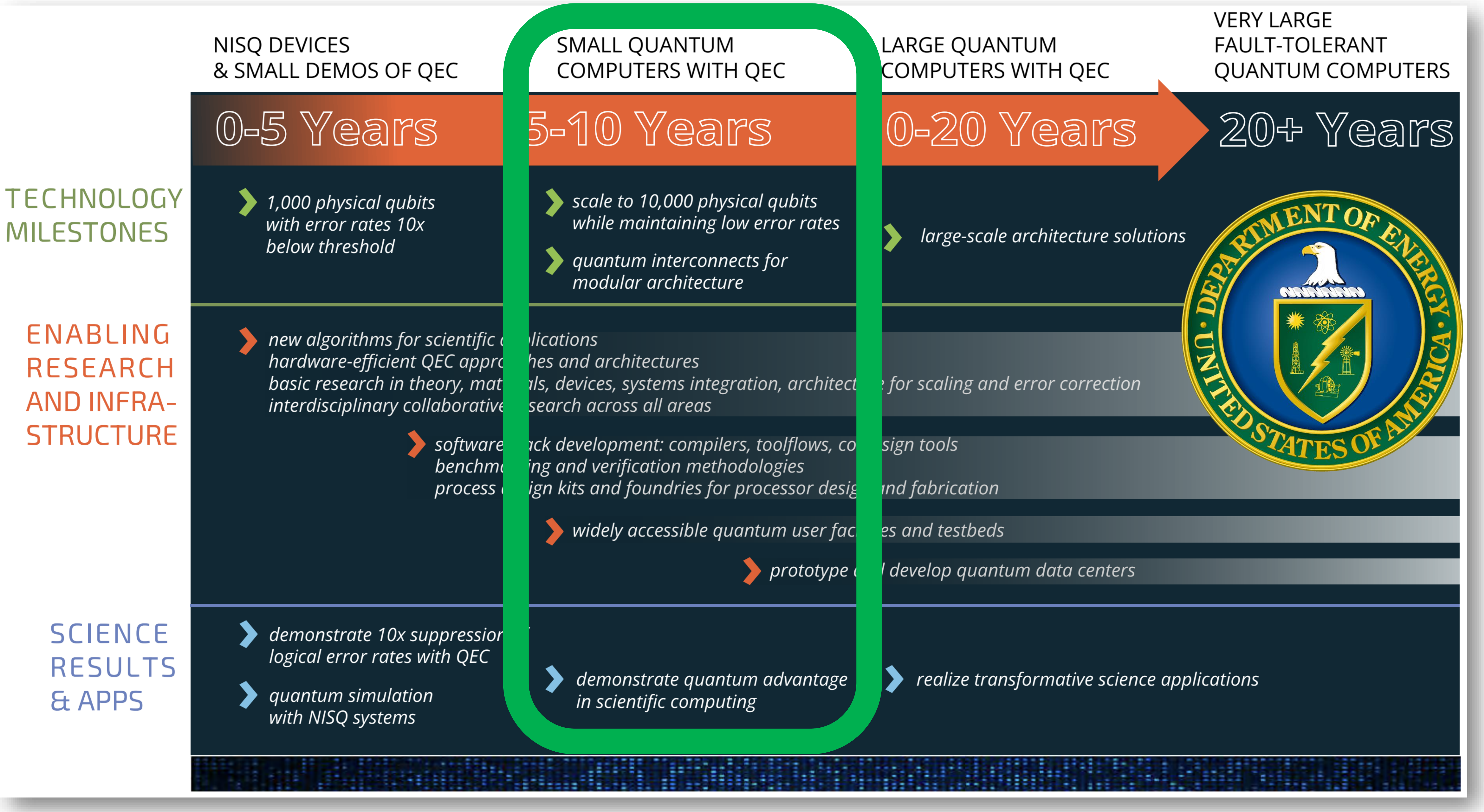
Efficient long-range entanglement using dynamic circuits
[arXiv:2308.13065]

101 qubits / 504 gates + meas.

“Getting ‘very useful quantum computers’ to market could take 15 to 30 years.”

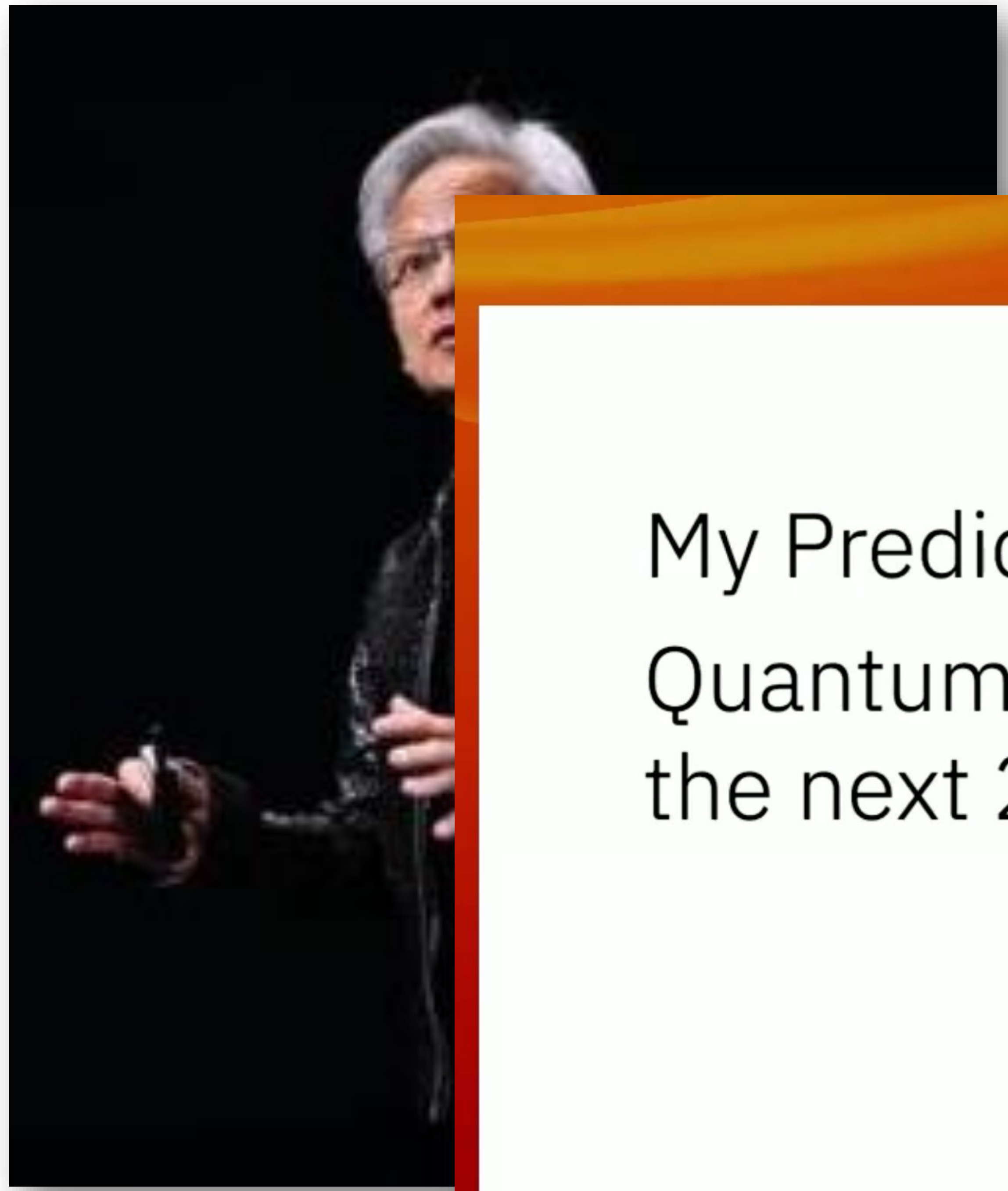
https://science.osti.gov/-/media/QIS/pdf/DOE_QIS_Roadmap_Final.pdf

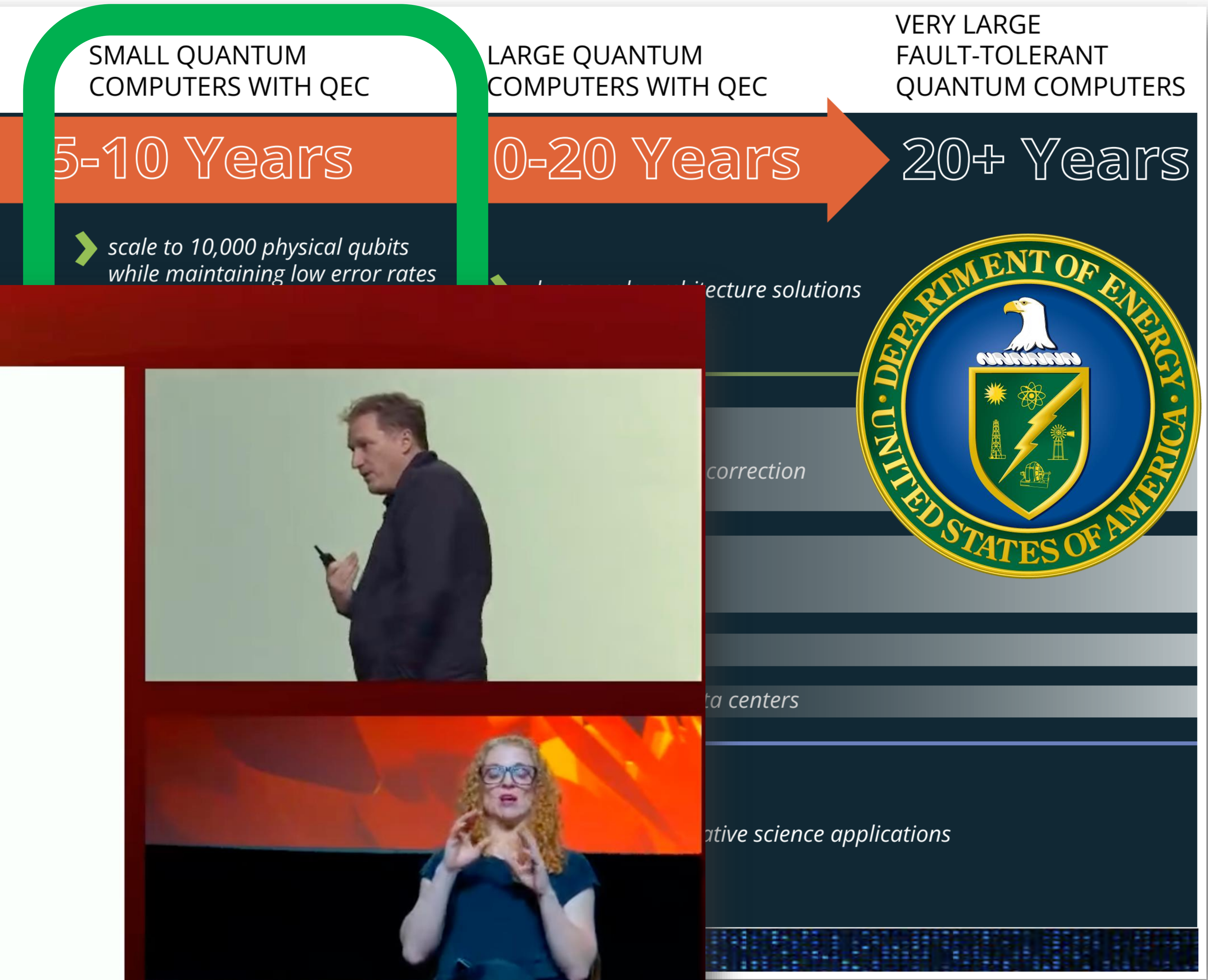
“Getting ‘very useful quantum computers’ to market could take 15 to 30 years.”

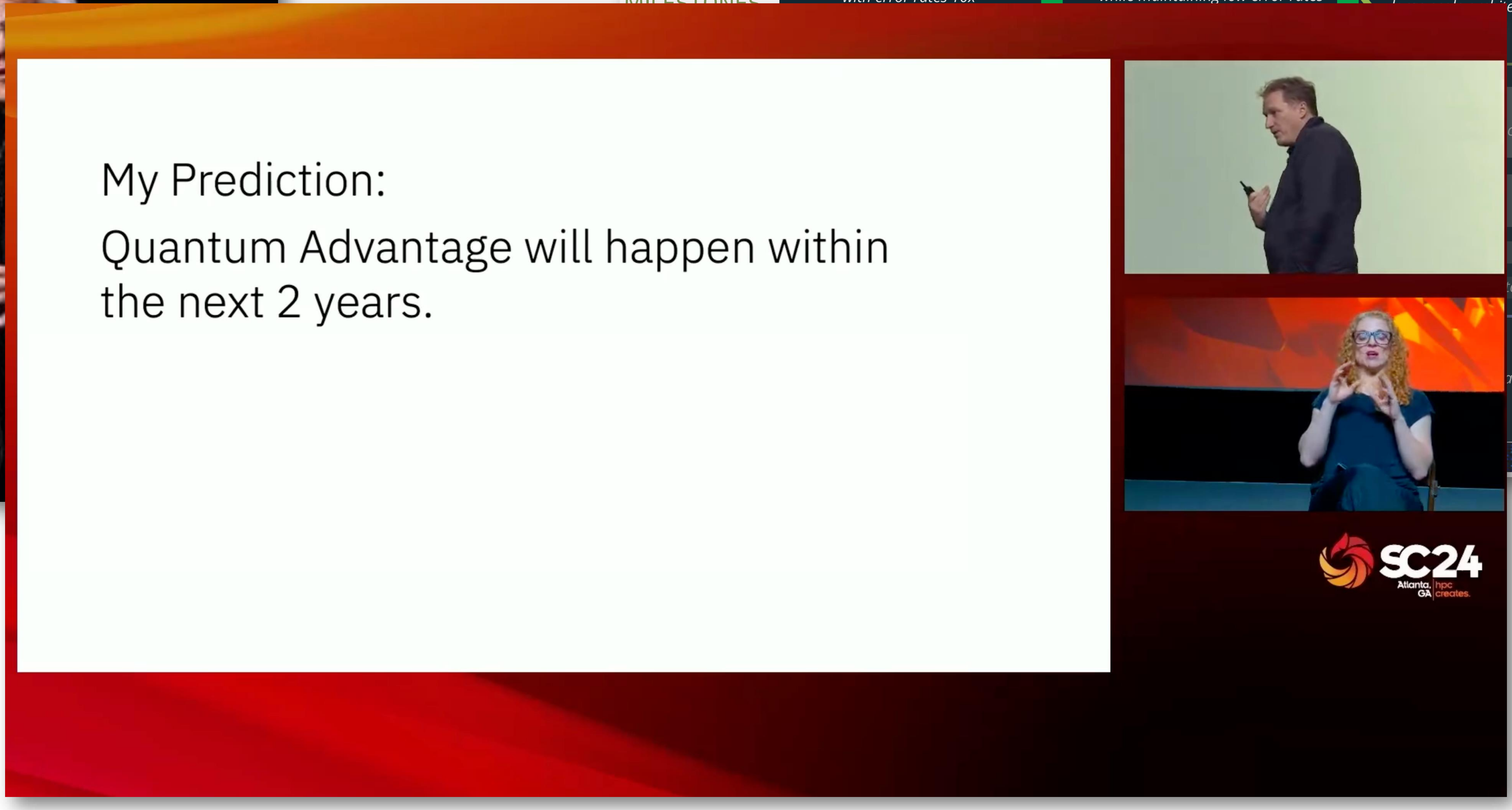


https://science.osti.gov/-/media/QIS/pdf/DOE_QIS_Roadmap_Final.pdf

"Getting 'very useful quantum computers' to market could take 15 to 30 years."

