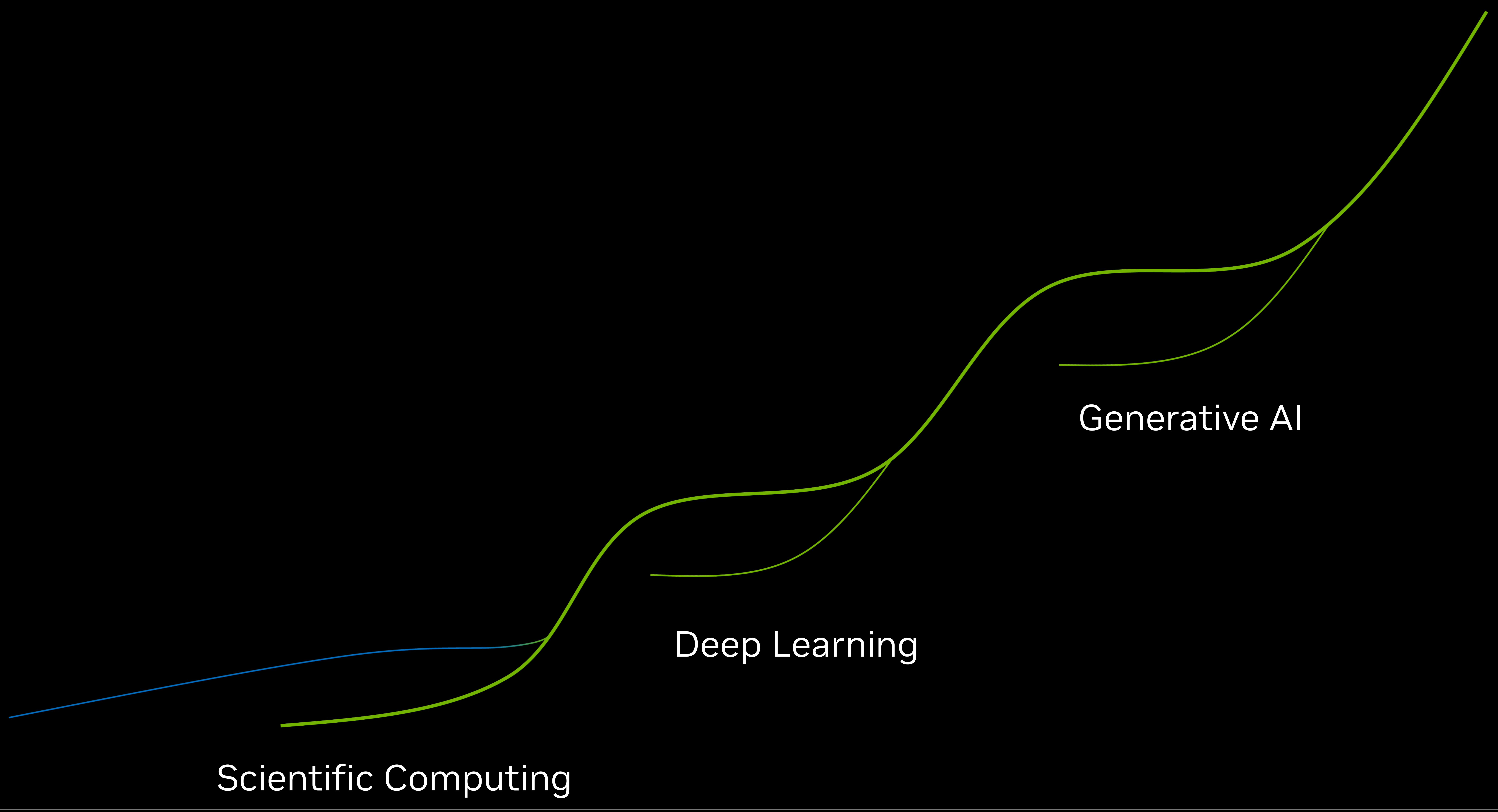
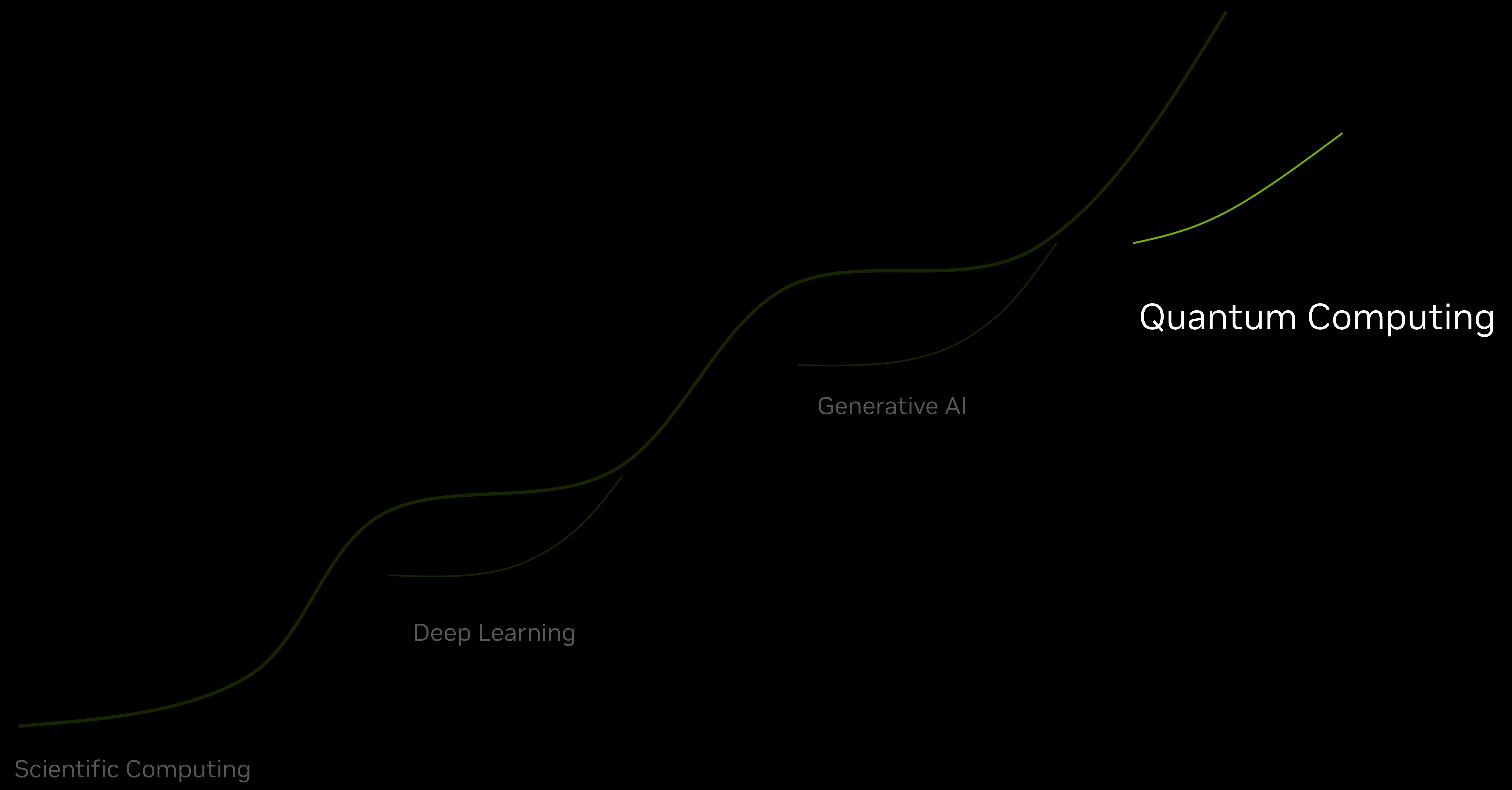
Accelerated Quantum Supercomputing using CUDA-Q