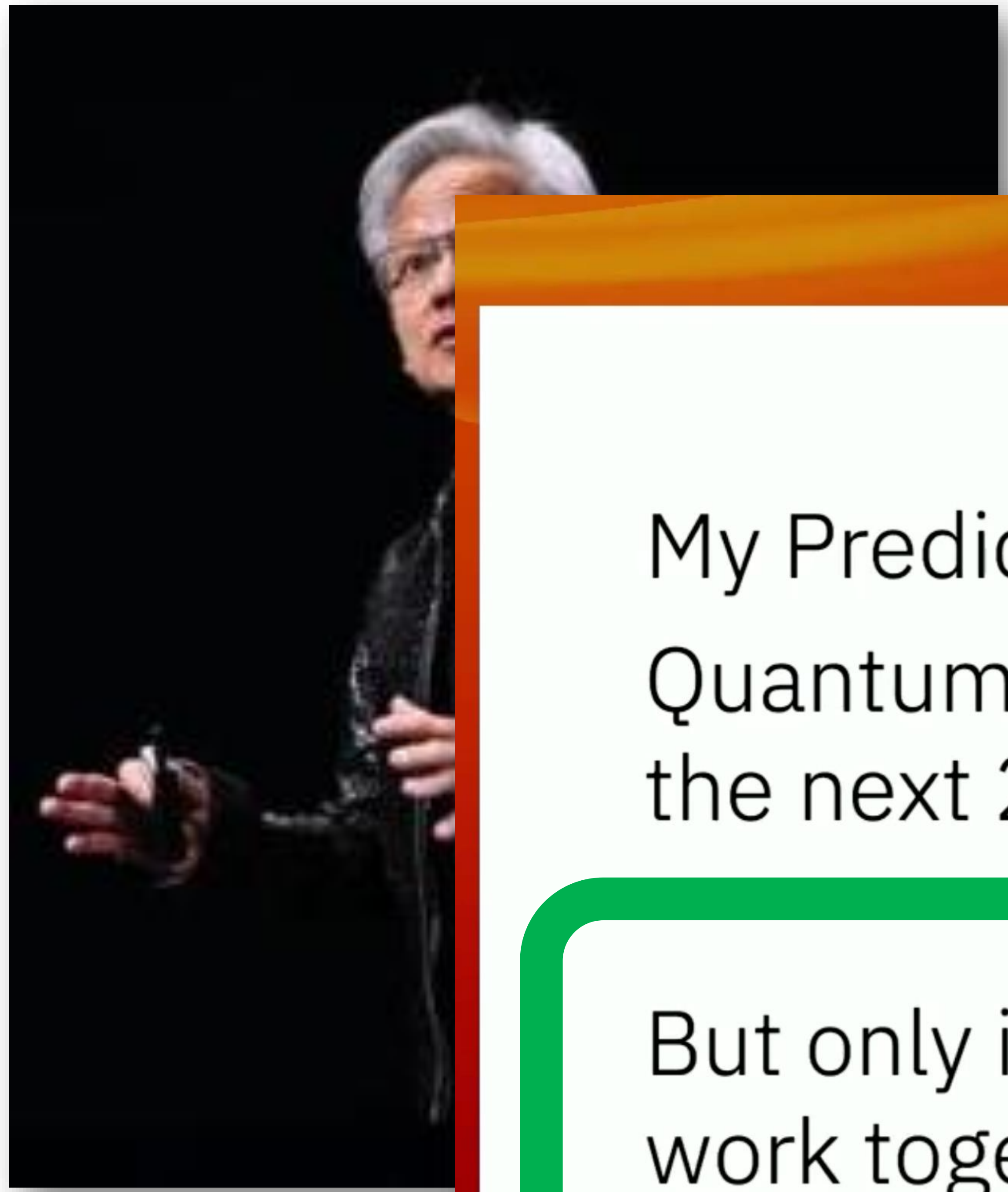


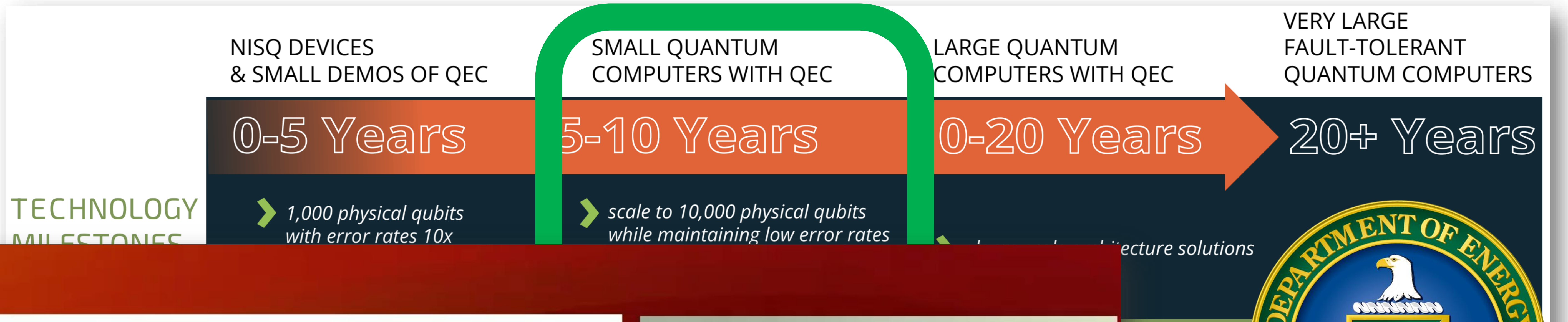


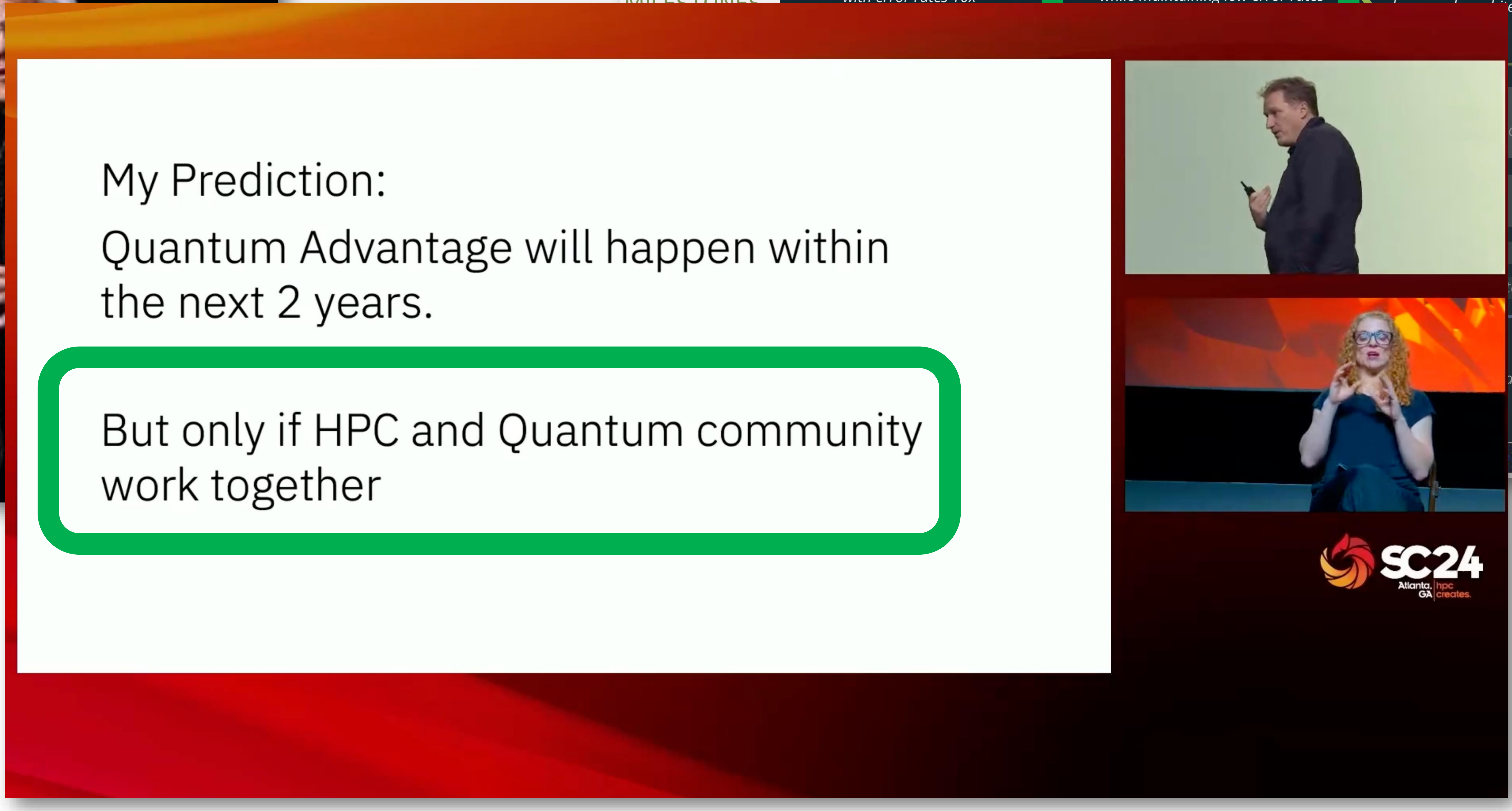


https://science.osti.gov/-/media/QIS/pdf/DOE_QIS_Roadmap_Final.pdf

“Getting ‘very useful quantum computers’ to market could take 15 to 30 years.”

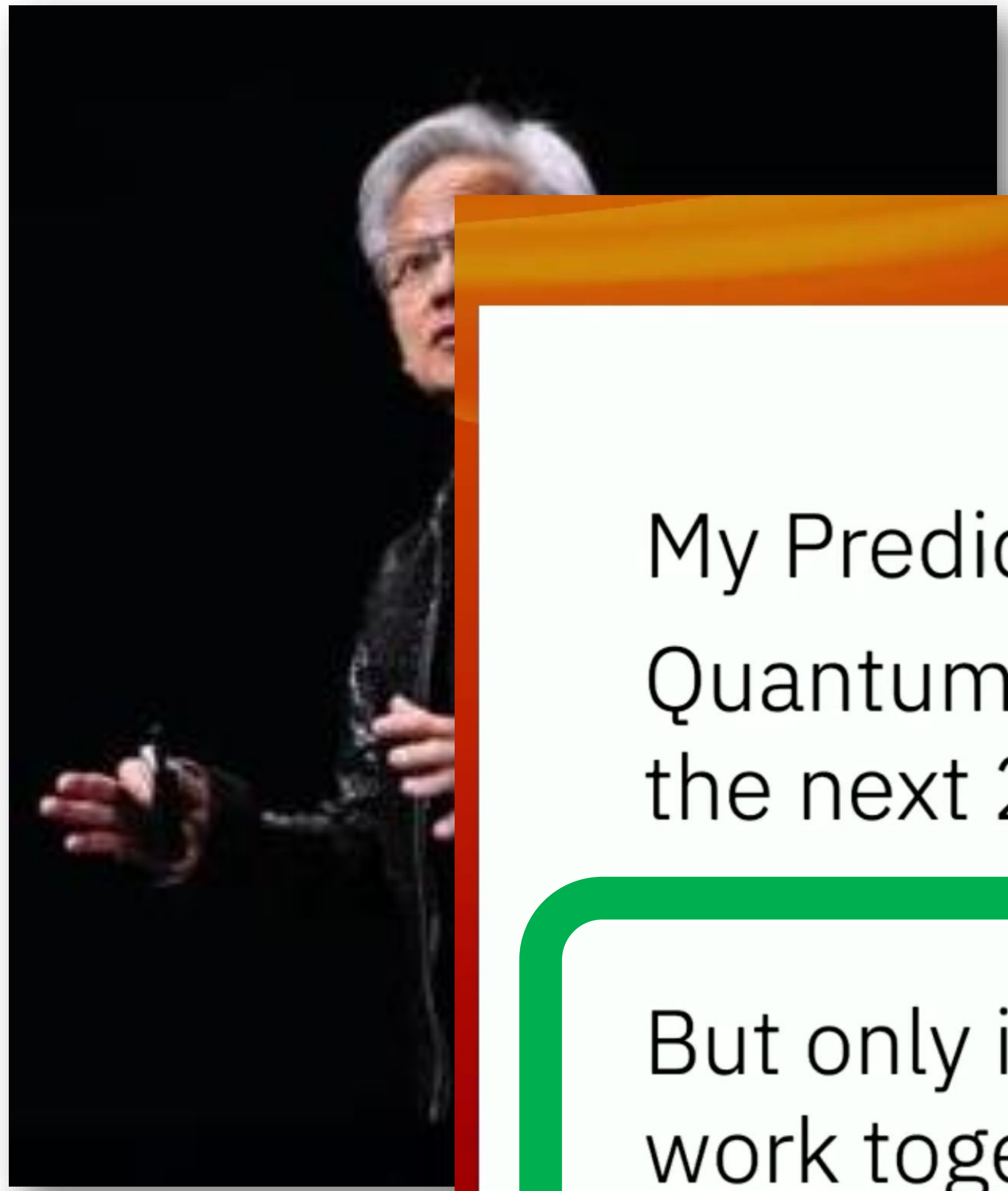


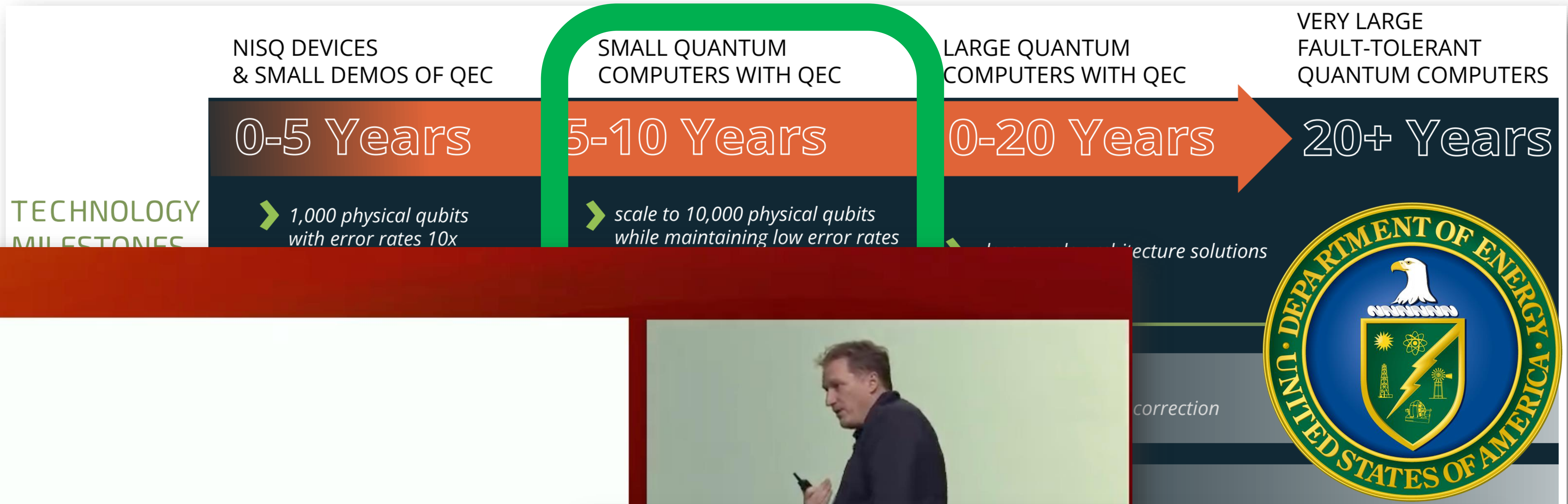


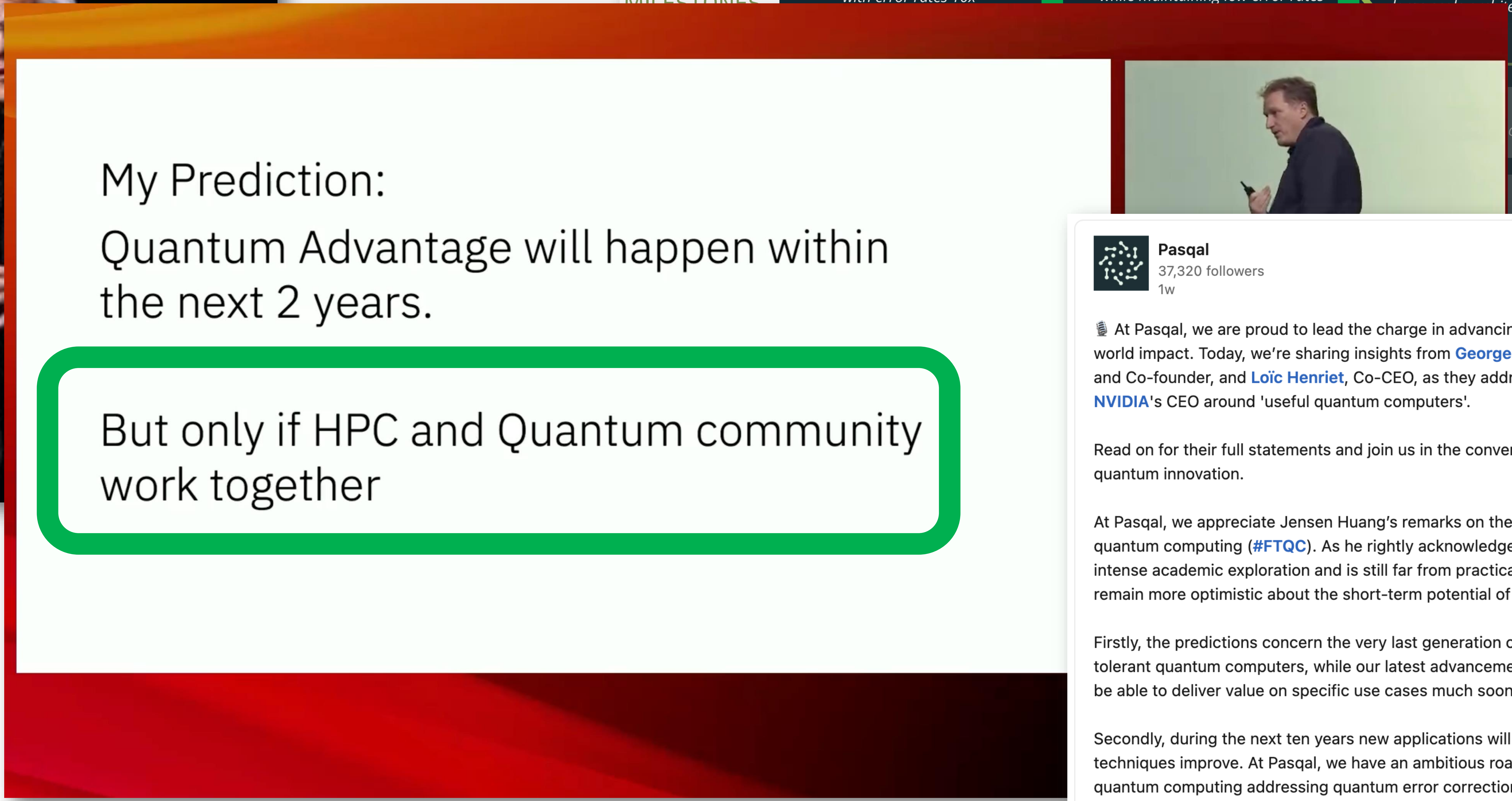


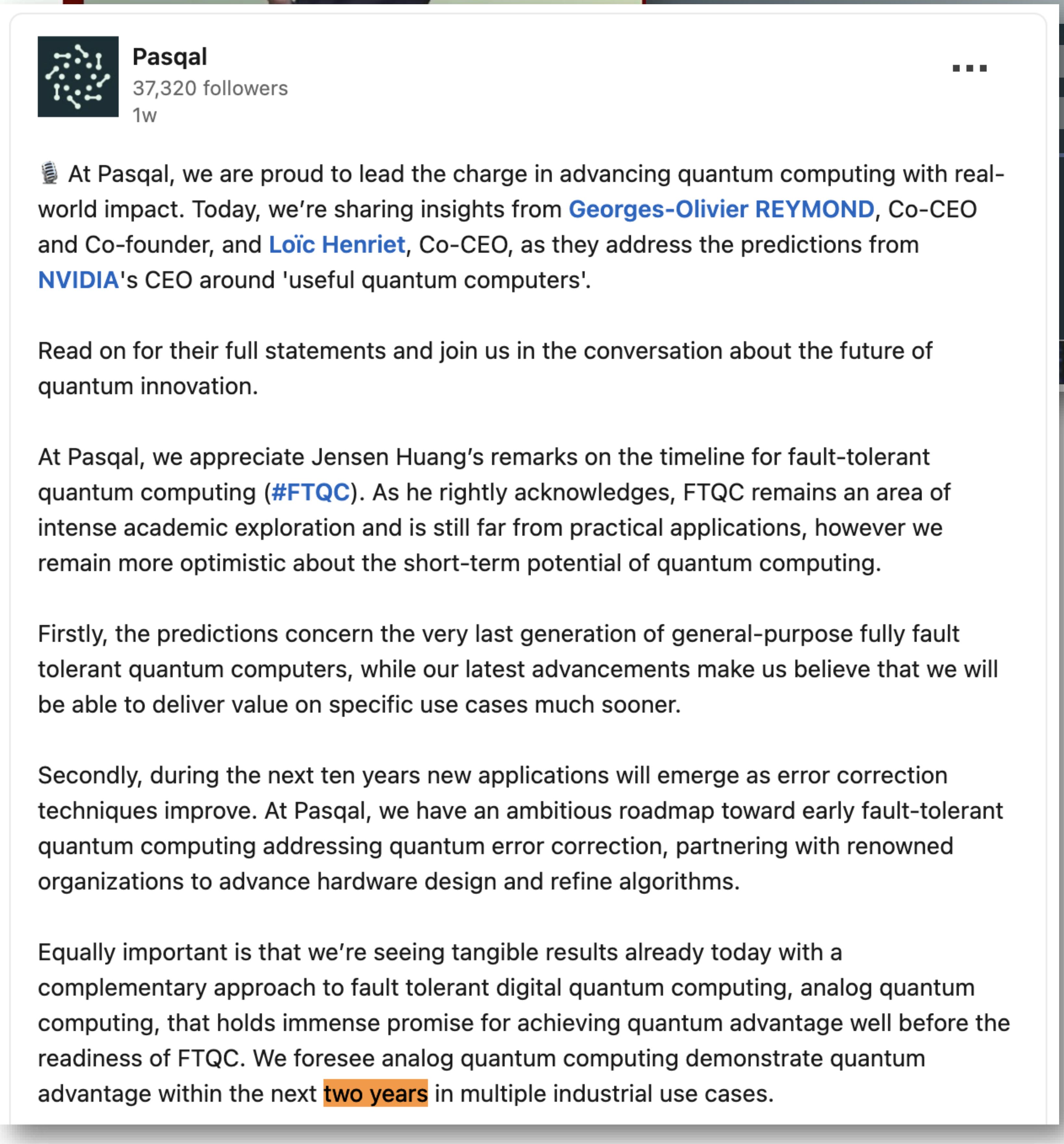
https://science.osti.gov/-/media/QIS/pdf/DOE_QIS_Roadmap_Final.pdf

“Getting ‘very useful quantum computers’ to market could take 15 to 30 years.”








Pasqal
37,320 followers
1w

At Pasqal, we are proud to lead the charge in advancing quantum computing with real-world impact. Today, we’re sharing insights from Georges-Olivier REYMOND, Co-CEO and Co-founder, and Loïc Henriet, Co-CEO, as they address the predictions from NVIDIA's CEO around 'useful quantum computers'.

Read on for their full statements and join us in the conversation about the future of quantum innovation.

At Pasqal, we appreciate Jensen Huang’s remarks on the timeline for fault-tolerant quantum computing (#FTQC). As he rightly acknowledges, FTQC remains an area of intense academic exploration and is still far from practical applications, however we remain more optimistic about the short-term potential of quantum computing.

Firstly, the predictions concern the very last generation of general-purpose fully fault tolerant quantum computers, while our latest advancements make us believe that we will be able to deliver value on specific use cases much sooner.

Secondly, during the next ten years new applications will emerge as error correction techniques improve. At Pasqal, we have an ambitious roadmap toward early fault-tolerant quantum computing addressing quantum error correction, partnering with renowned organizations to advance hardware design and refine algorithms.

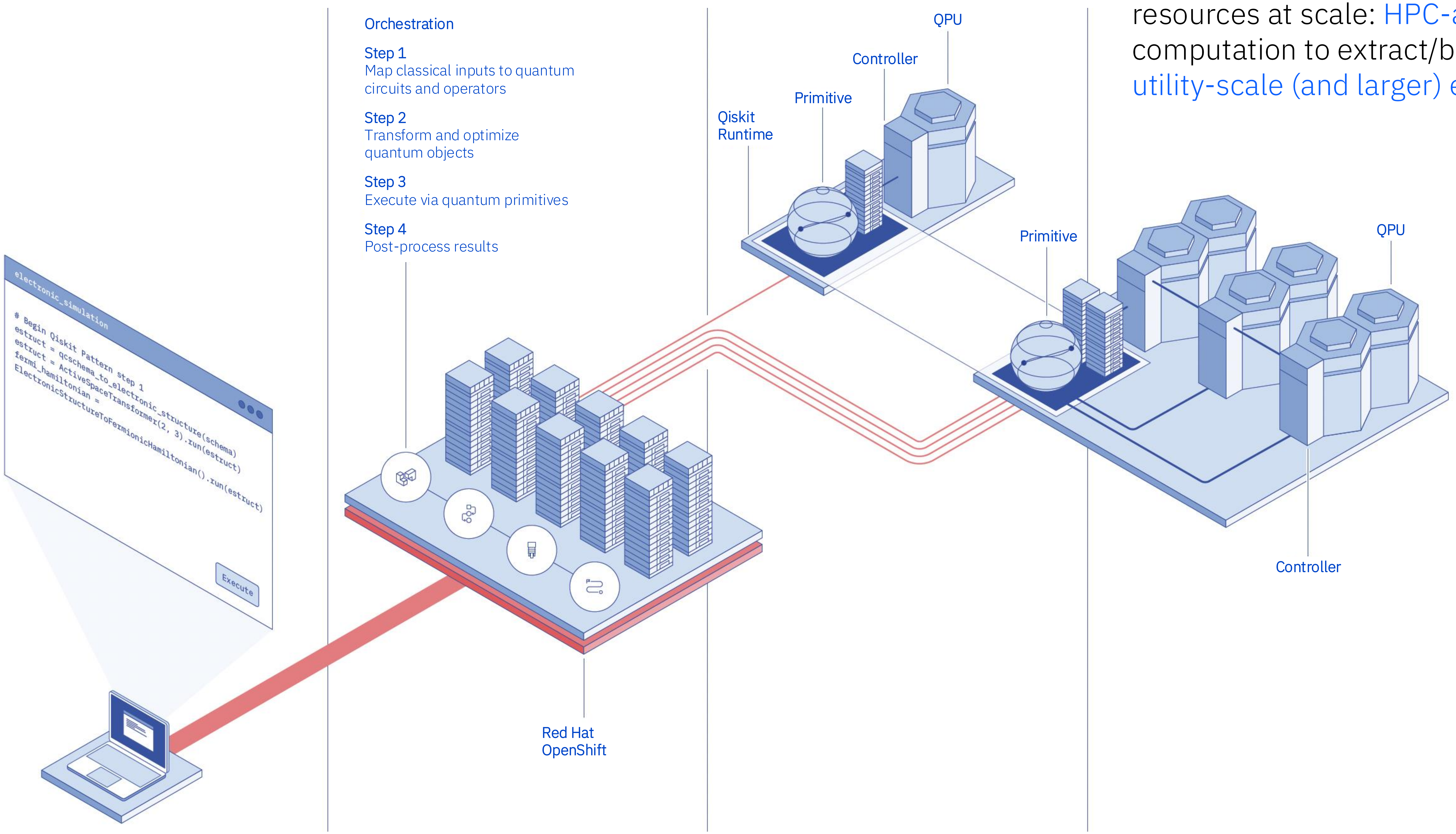
Equally important is that we’re seeing tangible results already today with a complementary approach to fault tolerant digital quantum computing, analog quantum computing, that holds immense promise for achieving quantum advantage well before the readiness of FTQC. We foresee analog quantum computing demonstrate quantum advantage within the next two years in multiple industrial use cases.

https://science.osti.gov/-/media/QIS/pdf/DOE

Quantum-centric Supercomputing (QCSC)

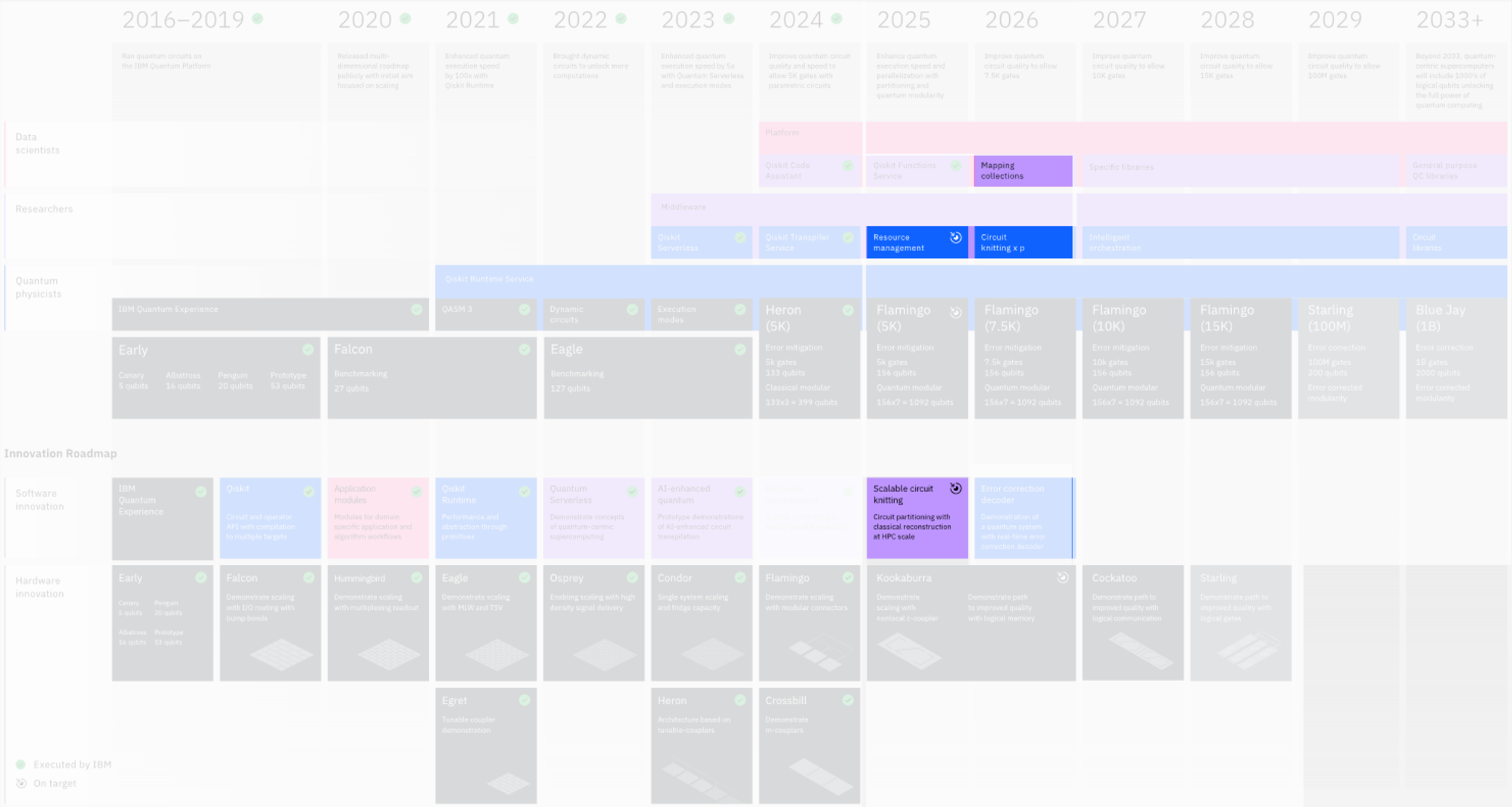
arXiv:2405.05068
arXiv:2501.09702

Delivering impactful quantum computing requires the interplay of quantum and classical resources at scale: HPC-assisted quantum computation to extract/boost useful signals in utility-scale (and larger) experiments.



Development Roadmap

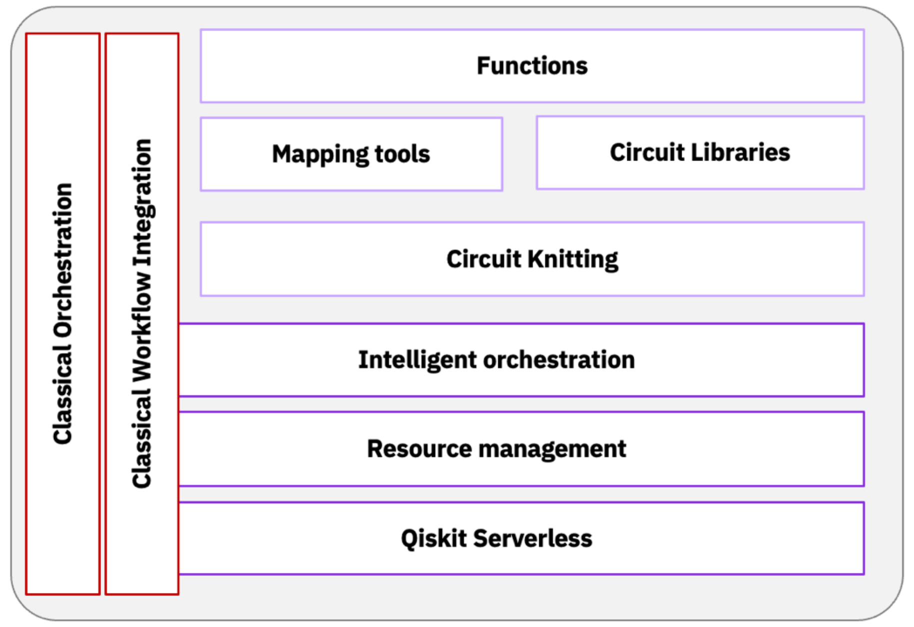
IBM Quantum

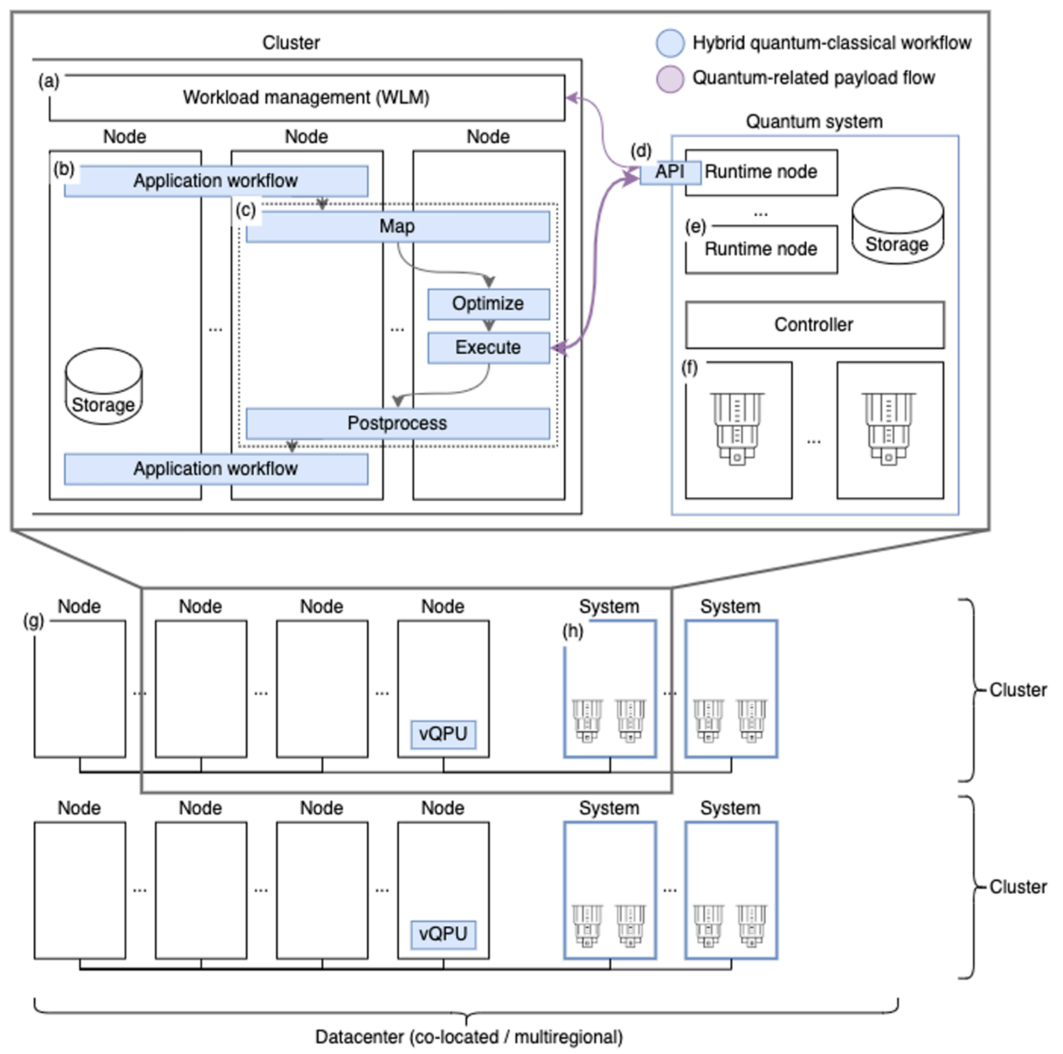


Workload Management

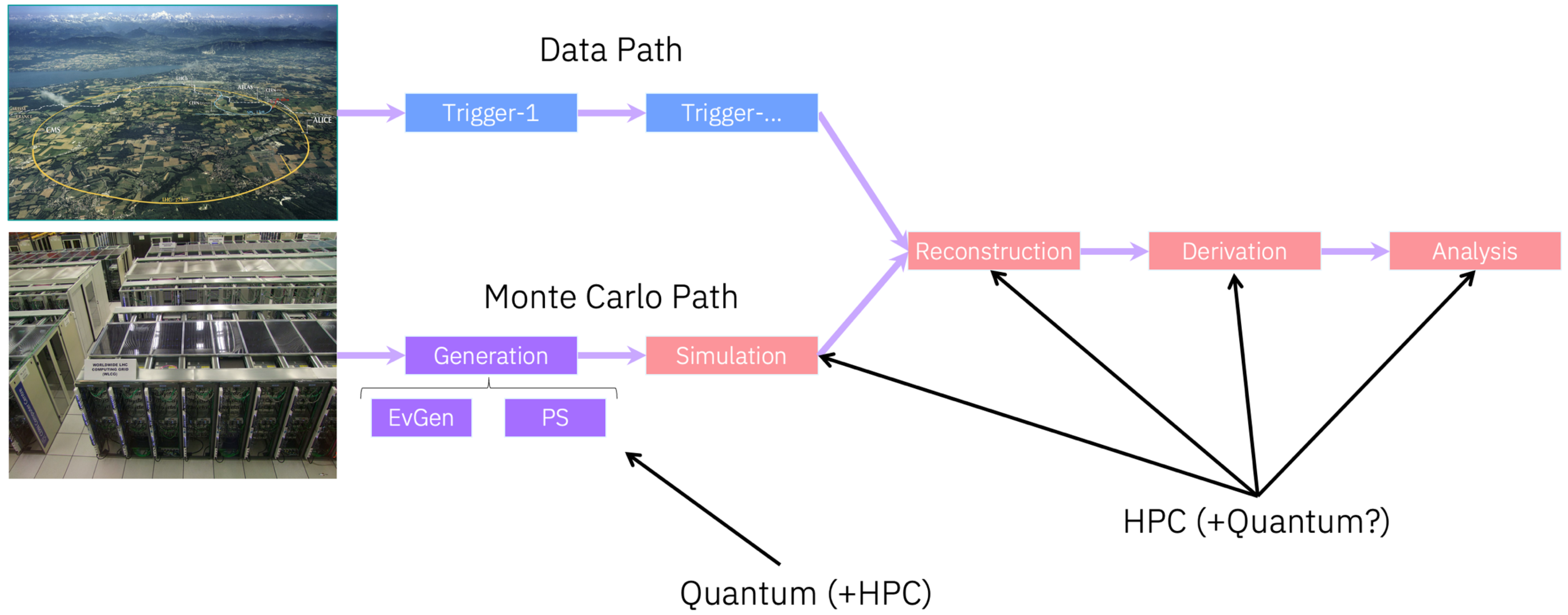
(a) - WLM
(b) - Application workflow
(c) - Qiskit Pattern
(d) - Direct API endpoint
(e) - Classical system node
(f) - QPU
(g) - Classical node
(h) - Quantum system

Quantum and Classical Middleware





Hybrid classical—quantum workflows



Programmability

Near-term

"Loosely-coupled" regime

Trivially separable classical and quantum executables, functions, etc.; latencies insignificant

Language-agnostic: "bag of tasks" to be executed on available resources; e.g., C/C++, Python (Qiskit) for quantum and AI/ML

Far-term

"Tightly-coupled" regime

Desire is to offload quantum-accelerated kernels while classical task(s) execute in parallel and/or asynchronously

Single-source application(s) part of a larger workflow that pervade all available resources; compiled language (C/C++), potentially with bindings

How will these systems be programmed?

arXiv:2405.05068

Sample-based Quantum Diagonalization

(aka, "HPC Estimator")

Input is a set of noisy samples, $\tilde{\mathcal{X}} = \{\mathbf{x} \mid \mathbf{x} \sim \tilde{P}_{\Psi}(\mathbf{x})\}$, generated by a quantum processor.

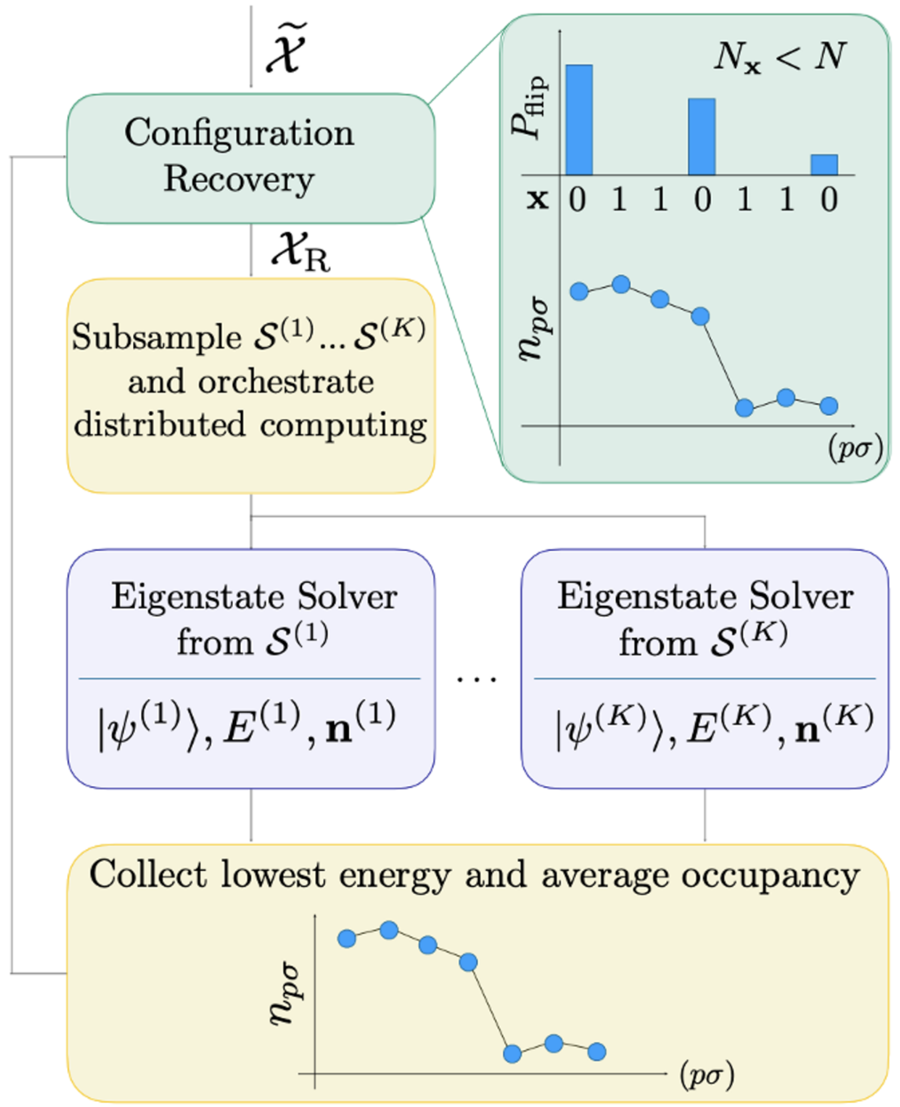
If particle number is not conserved, i.e., $N_x \neq N$, $|N_x - N|$, bits are flipped probabilistically to generate a new set $\tilde{\mathcal{X}}_R$ of "recovered configurations."