Esperanza Cuenca-Gomez, Developer Relations Manager, Quantum Computing

NVIDIA's History of Enabling Computing Revolutions



NVIDIA's History of Enabling Computing Revolutions

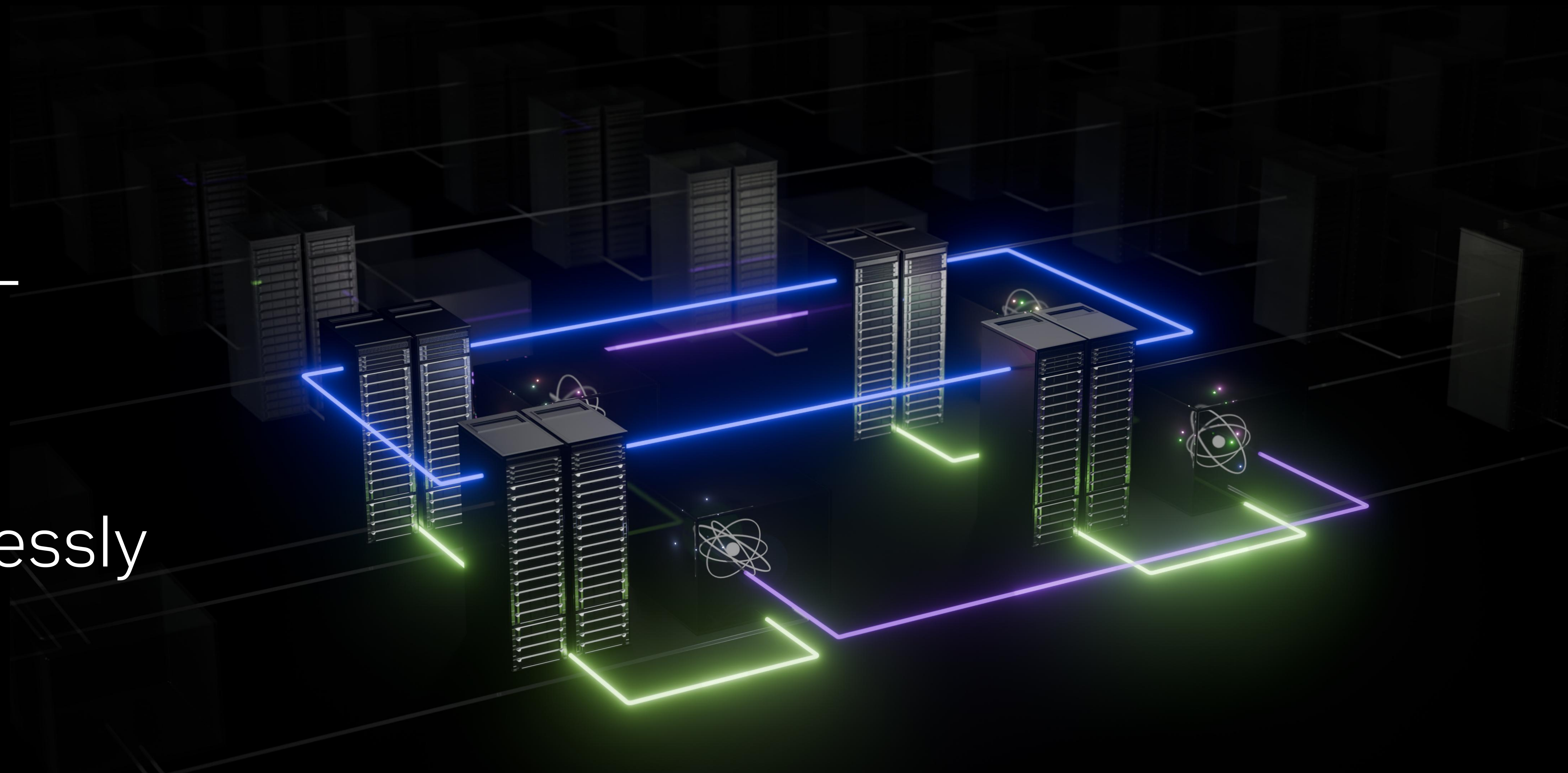


NVIDIA is not building
Qubits

NVIDIA is building all
Accelerated Quantum
Supercomputers

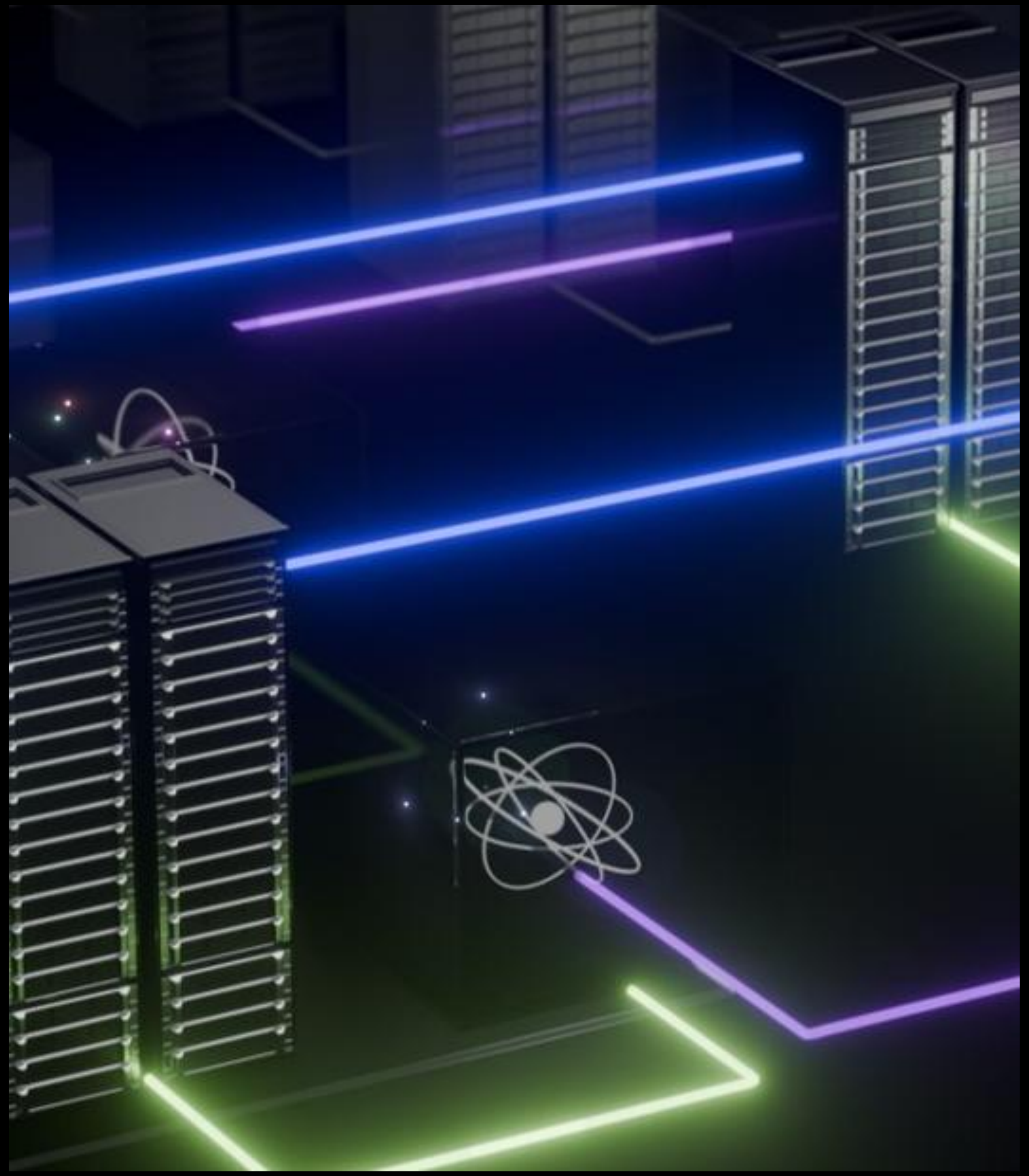
The Accelerated Quantum Supercomputer

- Supercomputing architecture **connecting quantum hardware**
- Ability to run **hybrid algorithms** - using GPUs and QPUs
- A **software platform** that seamlessly connects hybrid applications
- The ability to perform **qubit-agnostic** development of control and error correction



Quantum Computing Needs Accelerated Computing

AI SC for QC Deployments



Quantum Error Correction

Hybrid algorithms and applications

AI for
- Calibration
- Control
- Readout

AI SC for QC Development

Accelerated application development

AI assisted circuit design

Dynamical simulations

Noise modeling

Practical Post Quantum Cryptography

Powering the Quantum Ecosystem

The only quantum company that works with every other quantum company