Build K batches of d configurations, $\mathcal{S}^{(1)}, \ldots, \mathcal{S}^{(K)}$, from $\tilde{\mathcal{X}}_R$, distributed according to the frequencies of $\mathbf{x} \in \tilde{\mathcal{X}}_R$.

The Hamiltonian is projected and diagonalized over sub-samples from $\tilde{\mathcal{X}}_R$. In the largest experiments, using 6400 nodes, diagonalization took about 1.5 hrs.

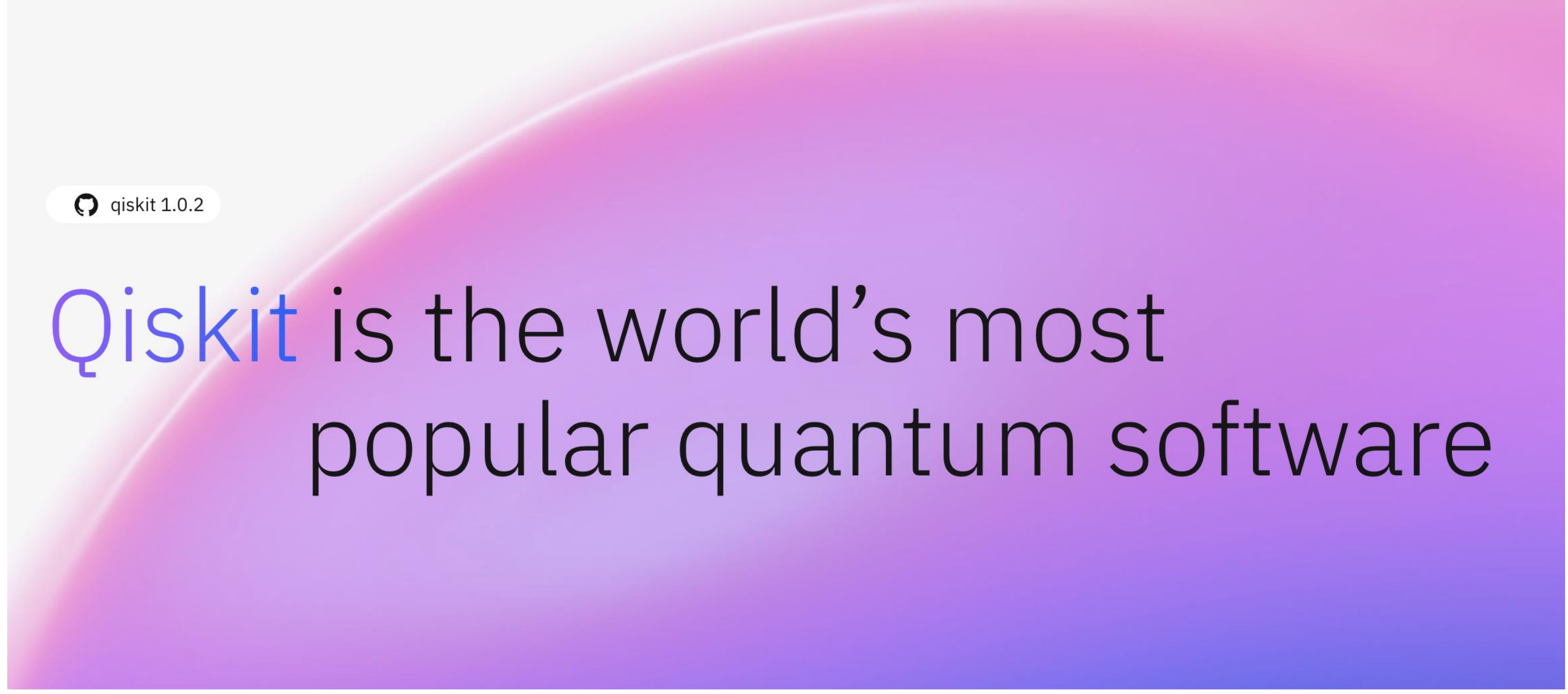
Ground states and energies are computed from $\hat{H}_{\mathcal{S}^{(k)}}$ and averaged over, which are used to calculate updated occupancies, $\mathbf{n}$.

Repeat until some stopping criteria is met.



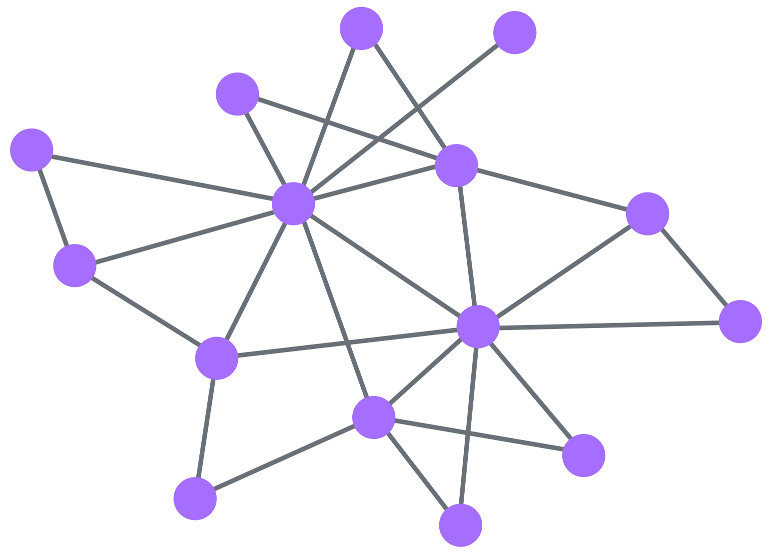
Qiskit

pip install qiskit
pip install qiskit-ibm-runtime



Step 1

Map classical inputs to a quantum problem.

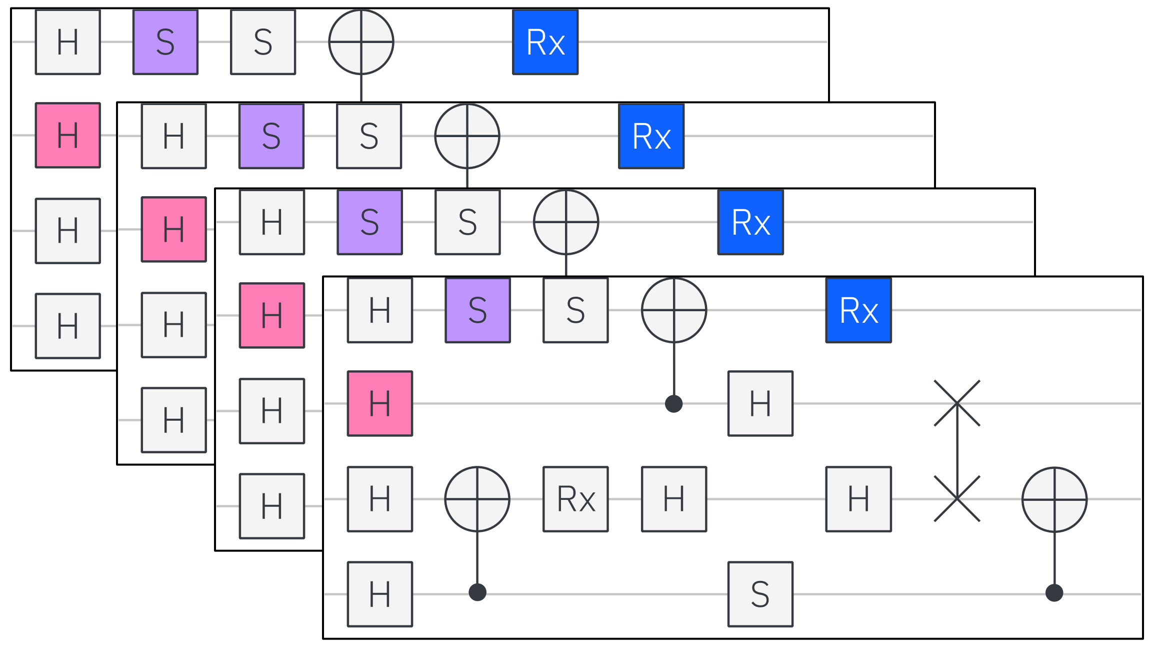


Circuit library

Quantum info library

Step 2

Translate problem for optimized quantum execution.

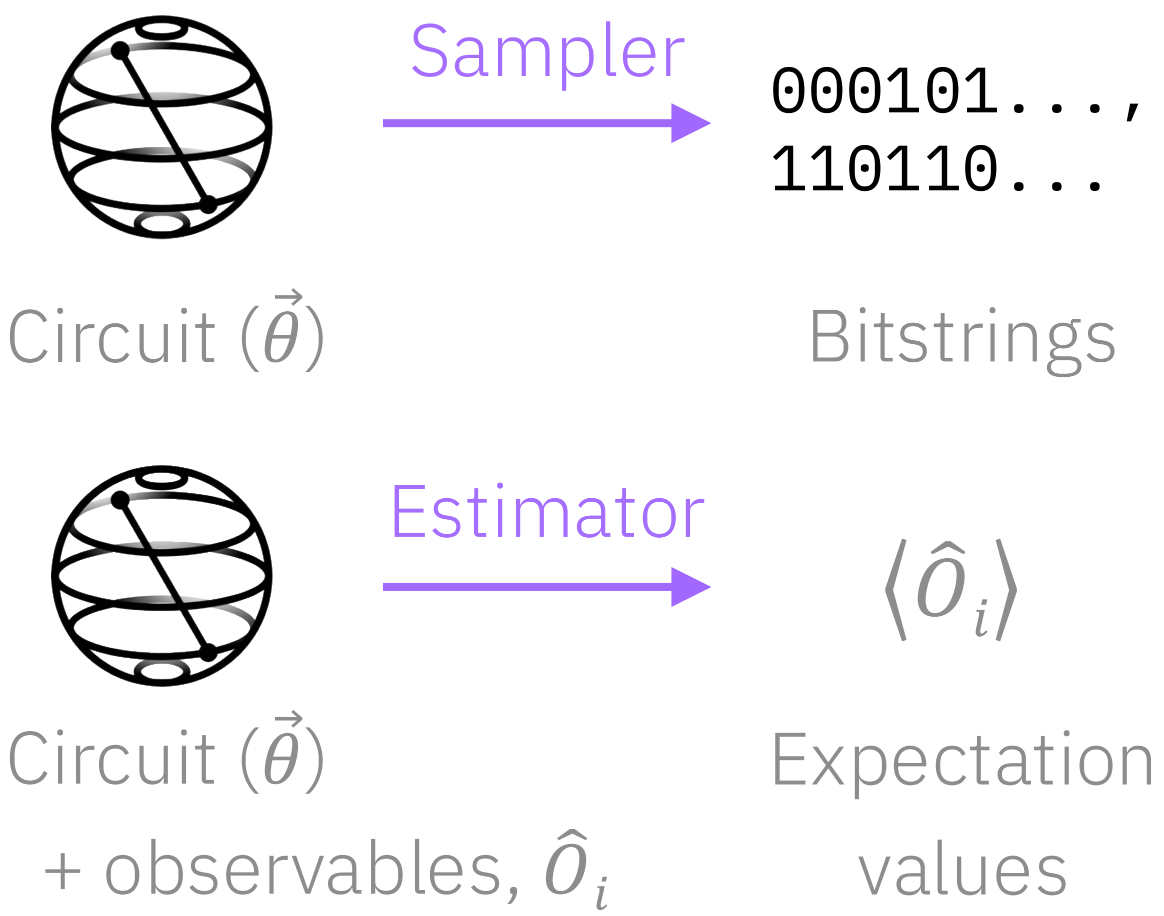


Transpiler

AI-enhanced transpiler

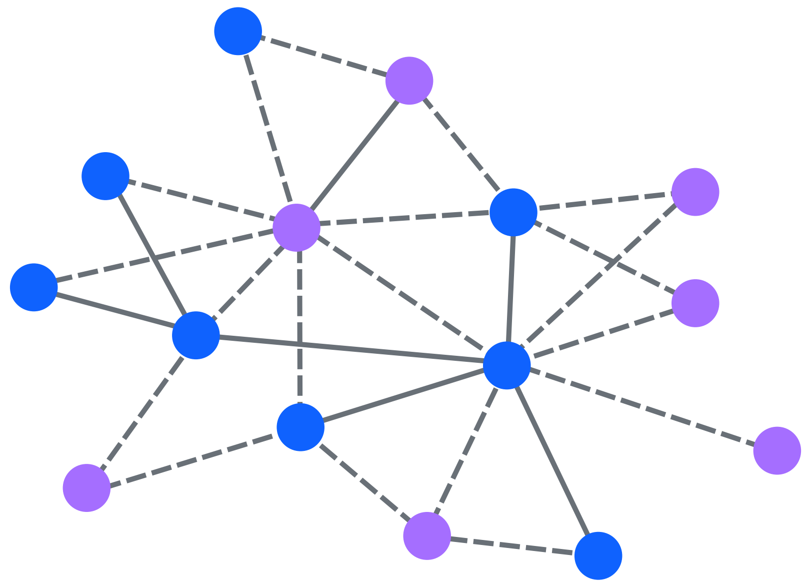
Step 3

Execute using Qiskit Runtime Primitives.



Step 4

Post-process, return results in classical format.