200+
NVIDIA Quantum Partners

>90%
Largest Startups Working with NVIDIA

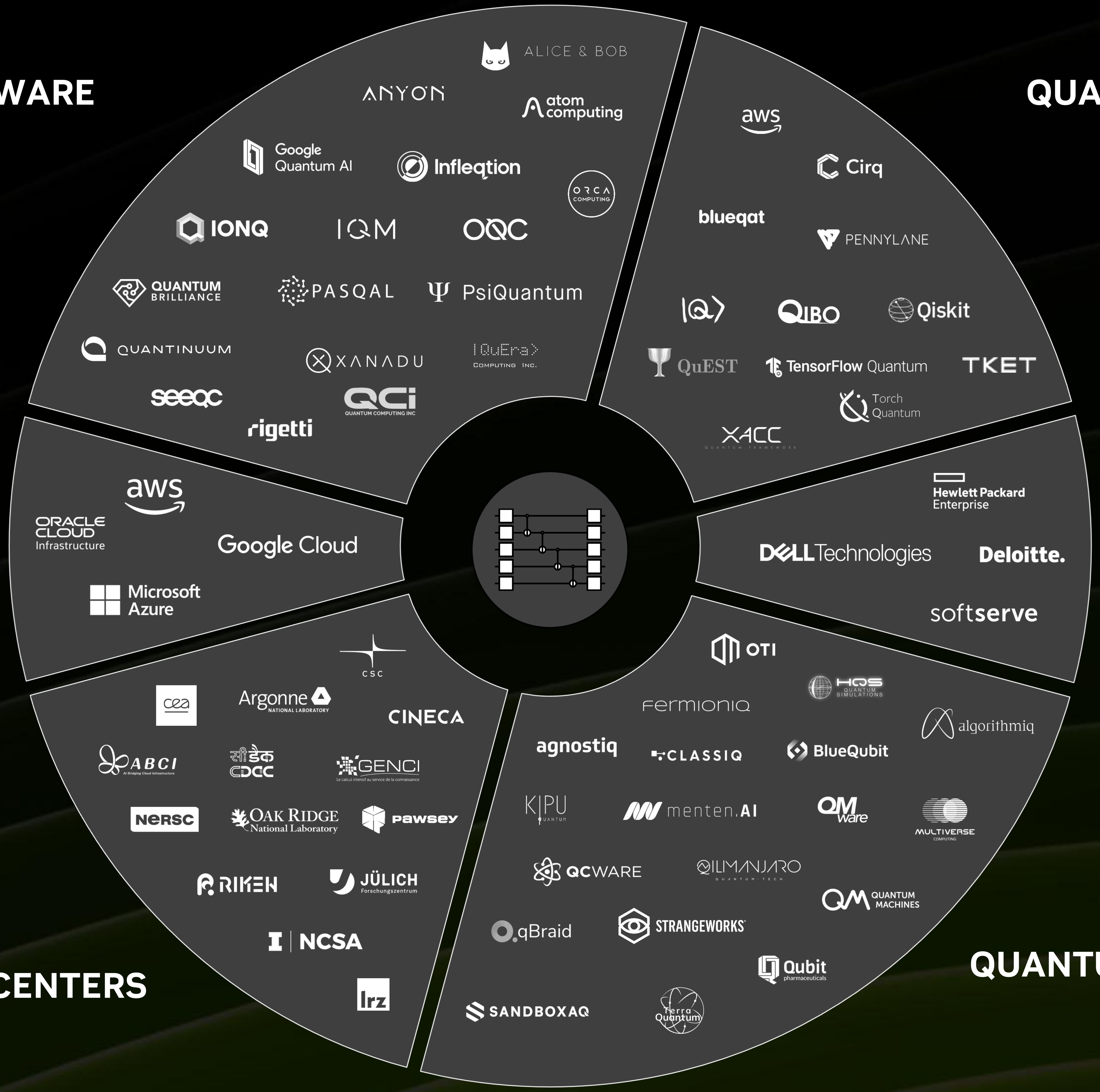
>80%
QPUs Integrating NVIDIA Software

100%
Leading Quantum Development Frameworks Accelerated

QUANTUM HARDWARE BUILDERS

CLOUD SERVICE PROVIDERS

RESEARCH CENTERS



QUANTUM SIMULATION FRAMEWORKS

SYSTEMS BUILDERS & INTEGRATORS

QUANTUM SOFTWARE AND SYSTEMS

NVIDIA

The NVIDIA Quantum Ecosystem

Accelerating the Quantum World

agnostiq, algorithmiq, ALICE & BOB, Amazon Braket, ANYON, Anyon Technologies, ATLANTIC QUANTUM

atom computing, blueqat, BlueQubit, CLASSIQ, diraq, D-Wave

Entropica Labs, equal1, fermioniq, Google Quantum AI, HQS Quantum Simulations, horizon quantum, Infleqtion

|Q>, IQM, IONQ, Kipu, menten.AI, Microsoft, Nord Quantique, ORCA Computing, OQC

OTI, Pasqal, PENNYLANE, .PLANQ, PsiQuantum, qBraid, QCDESIGN

QCi Quantum Computing Inc., QCWARE, QIBO, QILIMANJARO, Qiskit, QM Quantum Machines, QMware, QUANDELA

QUANTINUUM, Quantum Art, QUANTUM BRILLIANCE, QUANTUM ELEMENTS, Quantum Rings, Qubit Pharmaceuticals

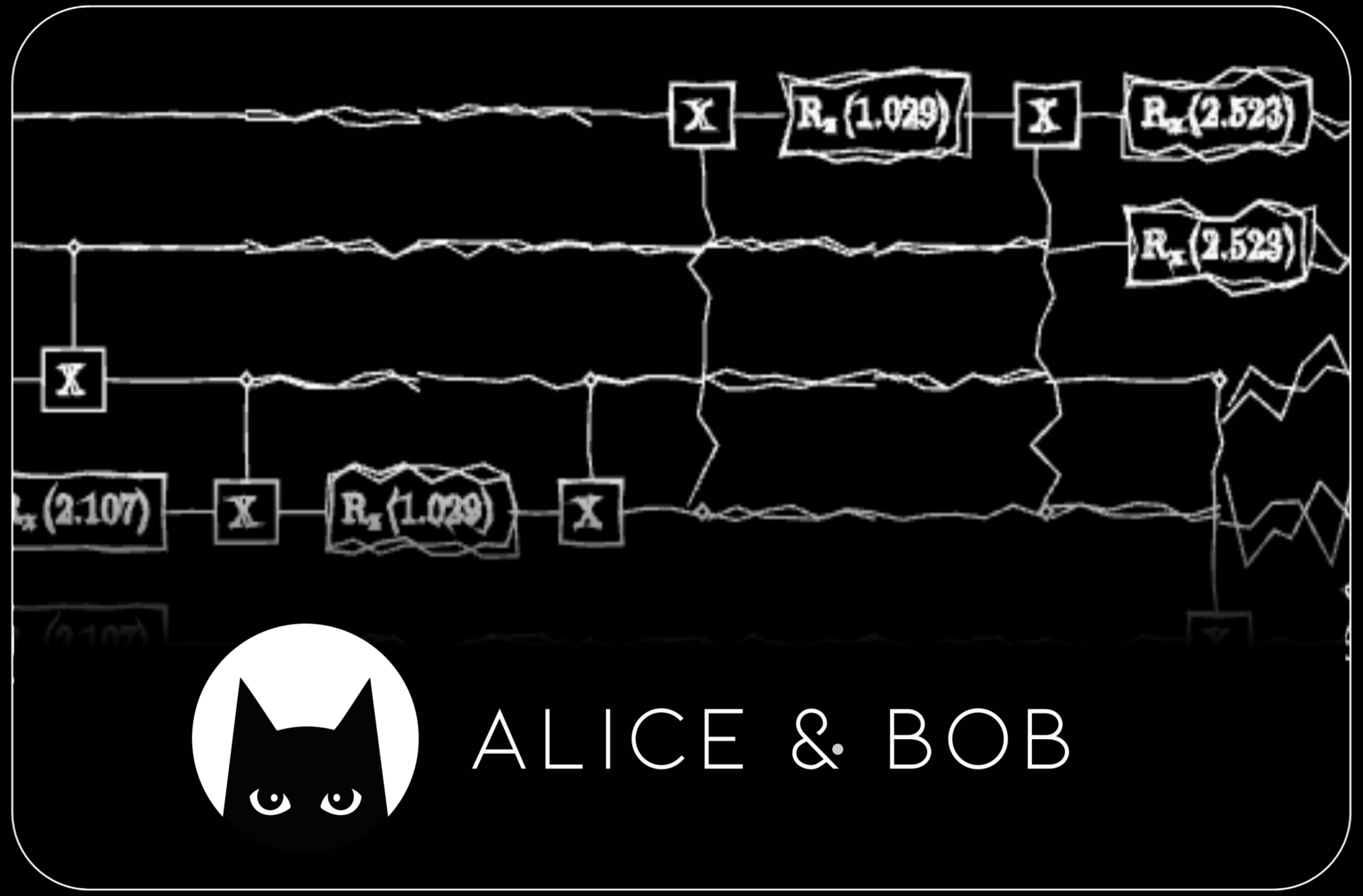
QuEra, QUDORA Technologies, QuEST, quobly, rigetti, seeqc, STRANGEWORKS