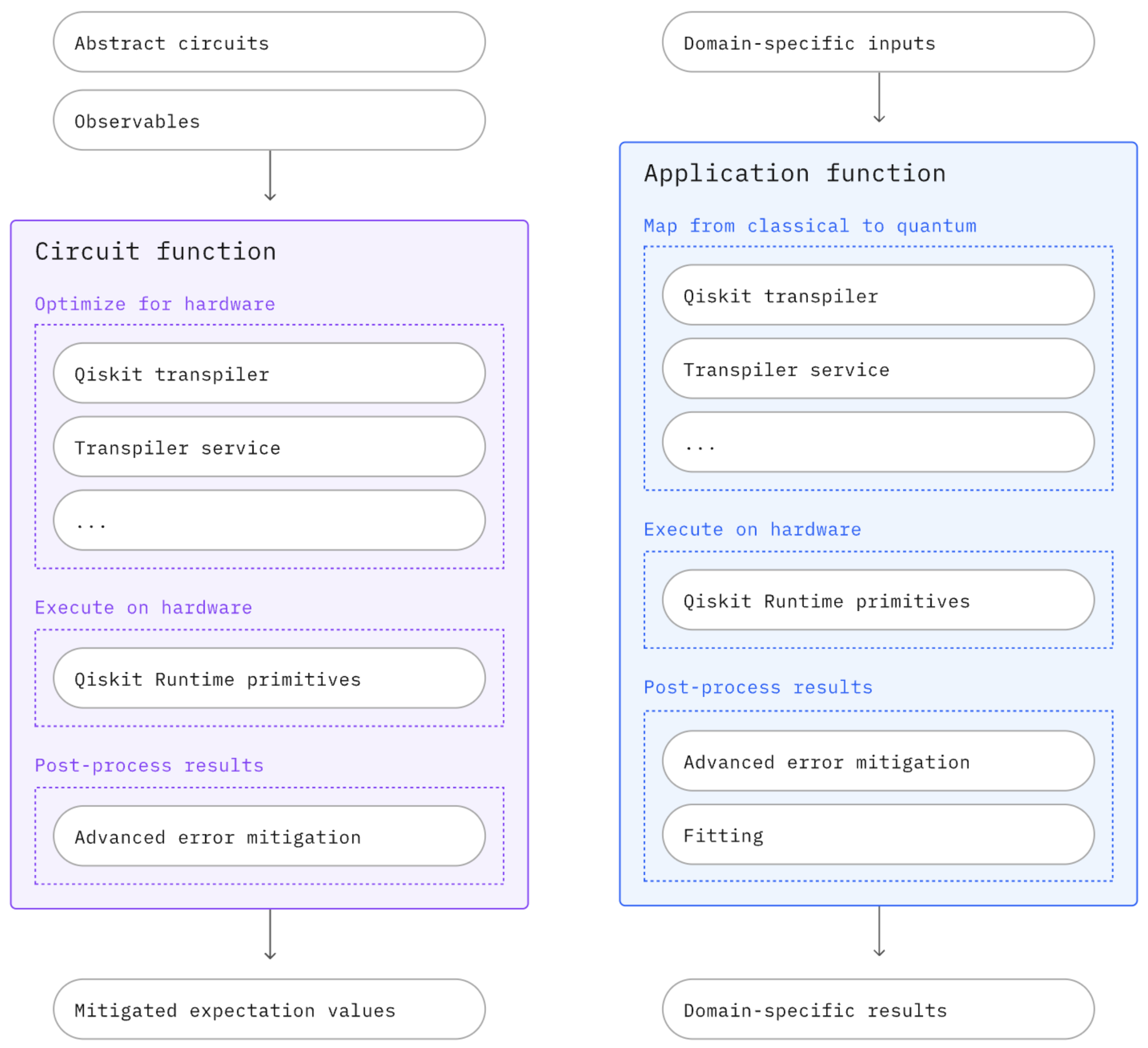


Quantum info library

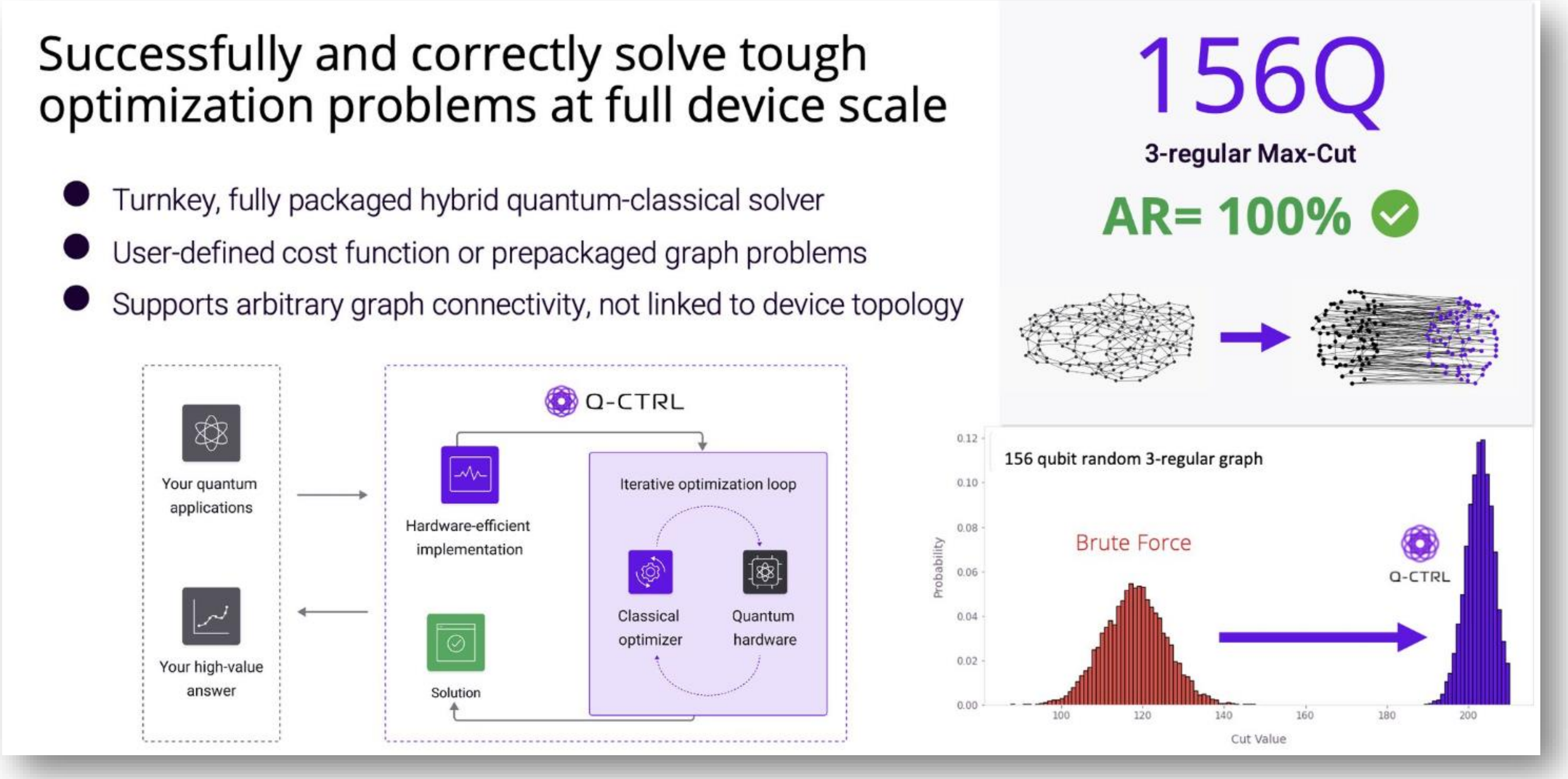
Visualization module

Q-CTRL

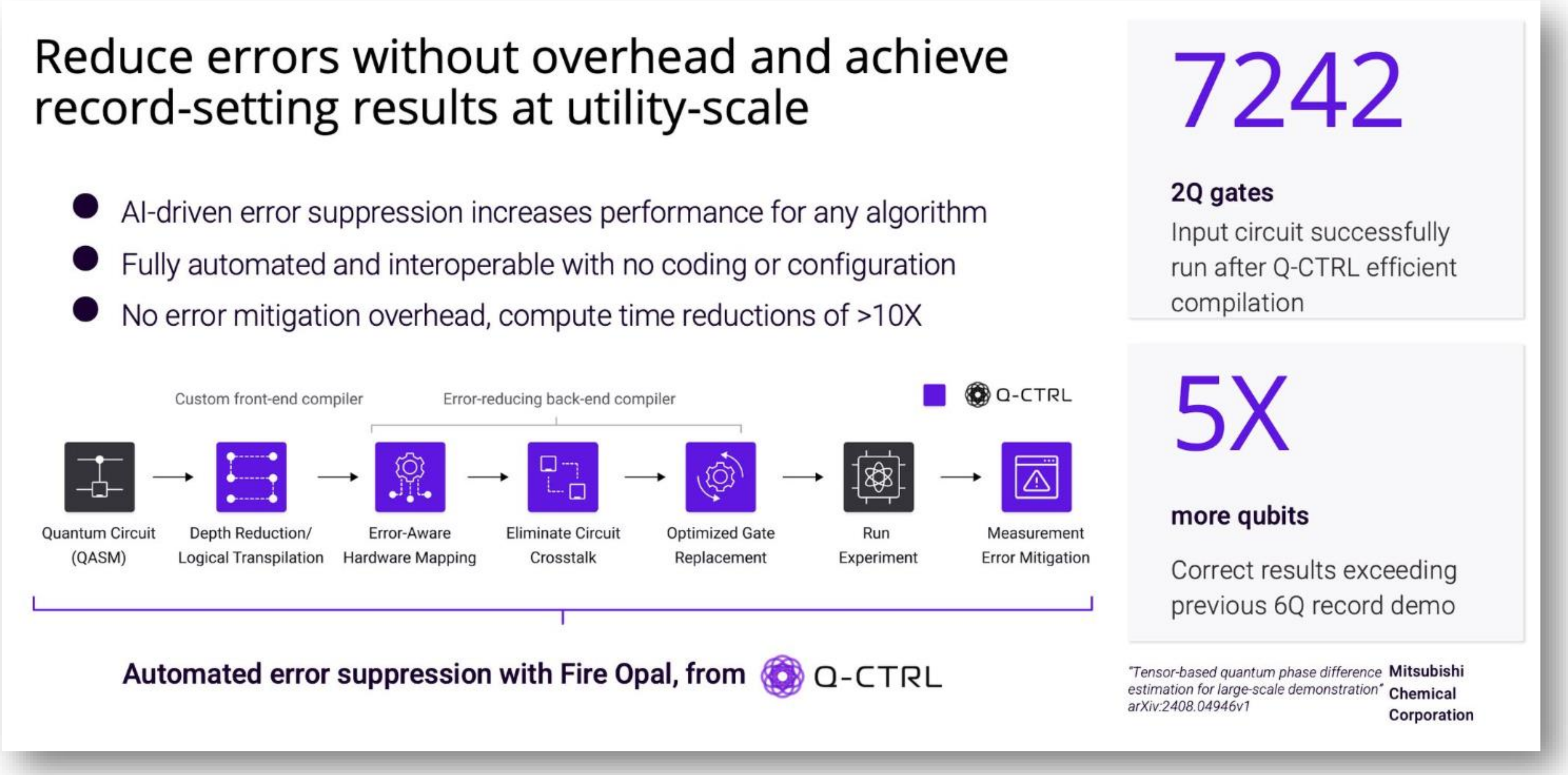
Simplify and accelerate utility-scale algorithm discovery and application development abstracting away parts of the quantum software development workflow.



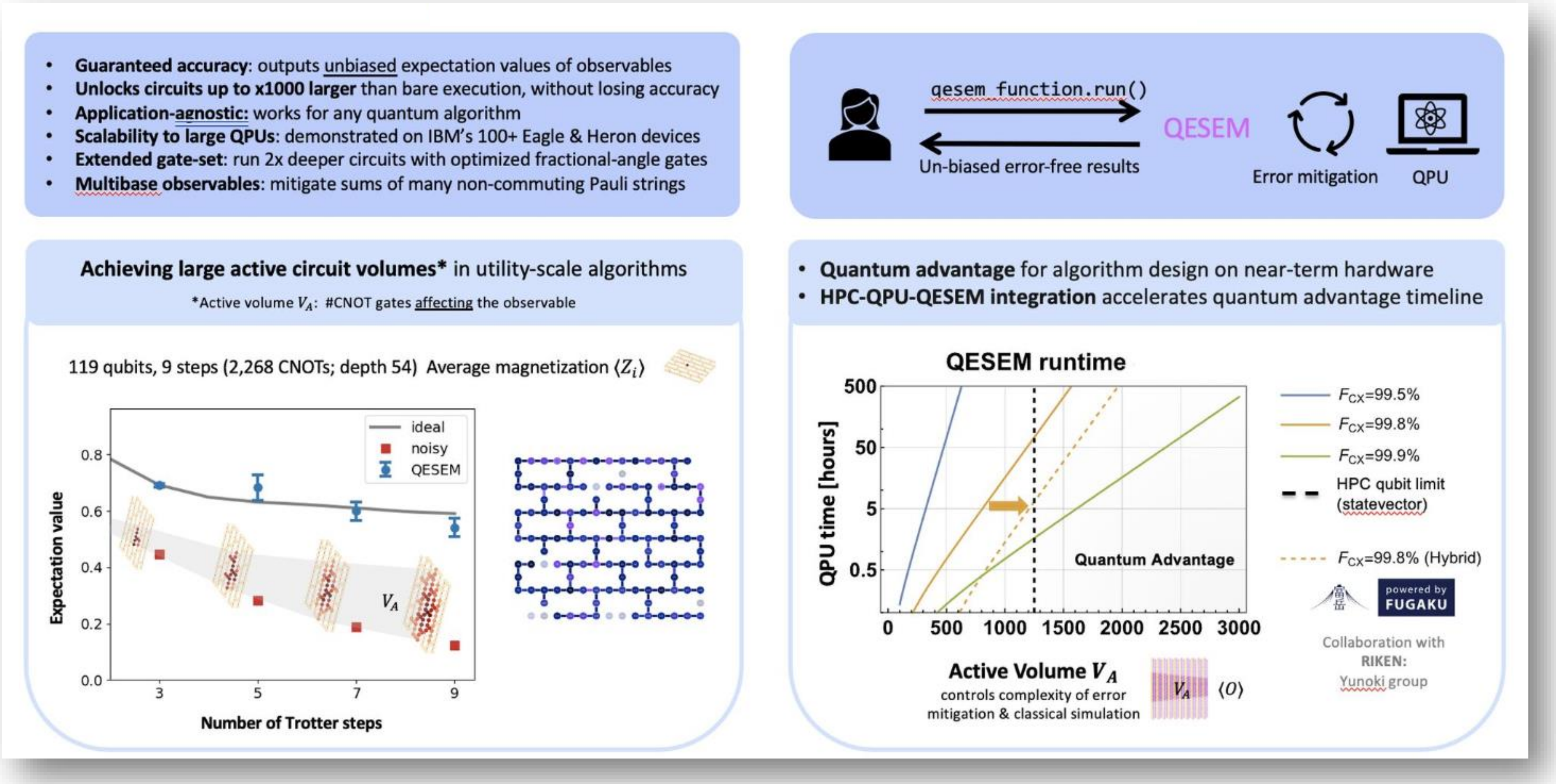
Optimization Solver



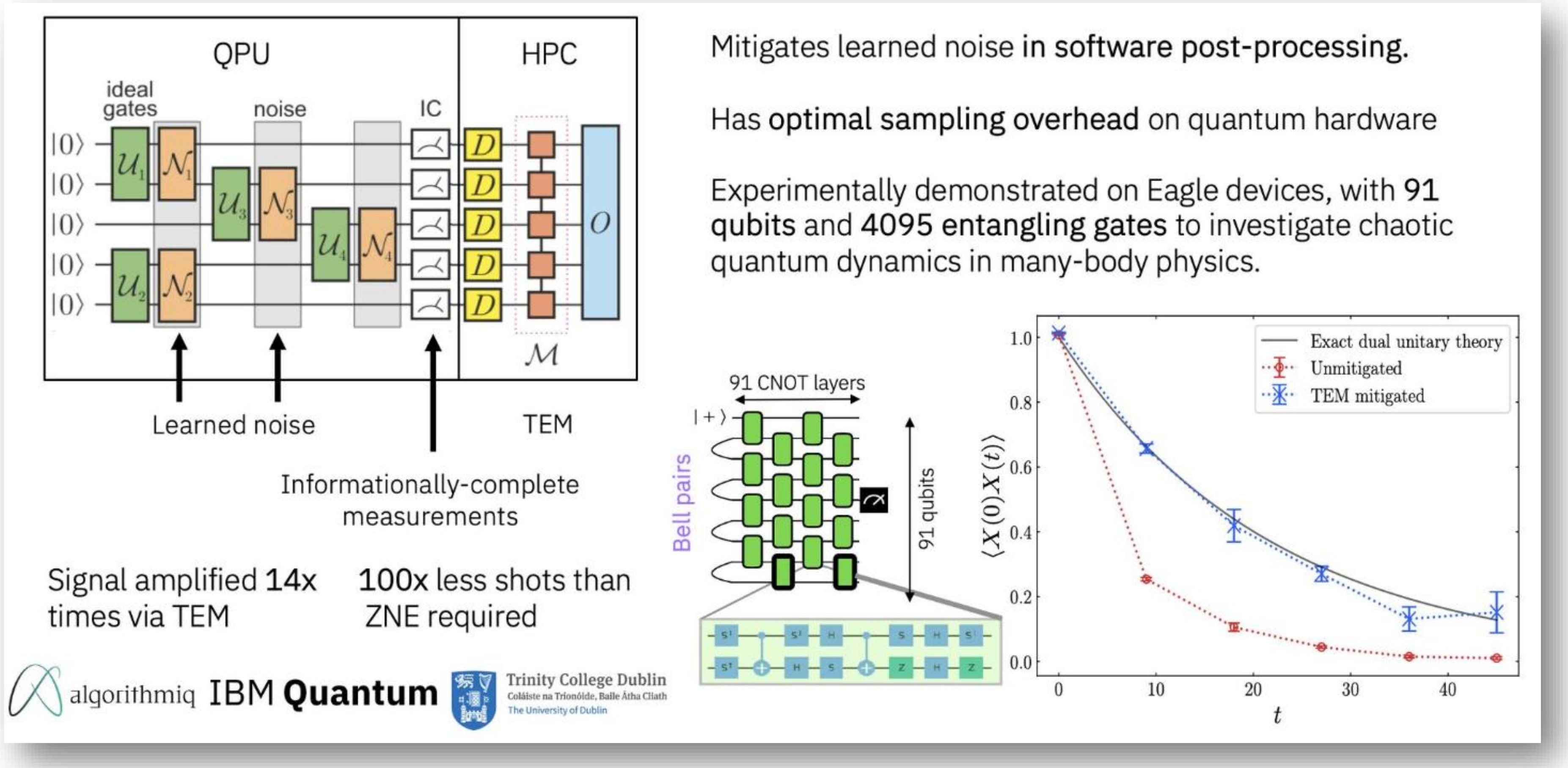
Performance Management



Quantum Error Suppression & Error Mitigation (QESEM, Qedma)



Tensor-network error mitigation (TEM)



Approximate quantum compilation with tensor networks (aqc-tensor) Enables users to compile the initial portion of a circuit into a nearly equivalent approximation of that circuit, but with much fewer layers.

Multi-product formulas (MPF)

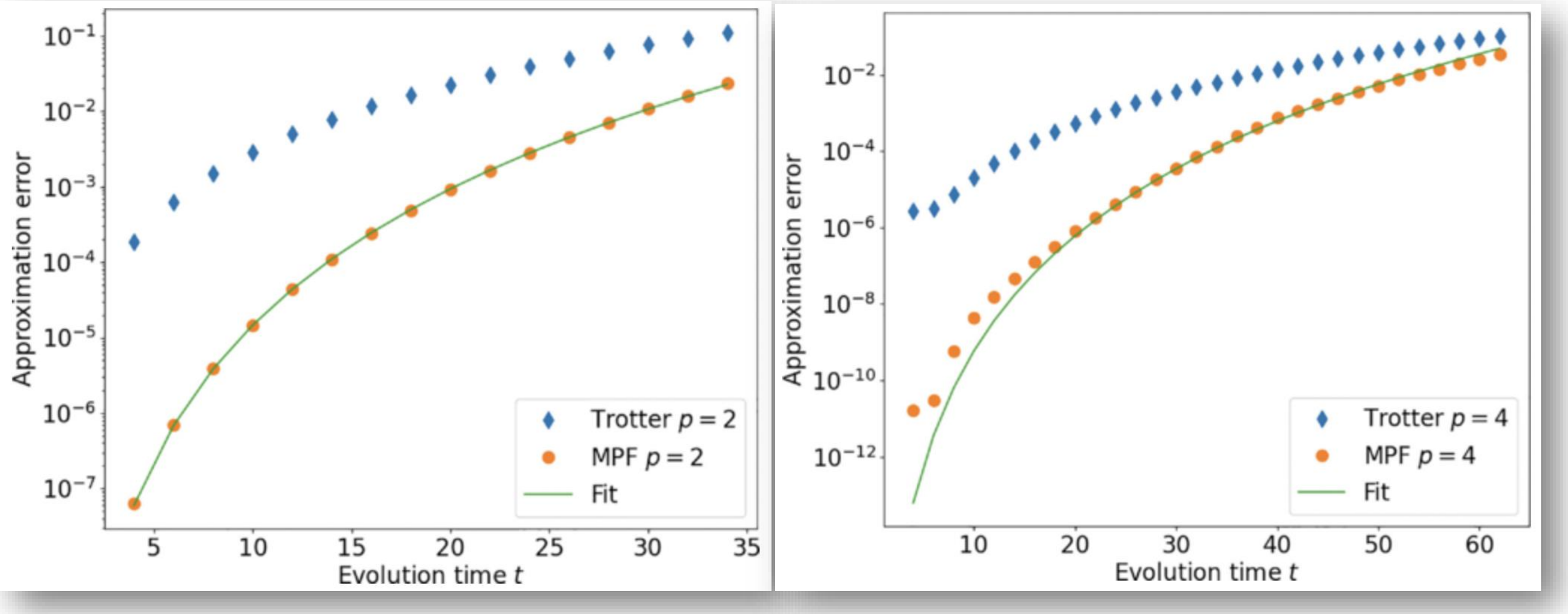
Can be used to reduce the Trotter error of Hamiltonian dynamics. [Quantum 7, 1067 (2023), Phys. Rev. Research 6, 033309 (2024)]

Circuit cutting (CC) Gates and/or wires are cut, resulting in smaller circuits. The original circuit can then be reconstructed at the cost of an exponential number of shots in the number of cuts.