SANDBOXAQ, Terra Quantum, TKET, Torch Quantum, XACC, XANADU

Accelerating Quantum Workloads Across Europe

Quantum Algorithm Development

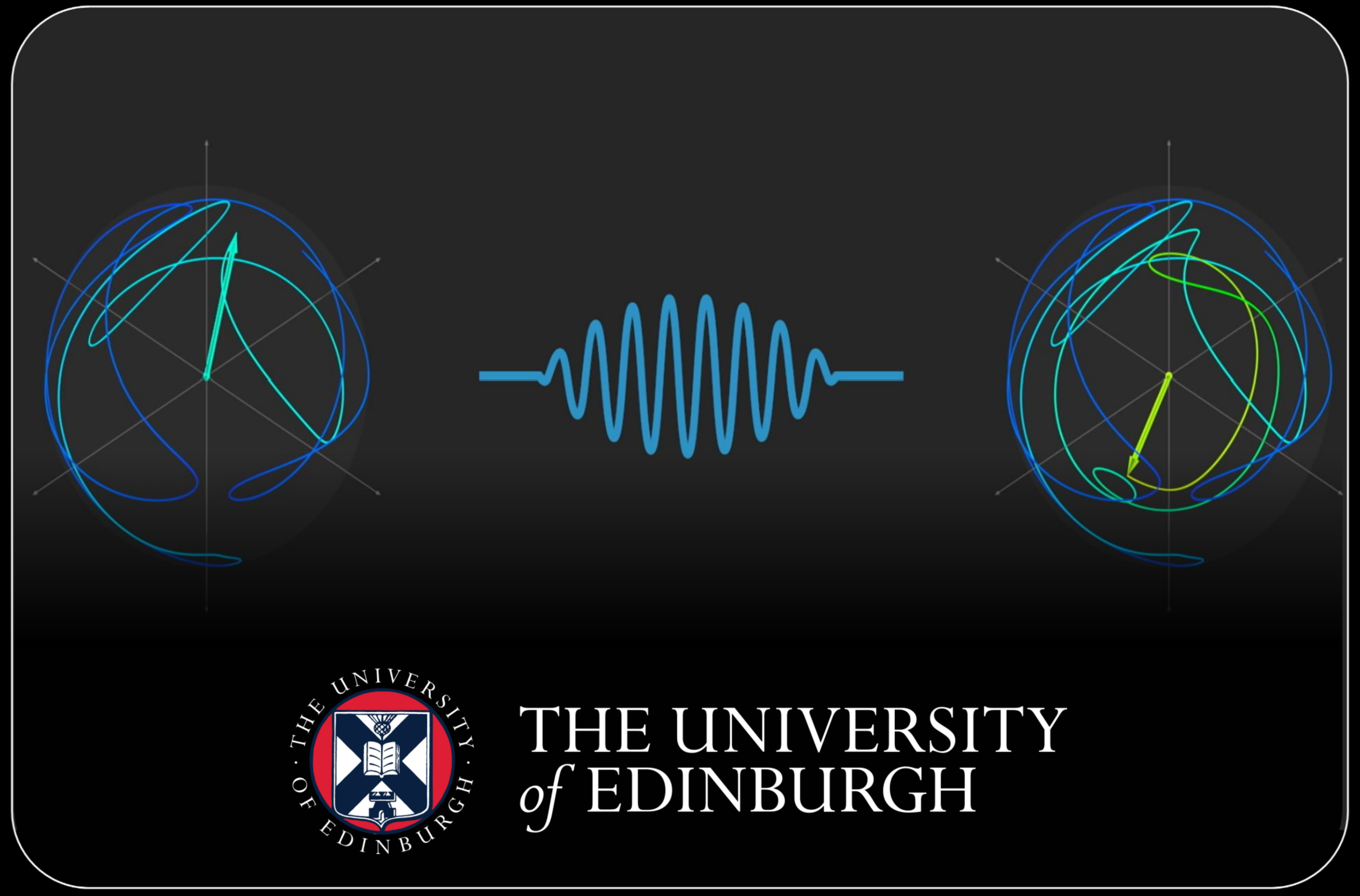
Quantum EDA



Quantum Data Generation

Hybrid Applications

Quantum Error Correction



Defining the Accelerated Quantum Supercomputer

A New Heterogenous Architecture

Programming model and compiler for heterogenous supercomputer

Low-latency interconnects for real-time hybrid computing

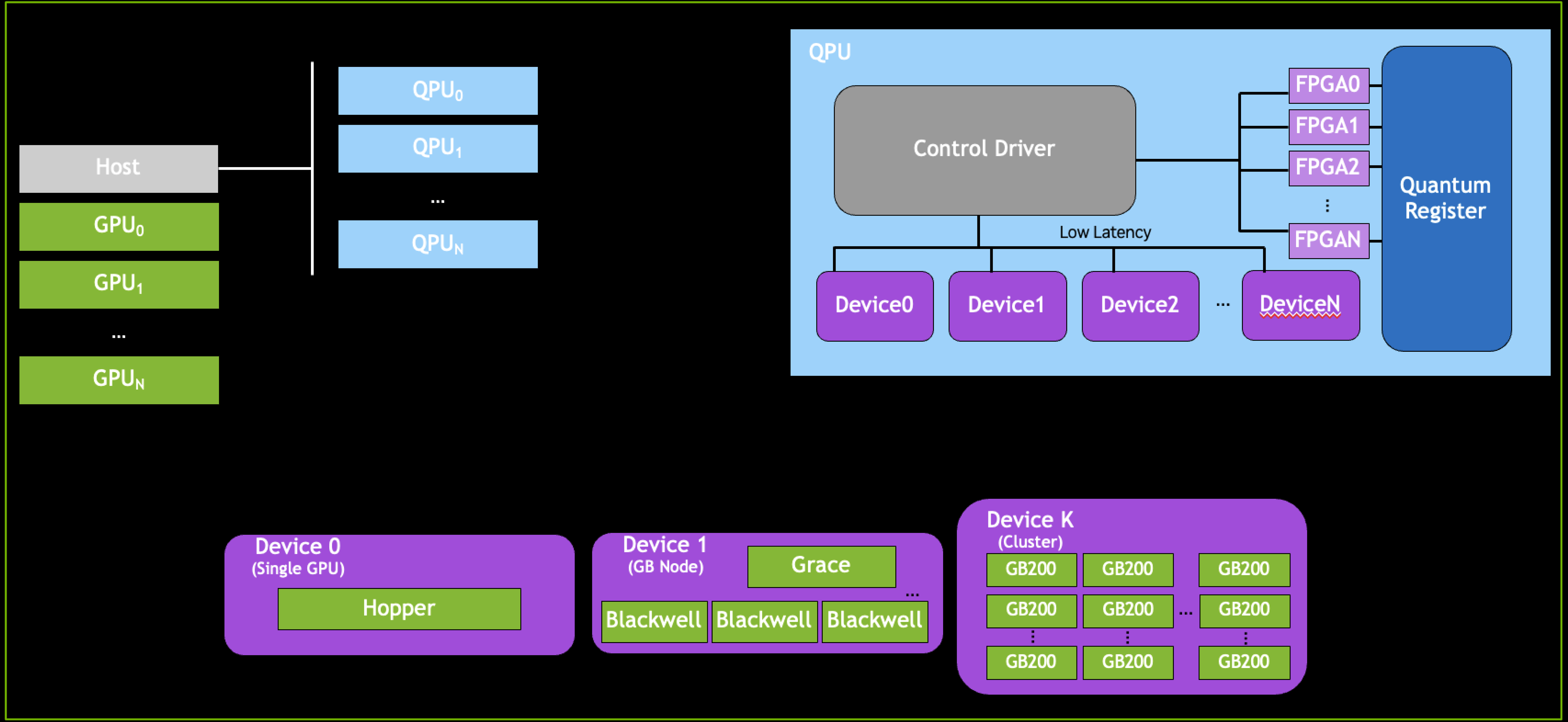
Libraries to enable domain scientists

Open source and qubit-agnostic

Libraries, Applications, Frameworks

CUDA-Q

Platform for Programming Accelerated Quantum Supercomputers



CUDA-Q

The platform for accelerated quantum computing

CUDA-Q

Libraries | Programming Model | Tools | QEC | QPU Design | Infrastructure

Simulation

QPU Hardware

OQC
QUANTINUUM
ORCA Computing
ALICE & BOB
IONQ
QUANTUM BRILLIANCE
PASQAL
IQM
QuEra Computing Inc.
seeqc
rigetti
Infleqtion
Anyon Technologies