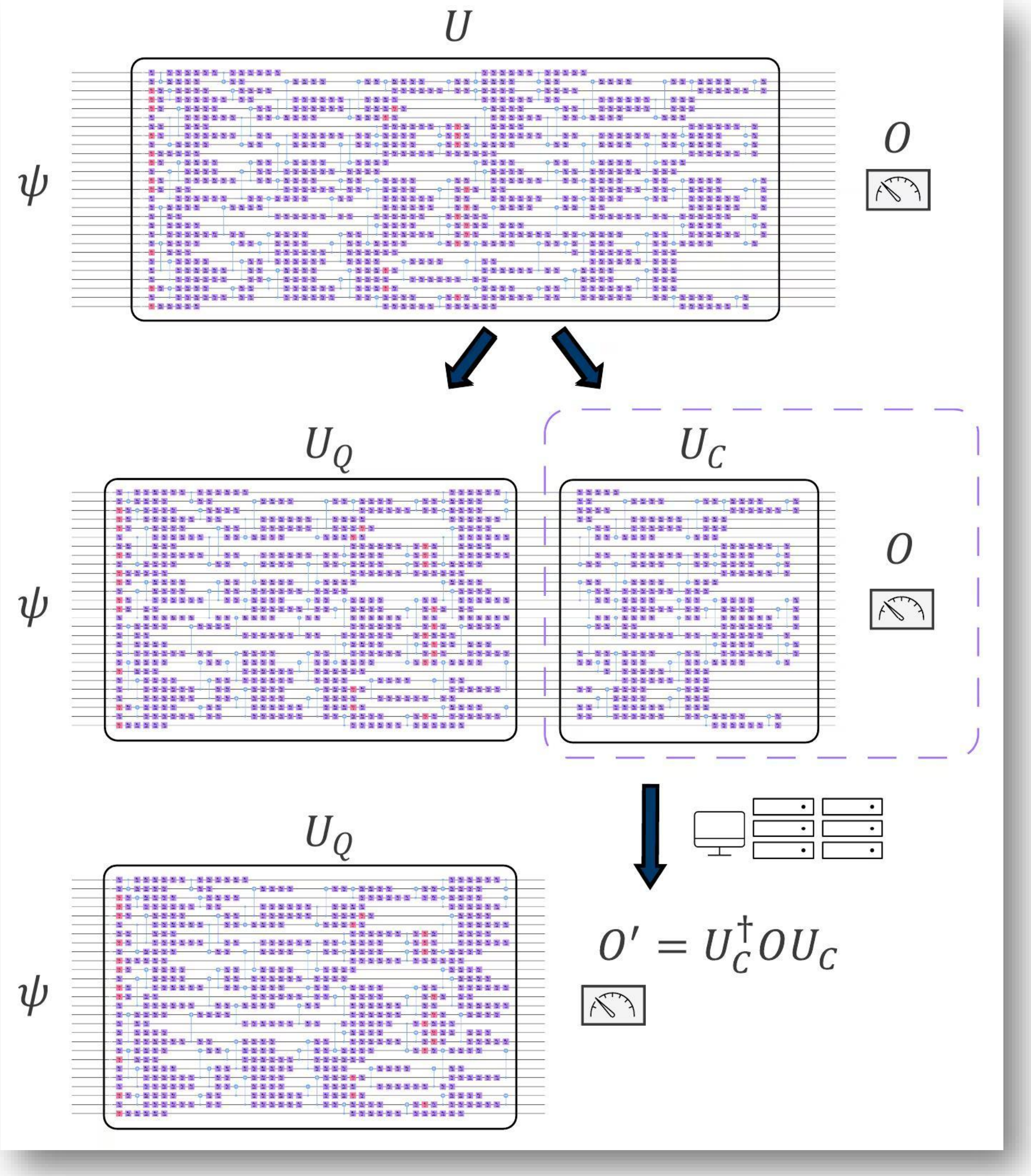
Operator backpropagation (OBP) Reduce circuit depth by absorbing/trimming operators at the end of a circuit at the cost of more terms in the observable.

aqc-tensor

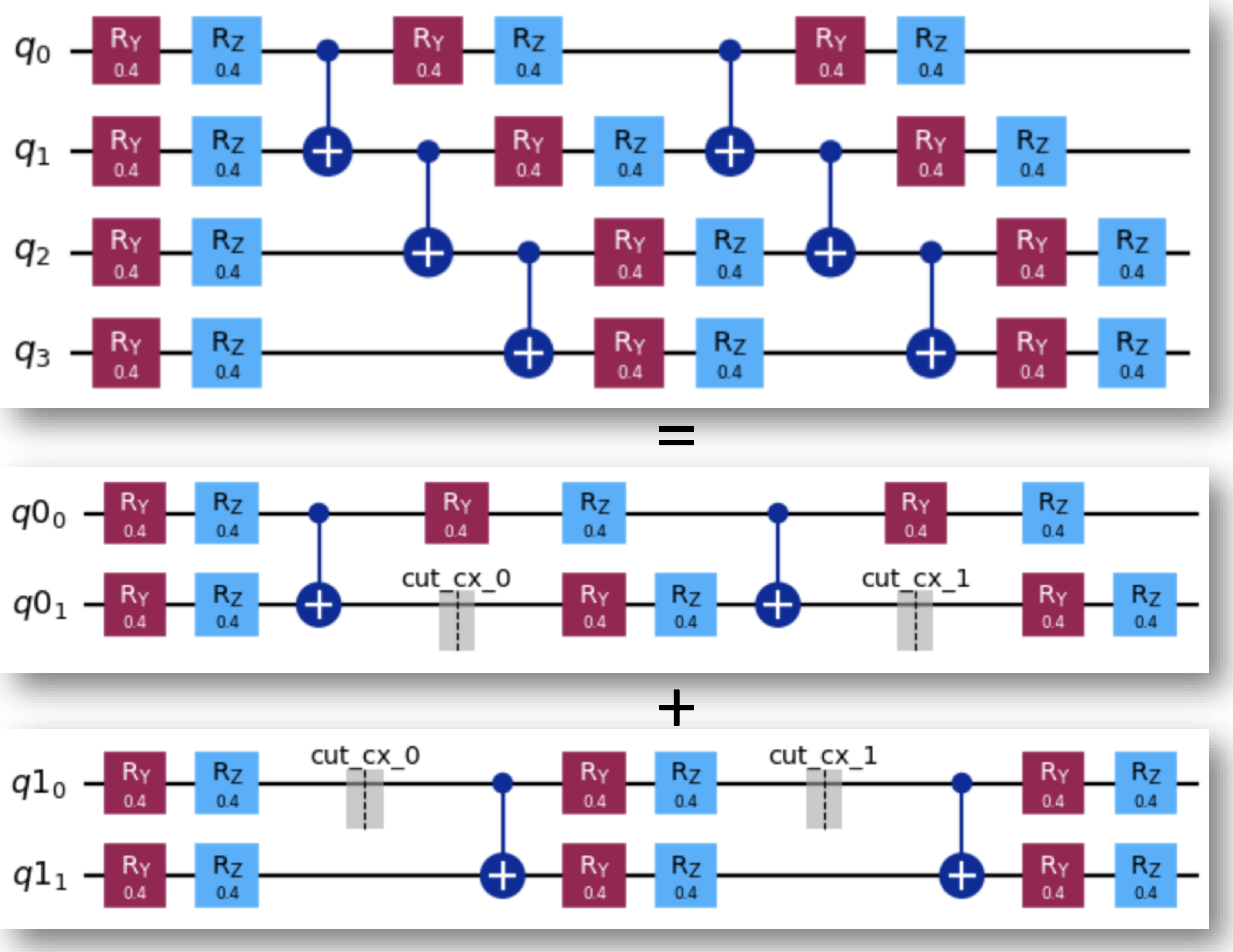
MPF



OBP



CC



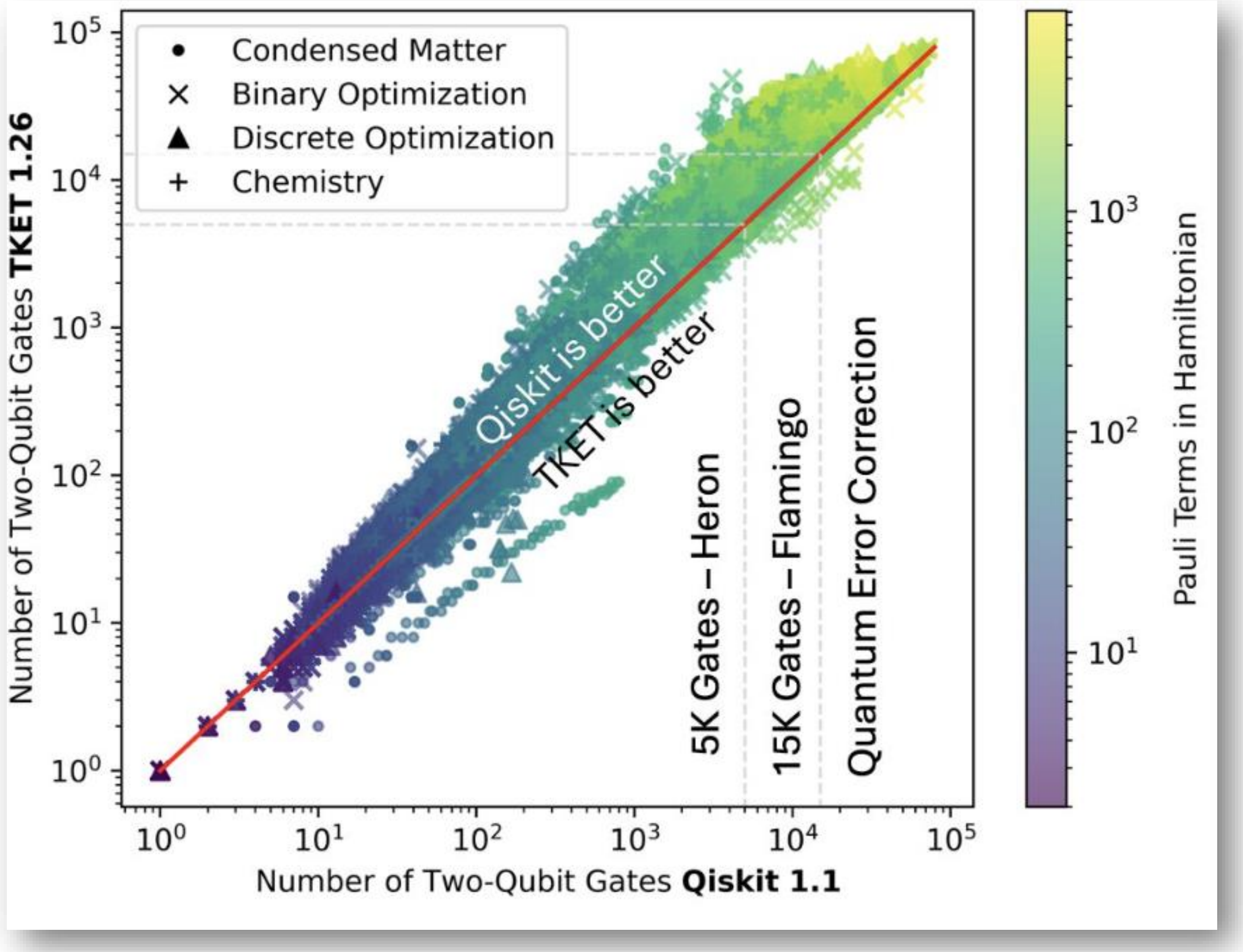
Qiskit performance

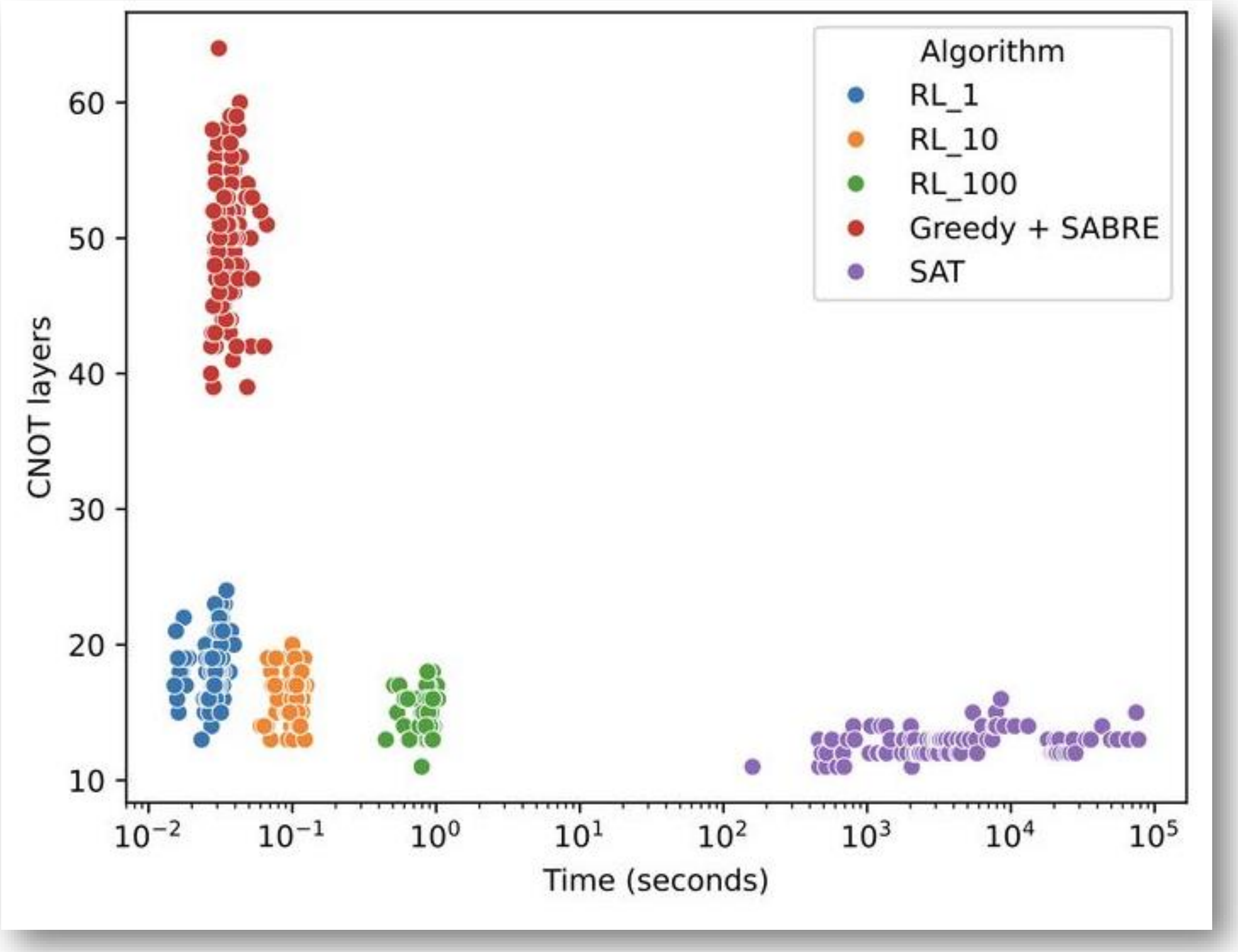
Based on Hamlib's [arXiv:2306.13126] roughly 860k circuits, **Qiskit outperforms** next best transpiler >86% of the time.

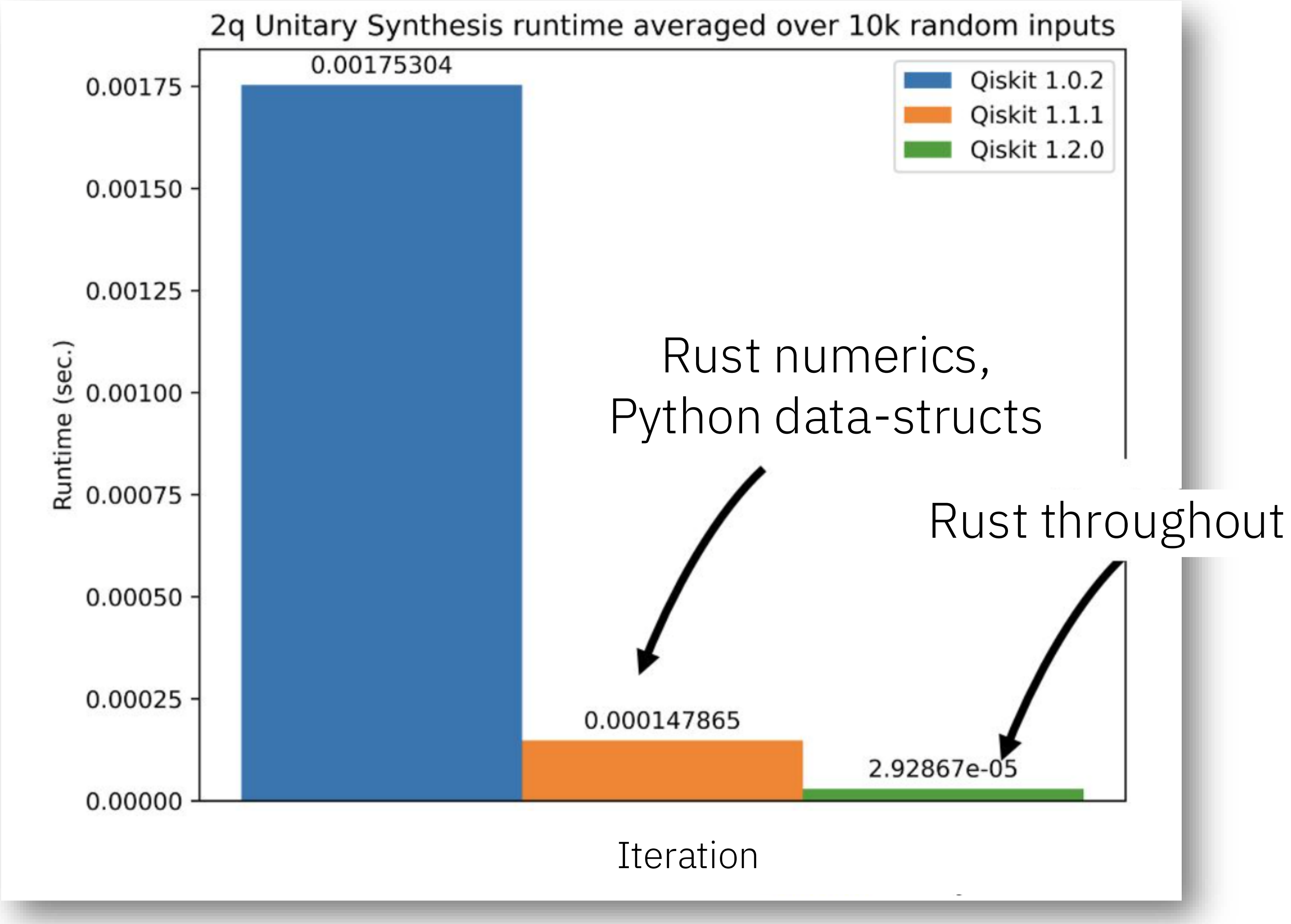
AI-enhanced (synthesis+routing) transpilation achieves a good balance between optimality and runtime, 24(36)% 2-qubit gate count(depth). [arXiv:2405.13196]

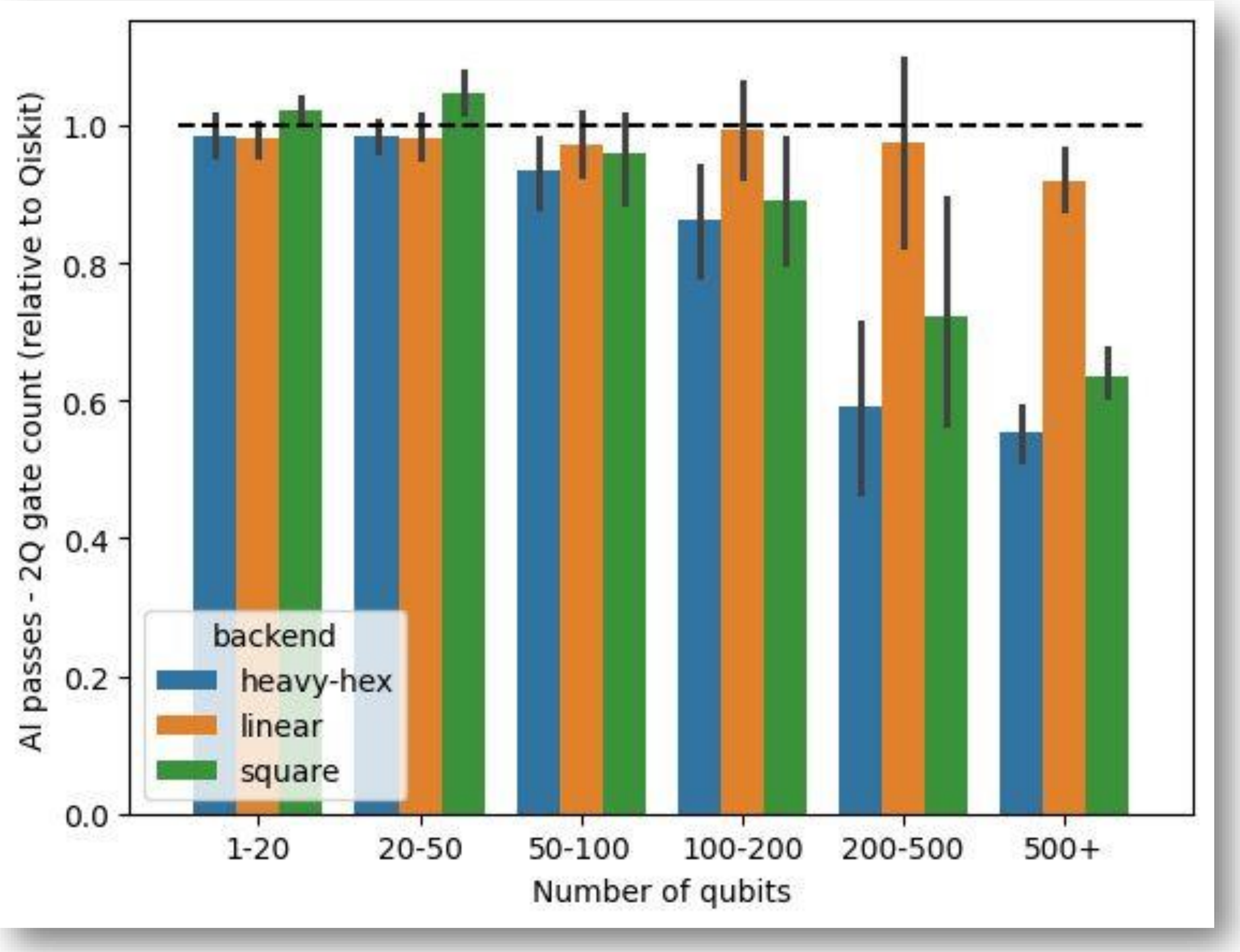
Oxidization of Qiskit improves overall performance by two orders of magnitude.

* **Core Qiskit components** also interfacing with C/C++ for HPC applications (let's talk!).









Coming together to solve domain-specific problems.

Quantum Working Groups
From utility to advantage

IBM Quantum helps facilitate working groups to bring together the best scientists across domains to accelerate our path to achieving Quantum Advantage by 2026.

Optimization

Quantum Optimization: Potential, Challenges, and the Path Forward

arXiv:2312.02279

Materials & HPC

Quantum-centric Supercomputing for Materials Science: A Perspective on Challenges and Future Directions

arXiv:2312.09733

High Energy Physics

Quantum Computing for High-Energy Physics: State of the Art and Challenges. Summary of the QC4HEP Working Group

arXiv:2307.03236

Healthcare & Life Sciences

Towards quantum-enabled cell-centric therapeutics.

arXiv:2307.05734

Sustainability

Collaborative projects in the fields of Materials and Energy leveraging quantum computers.

Quantum Algorithm Engineering (QAE)

Guidance, best practices, benchmarking, and collaboration

1. Project design and mapping

1. Strategic positioning
2. Scaling tactics and evaluation
3. Qiskit patterns
4. Approx. resource estimation
5. Algorithm design (e.g., subspace methods)

2. Workload optimization

1. Circuit transpilation
2. Circuit cutting and LOCC
3. Operator backpropagation (OBP)
4. Multiproduct formulas (MPF)
5. Dynamic circuits (DC)

3. Error suppression and mitigation

1. Error suppression:
   - Dynamical decoupling
   - Twirling
2. Error mitigation:
   - TREX
   - ZNE (Folding and PEA)

4. Validation and benchmarking

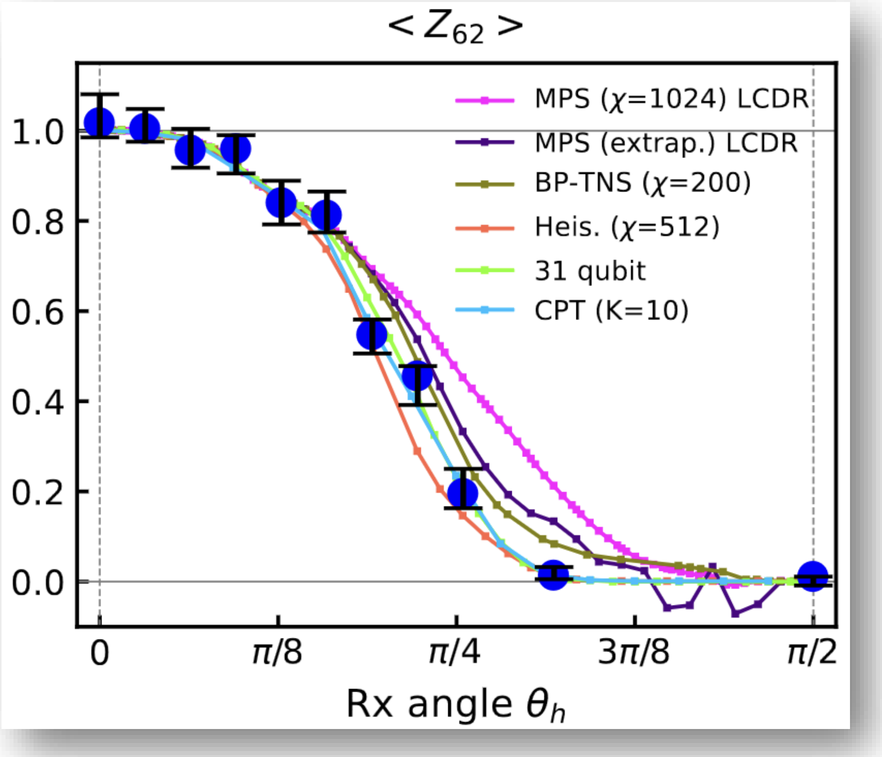
1. Result validation:
   - Circuit cliffordization
   - Statistical emulation
2. Benchmarking:
   - RB, EPG and EPLG
   - Pauli-Lindblad noise

5. Quantum-centric supercomputing (QCSC)

1. Middleware and building blocks
2. Quantum serverless and high-performance computing integration (HPC)
3. Large throughput post-selection and configuration recovery

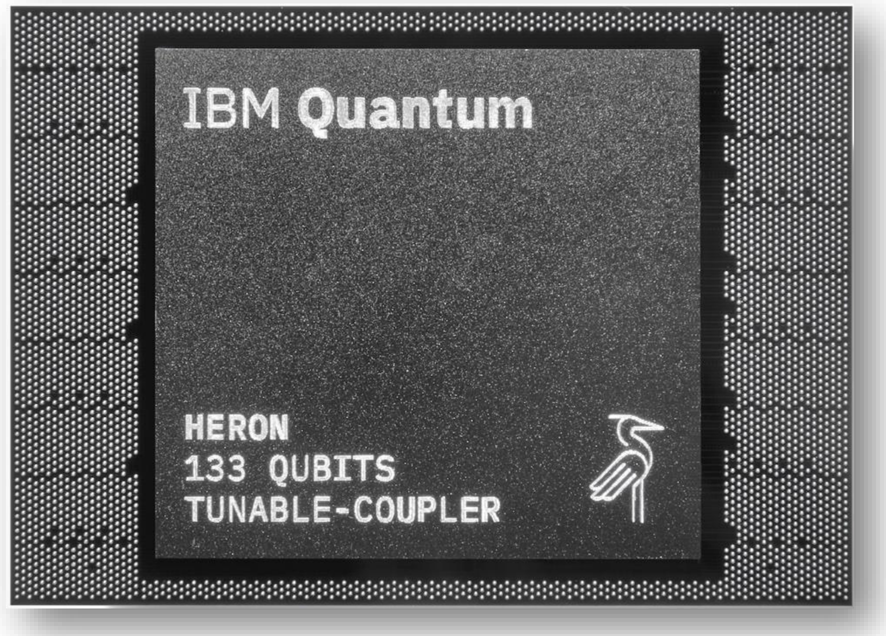
6. Frontiers of quantum computing

1. Latest methods in quantum computing research
2. New technologies and software product capabilities
3. Strategic alignment with most recent advancements in the field



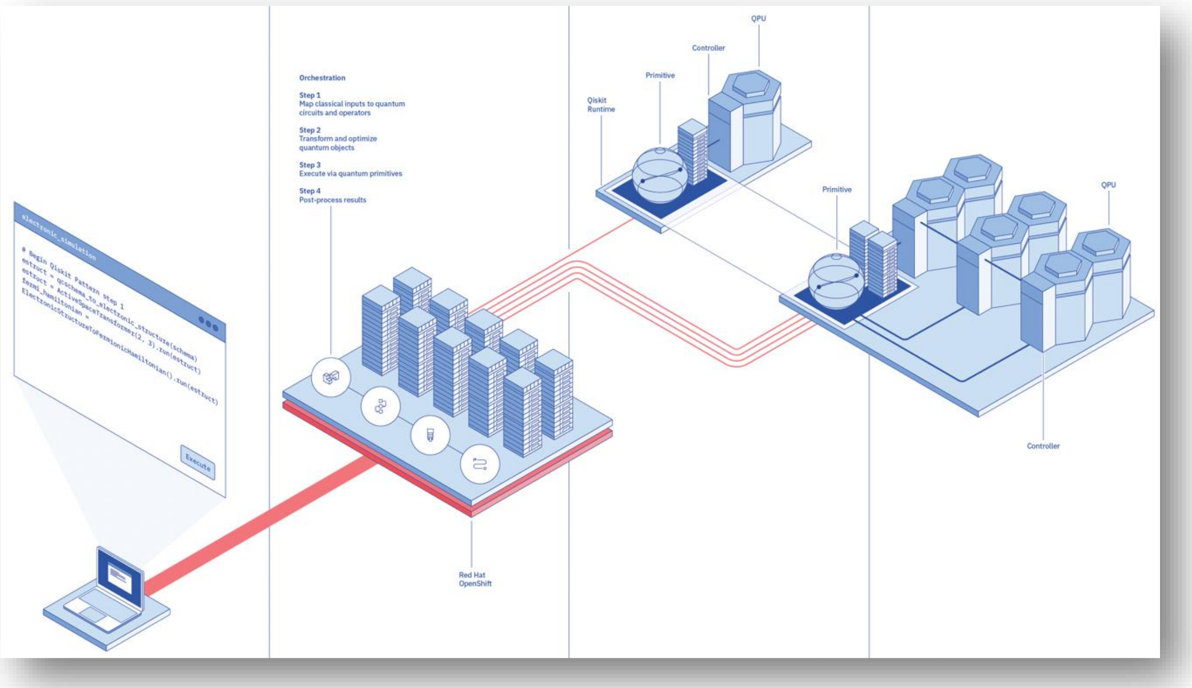
Era of Utility

Using today's quantum computers serve as tools to explore calculations beyond brute-force classical simulation.



Pushing the limits of quantum

Continual developments and innovations will drive quantum computing forward. Cross-cutting expertise from industry and academia aim to find and solve difficult problems.

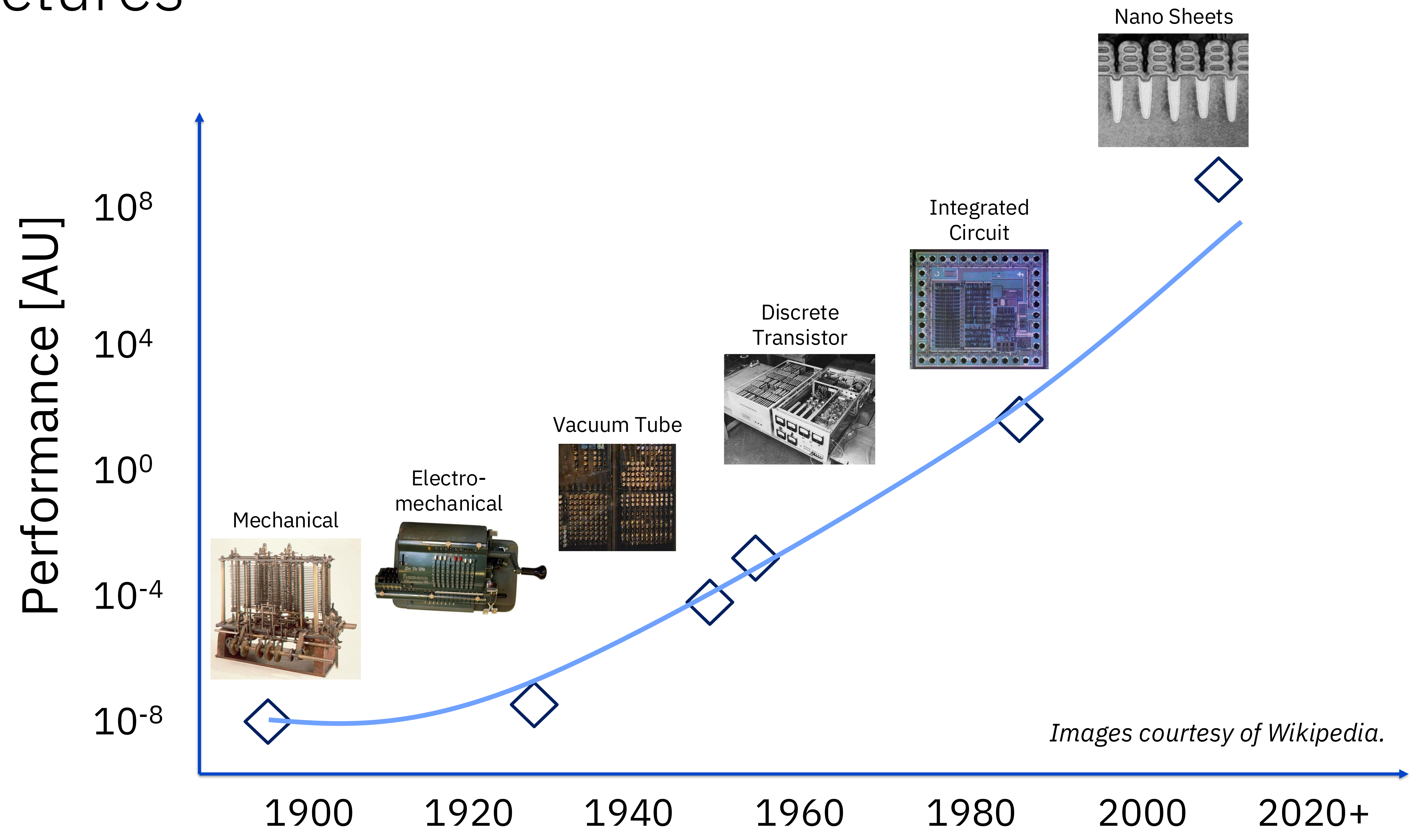


Quantum-centric Supercomputing

Architectures that can provide computational capabilities to address problems too difficult for isolated machines.

IBM

Evolution of computer architectures

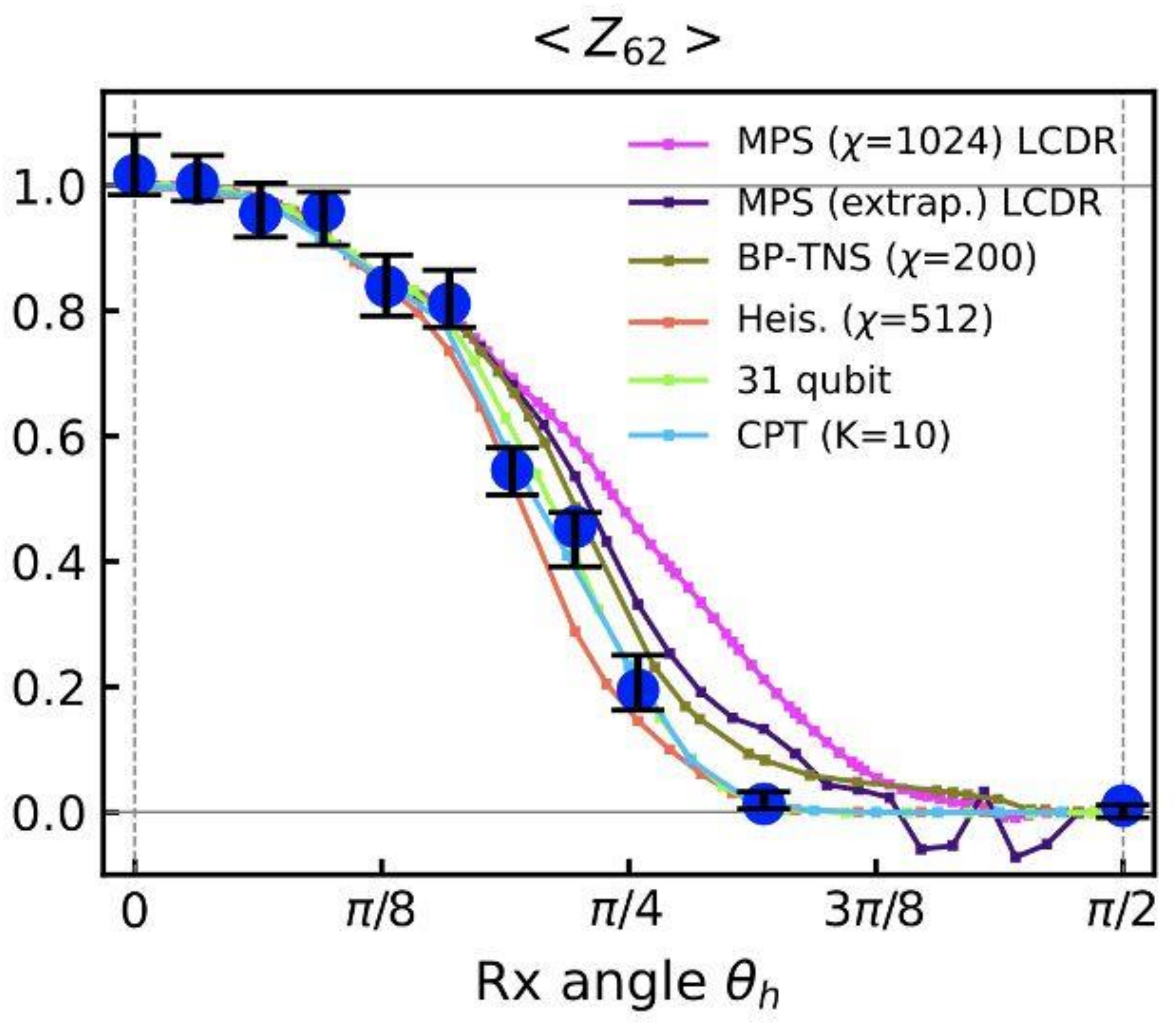


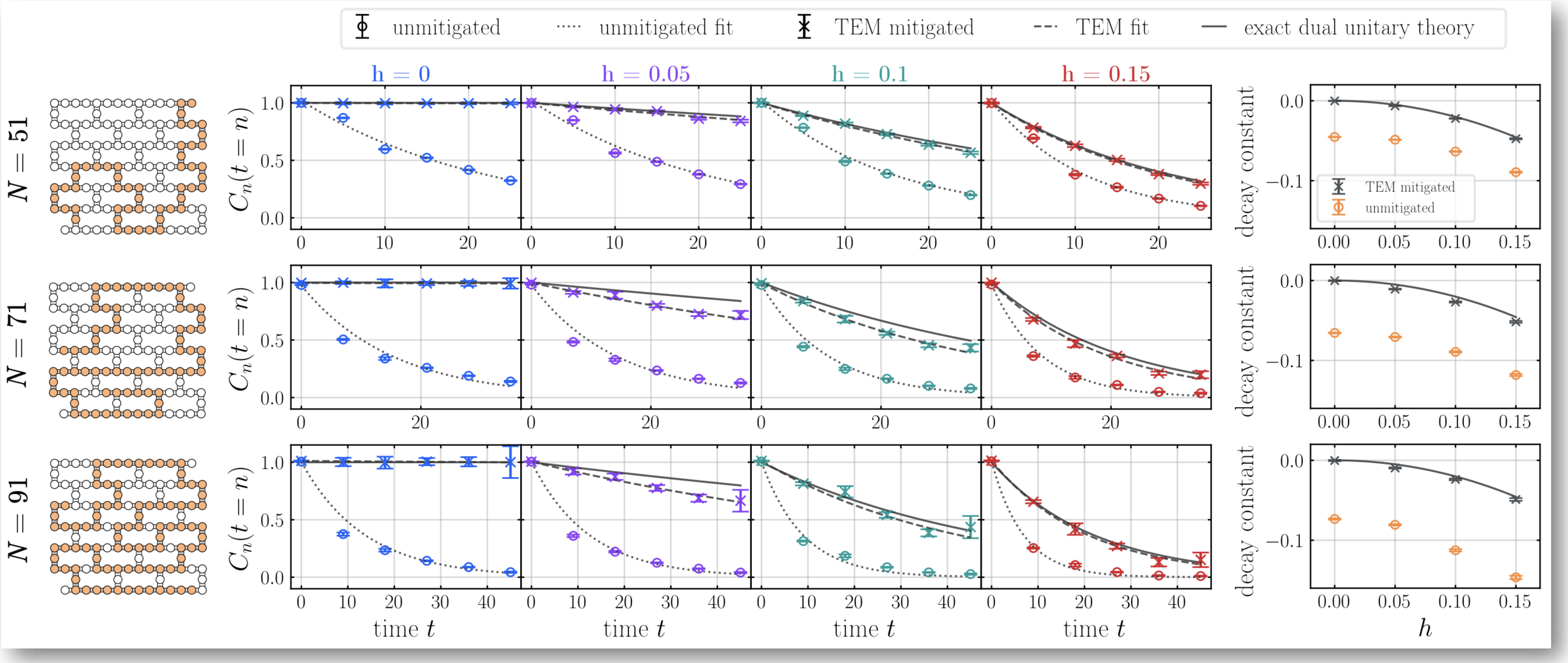
Error resilience time scales on QCSC architectures