AI Supercomputing

CUDA-Q

The CUDA-Q stack

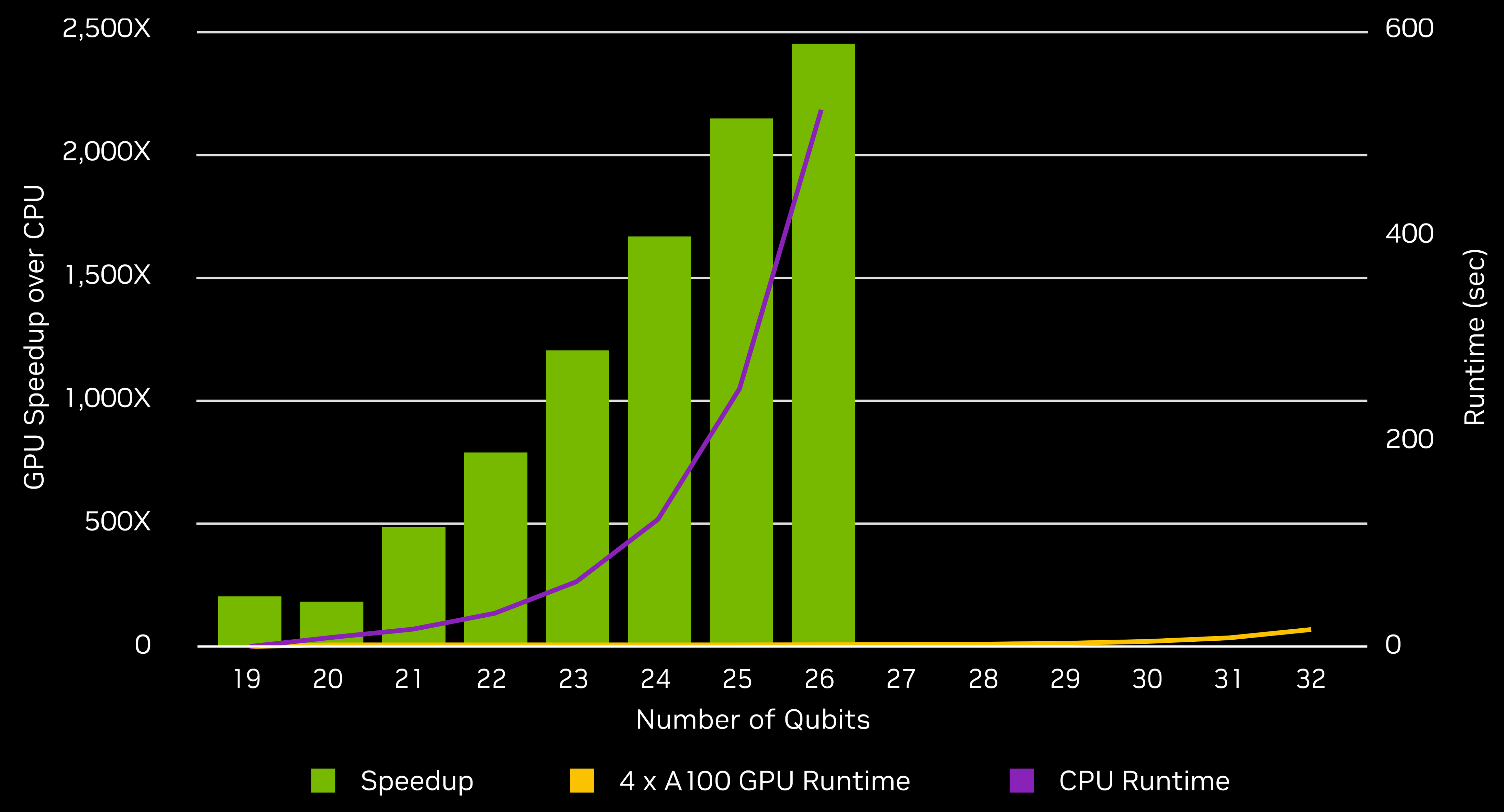
CUDA-Q

The platform for accelerated quantum computing

Features

- **Python and C++**
  - Access via familiar & powerful languages
- **QPU agnostic**
  - Supports backends from all major QPU vendors and qubit types
- **GPU-accelerated** simulation
  - Quantum simulators that scale to large-scale quantum computers
- Fully **kernel system for hybrid computing interface**
  - Seamlessly combine GPU and QPU resources
- Supports **QEC HW development**
  - DGX-Quantum reference architecture allows decoder and calibration development
- Access to classical **CUDA-X libraries**
  - Conventional parts of hybrid algorithms can draw on fastest implementations
- Comprehensive **educational tools**
  - CUDA-Q Academic onboards users to accelerated quantum supercomputing

Performance



QML workflow in CUDA-Q using multithreaded CPU versus NVIDIA A100 Tensor Core GPUs

Getting started with CUDA-Q

CUDA-Q Overview
https://developer.nvidia.com/cuda-q

CUDA-Q Academic
https://github.com/NVIDIA/cuda-q-academic

CUDA-Q Docs
https://nvidia.github.io/cuda-quantum/latest/index.html

CUDA-Q Apps
https://nvidia.github.io/cuda-quantum/latest/using/tutorials.html

NVIDIA

Announcing Dynamics in CUDA-Q

Enabling QPU developers everywhere to accelerate their design process

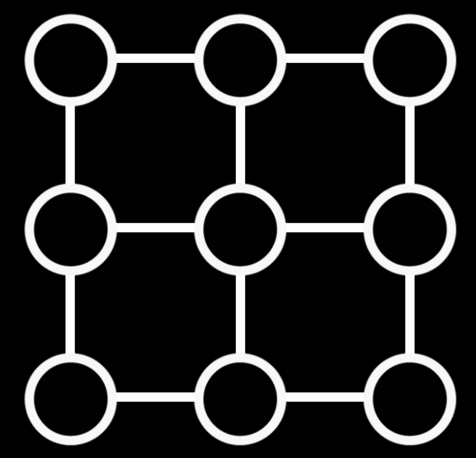
CPU: Intel Xeon 8480CL | GPU: DGX H100
Energy levels: 32 Transmon x 128 Resonator x 4 Purcell Filter

CUDA-Q Libraries

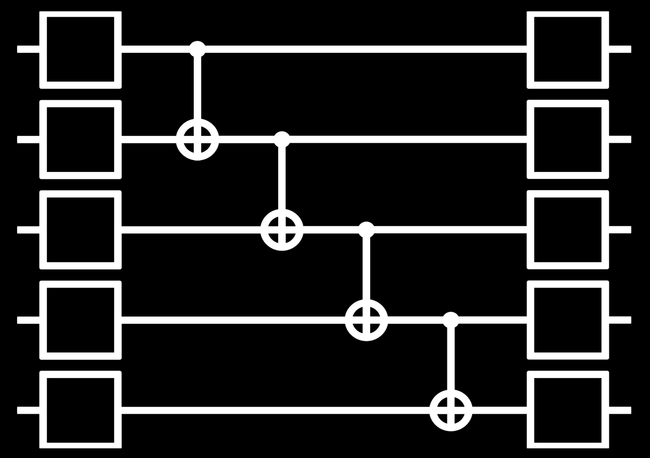
Bringing the power of CUDA-Q directly to quantum computing research problems

NVIDIA CUDA-QX

Seamless integration of accelerated **QEC primitives** into CUDA-Q workflows



CUDA-Q QEC



CUDA-Q Solvers

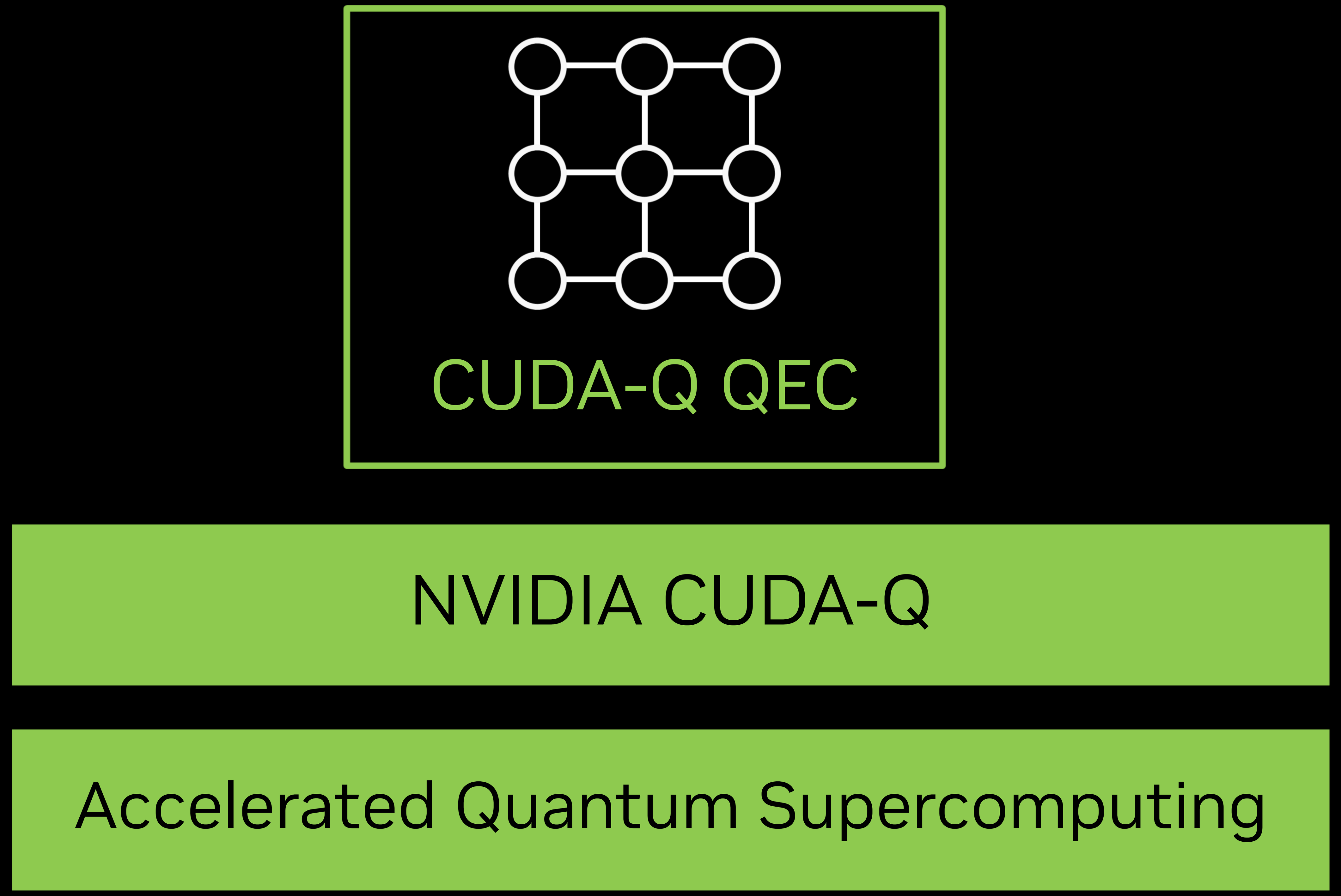
Accelerating **solvers** – key algorithms within many quantum applications

NVIDIA CUDA-Q

Accelerated Quantum Supercomputing

CUDA-Q QEC

- Prebuilt optimized codes and decoders in CUDA-Q
- Builtin stabilizer simulator – stim
- Extension point to define custom codes and custom decoders in CUDA-Q
- Open source – available at https://github.com/NVIDIA/cudaqx



AI for QC

Review paper - https://arxiv.org/abs/2411.09131

Section 2: QC Hardware Development and Design

- System characterization
- Platform design
- Gate and Pulse optimization

Section 3: Preprocessing

- Quantum circuit synthesis
- Circuit parameter learning
- State preparation

Section 4: Device Control and Optimization

- Device tuning
- Qubit control
- Device characterization

Section 5: Quantum Error Correction

- QEC decoding
- Code discovery

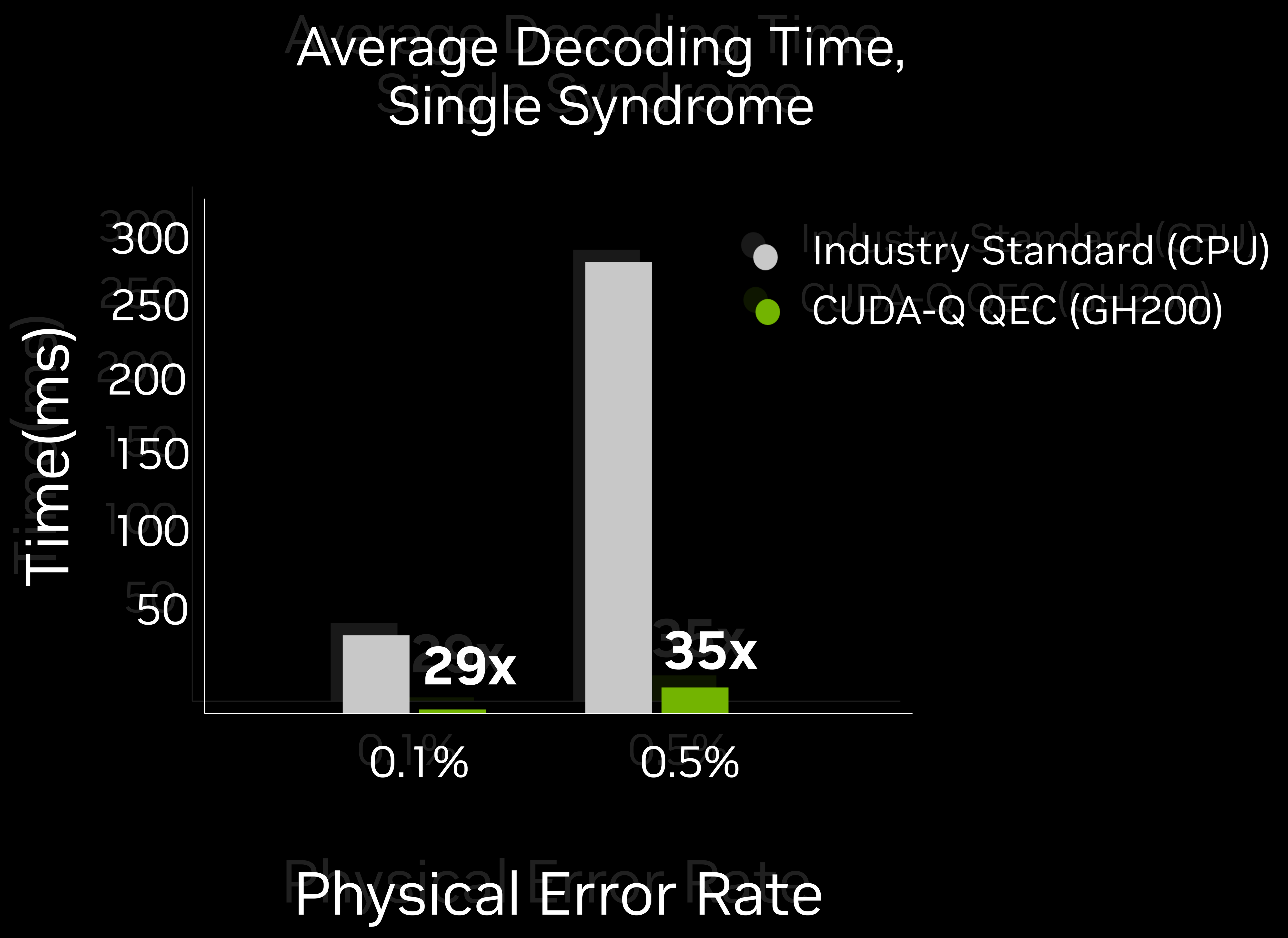
Section 6: Postprocessing

- Observable estimation and tomography
- Qubit Readout
- Error mitigation

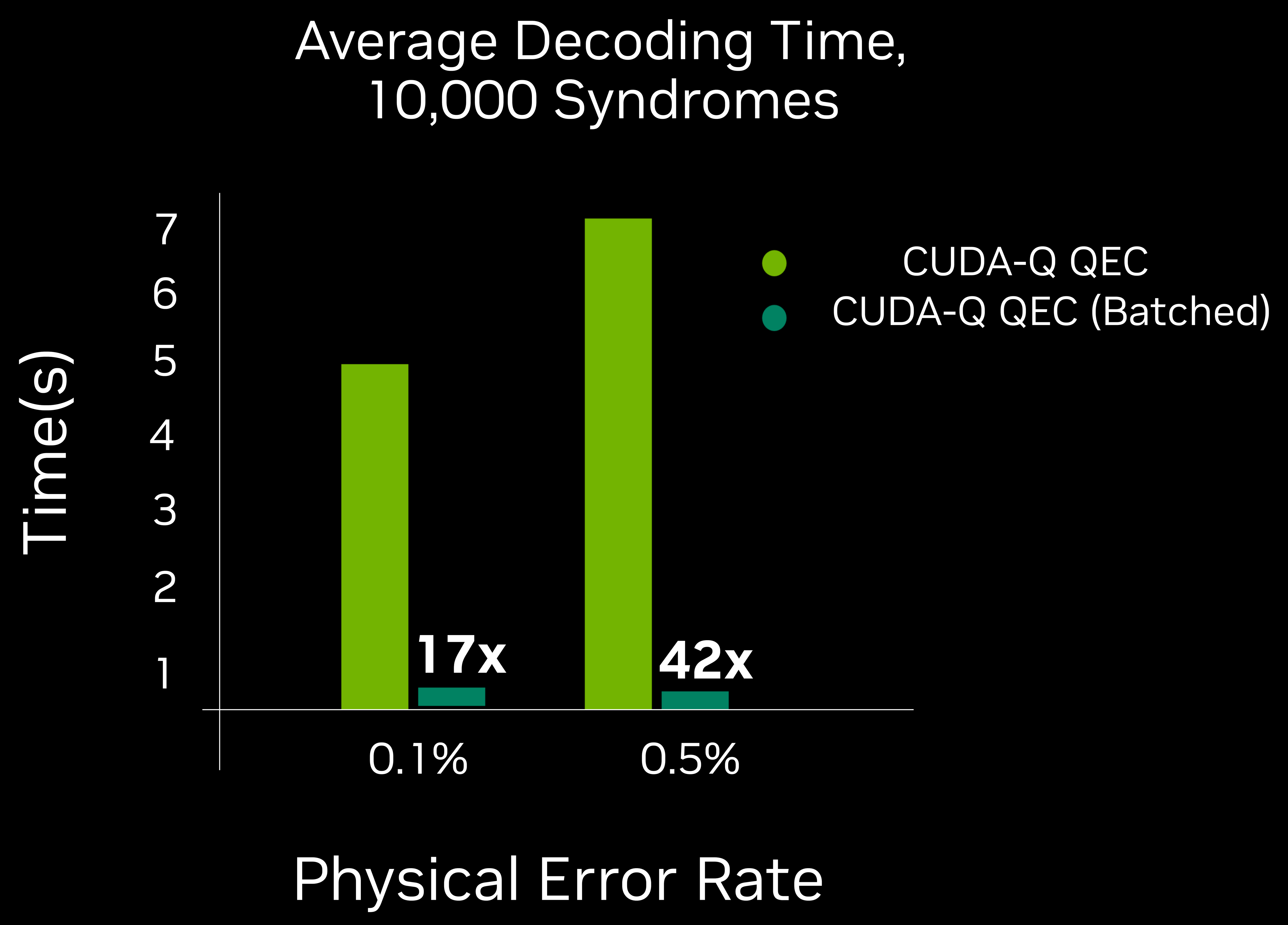
Announcing CUDA-Q QEC 0.2

Accelerating BP-OSD Decoding

Latency