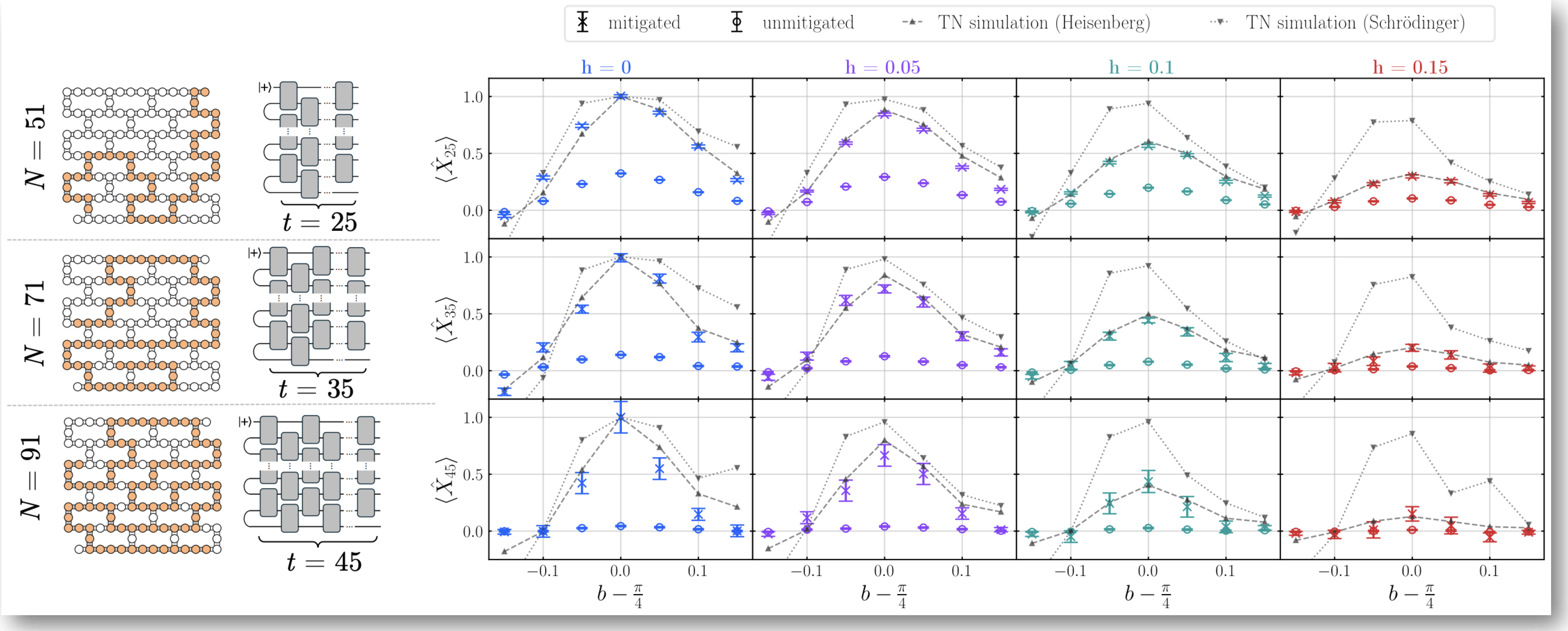
● QEC
■ QEM Unbiased
■ QEM Biased

	Middleware		Qiskit Runtime		Quantum Hardware
Timescale	Cloud loop time	Server loop time	Near-time		Real-time
	hours – minutes	seconds	ms	µs-ns	ns
Error resilience related function	Unitary parameter update Custom QPU info share Logical circuit schedule and compile optimization Final decode/reconstruct/correction	Compile probabilistic circuits (■) Noise model update (■ ■) Optimal qubit maps (■ ■) Multi-QPU job optimization	Physical parameter update Decoder update (●) Decode/Correct (●)	Probabilistic gates (■ ■) Stretched gates (■) Dynamic logical ops (●) Fast decode, high BW (●) Decode data compression (●)	Mid-circuit measurement and feedforward (●) Non-local real-time operations (●)
Hardware challenges	Pipelining with QPUs Low latency HPC optimization Final level of decoding	Parallel QPU utilization Data BW Decode compute	FPGA Update AoT compiling	Fast reinitialization (■ ■) Intra-qubit comm Local decode and compress	Local memory size High-BW acquisition Intra-qubit latency

The era of quantum utility







arXiv:2306.17839
arXiv:2411.00765

Different Types of Users

Quantum Physicist

Who they are
Quantum physics experts with practical experience testing and tuning quantum hardware

Skills
System capability discovery for using quantum systems at larger scales and complexities

Pain points
Access to powerful tools for configuring capabilities

Quantum Computational Scientist

Who they are
Research Scientists that started with quantum early and have experience designing quantum algorithm for domain problems

Skills
Focus on algorithms that generate circuits, not on optimizing the quantum execution for our hardware

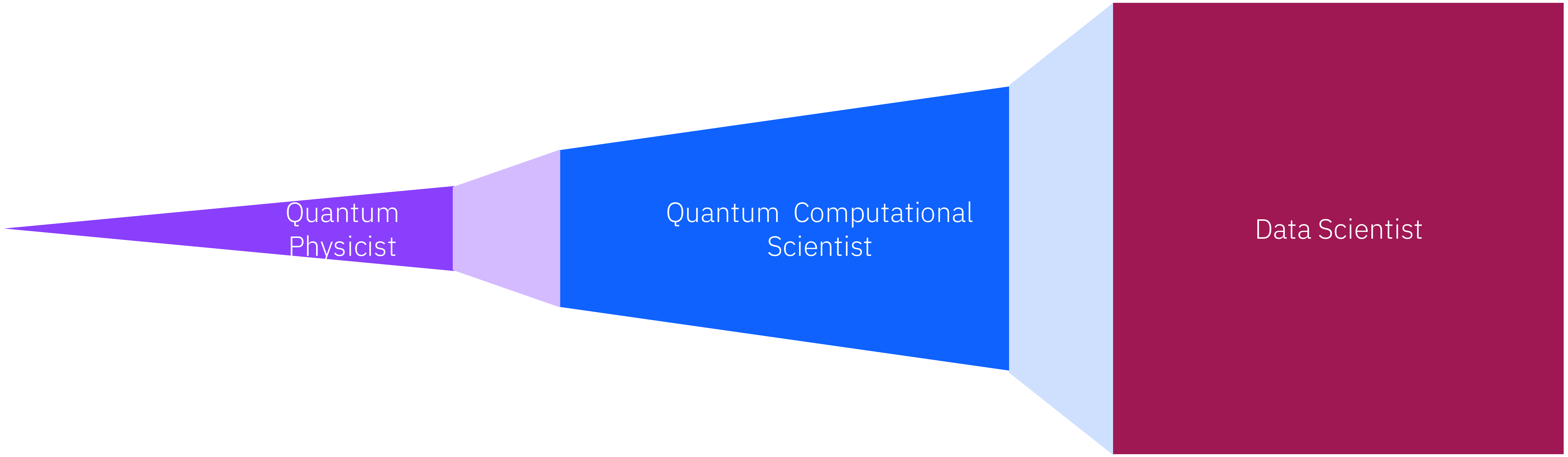
Pain points
Keeping up with new relevant capabilities for improving result quality on quantum hardware

Data Scientist

Who they are
Industry research scientists that understand the limits of current state of the art solutions for key application areas

Skills
Focus on problem domain, not on circuits

Pain points
Knowing how/which quantum approach to incorporate into application use case



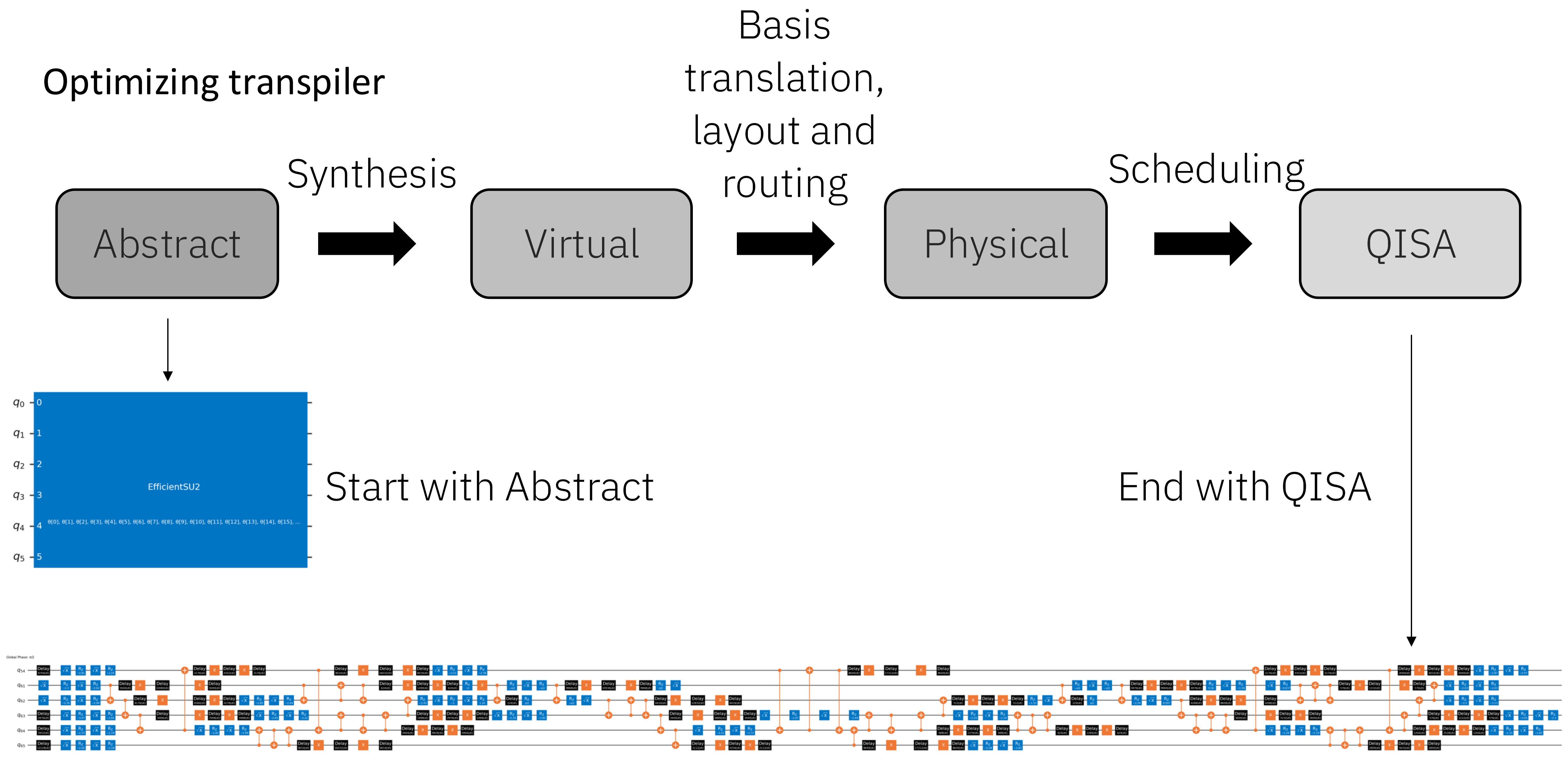
Two main components of Qiskit

Circuit and operator building APIs

```
with circuit.if_test((c0, 0b1)) as else_:
    circuit.x(q0)
with else_:
    circuit.z(q0)

Z = SparsePauliOp('Z')
X = SparsePauliOp('X')
Id = SparsePauliOp('I')
ZZ = Id ^ Id ^ Id ^ Z ^ Z
XX = Id ^ Id ^ Id ^ X ^ X
```

Optimizing transpiler

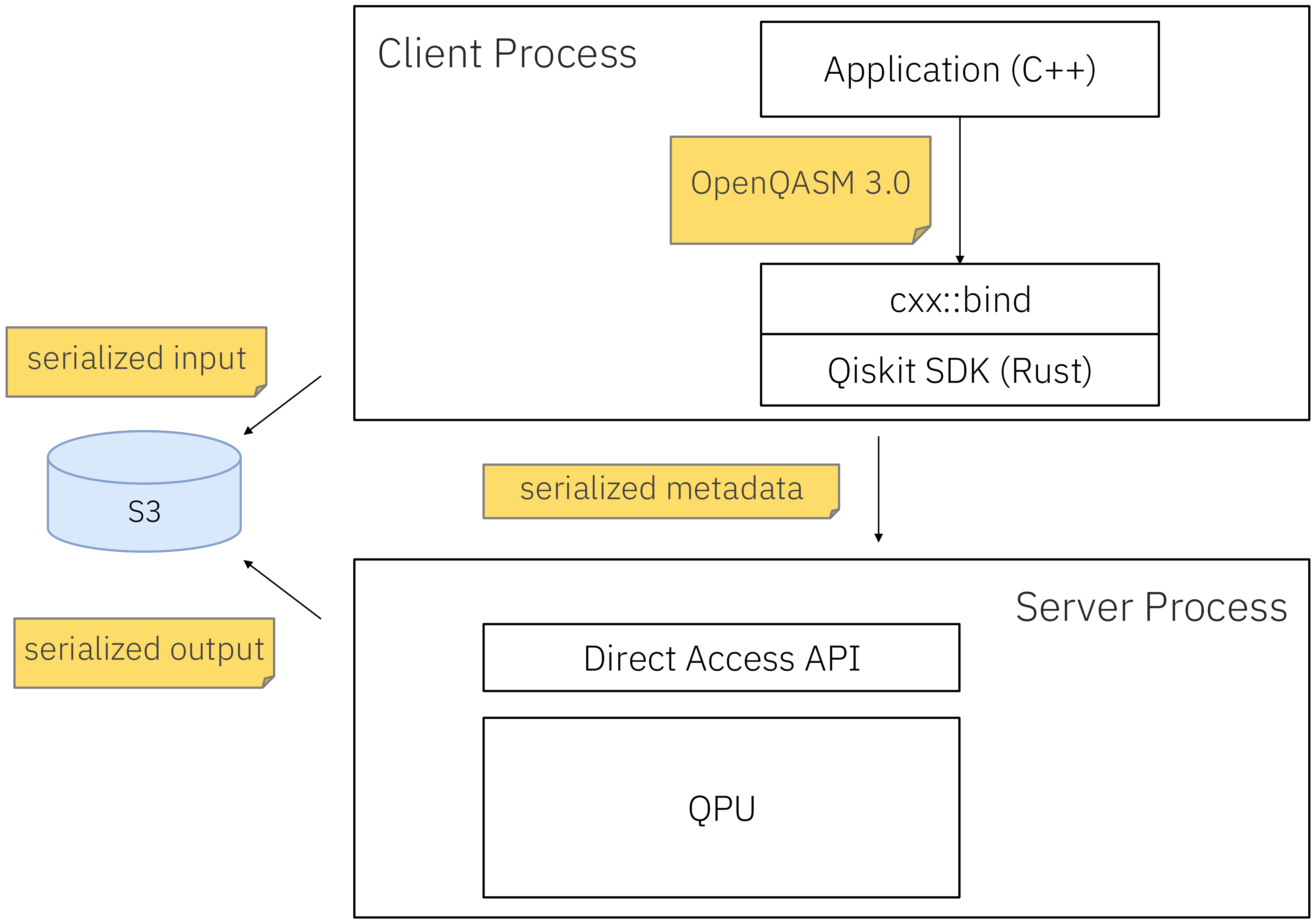


Qiskit SDK for high performance

1. Continued “oxidation” (conversion to Rust) of Qiskit SDK’s core

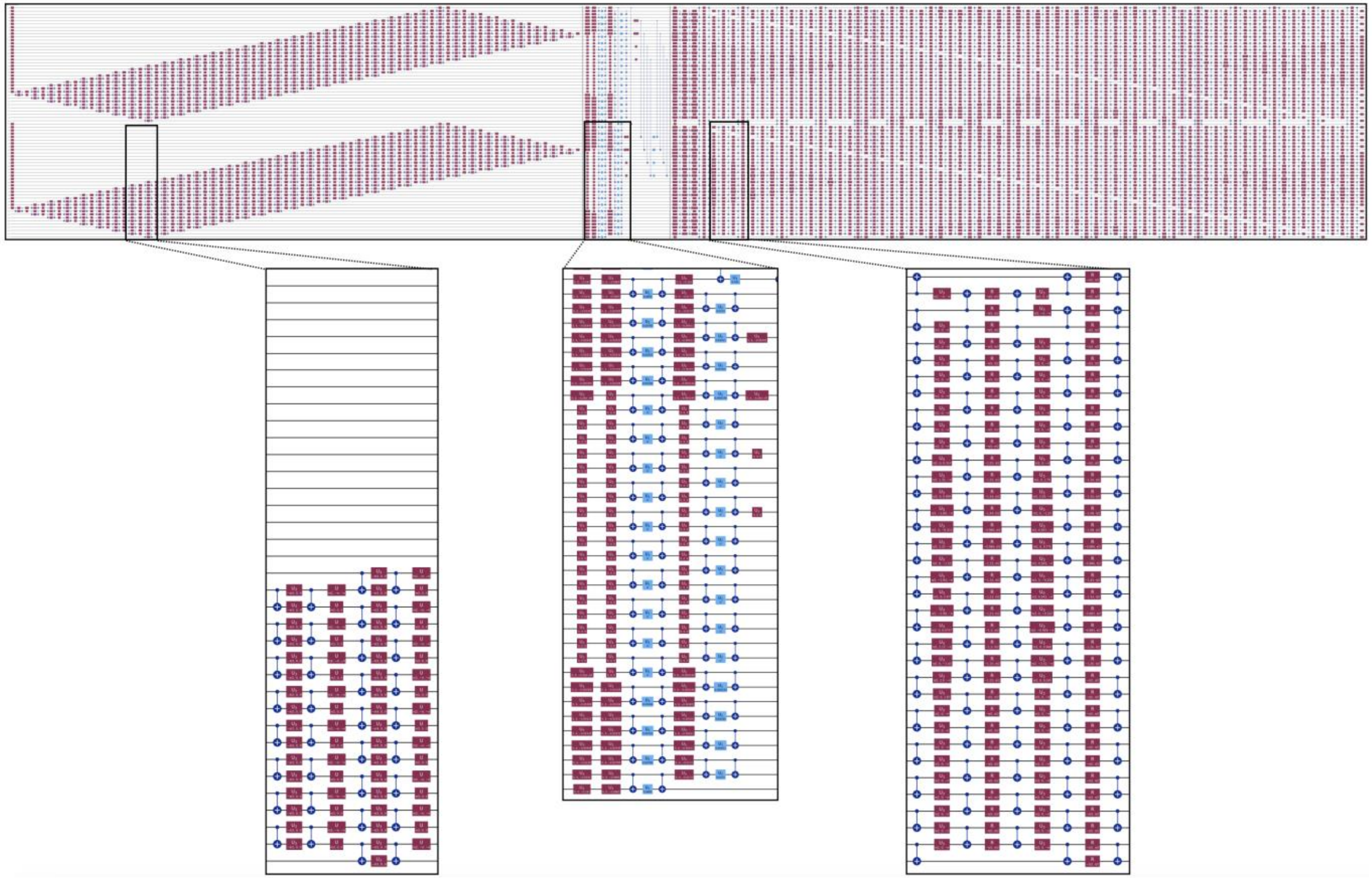
100+ transpiler passes in Qiskit
~10% converted to Rust

2. High performance application compiling



[4Fe-4S] circuit structure

Robledo-Moreno *et al*. arXiv:2405.05068



LUCJ circuit used to simulate the ground state of the [4Fe-4S] cluster, compiled into single-qubit (red, purple blocks) and CNOT (light blue symbols) gates.

Magnified views (left to right) of gates from the first orbital rotation, density-density interaction, and second orbital rotation.

3590 2-qubit gates (10,570 gates total)