29-35x speedup over industry standard



Throughput
Additional 17-42x speedup for batched decoding



All benchmarks on [[144, 12, 12]] code

The Generative Quantum Eigensolver

First demonstration of GPT-generated circuits

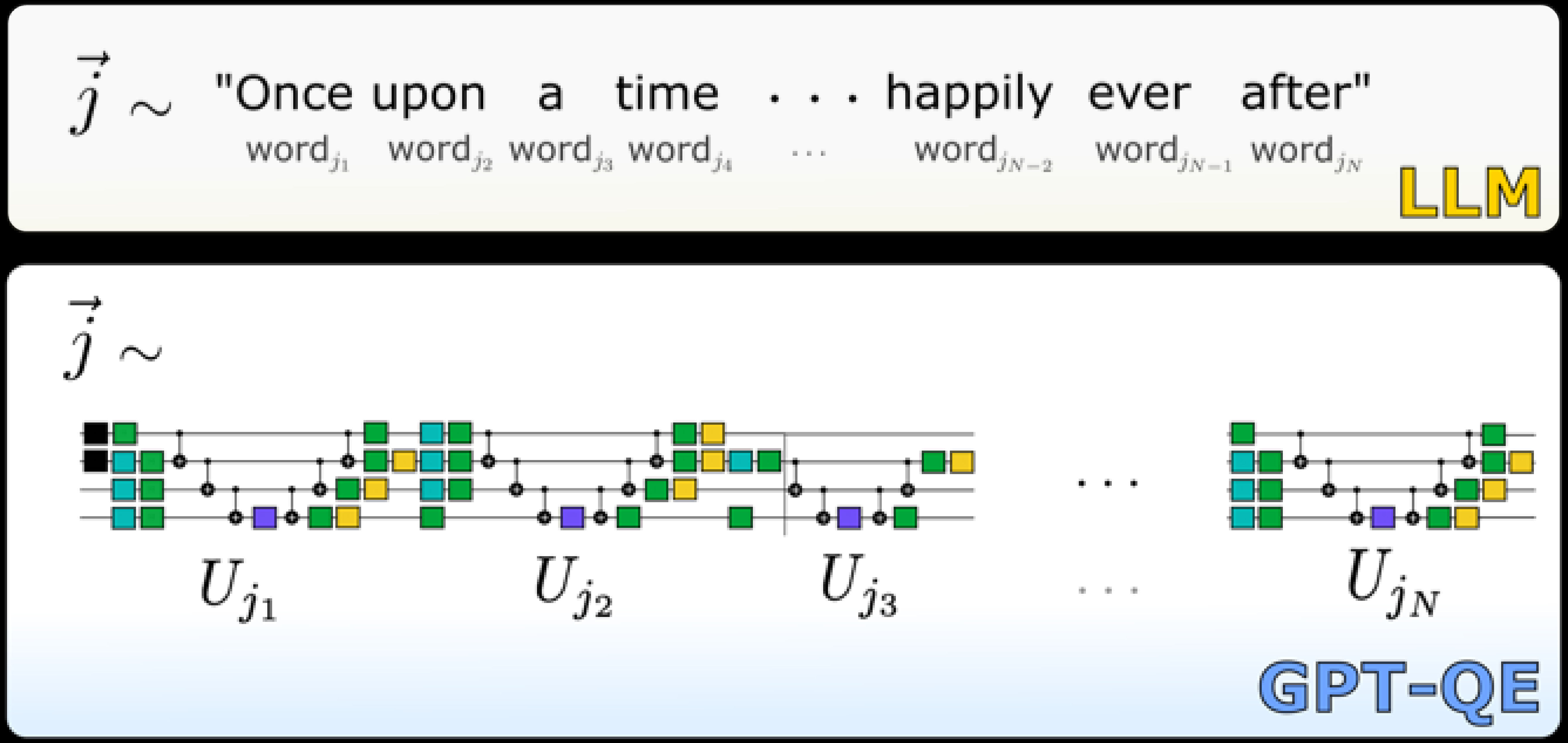
CUDA-Q + St. Jude Children's Research Hospital + University of Toronto

Challenge

- Variational quantum algorithms offered promise for running drug-discovery applications on small quantum devices - but **suffer from serious optimization issues**.
- Many of these problems are tied to how circuits are parametrized.

Solution

- The generative quantum eigensolver acts like a Large Language Model – but generating quantum circuits from quantum operations, rather than sentences from words.
- Using a generative model like GPT to create quantum circuits **avoids the limitations of traditional variational quantum algorithms**



New approach in Using AI for building quantum applications

Can be extended to various application areas

40X

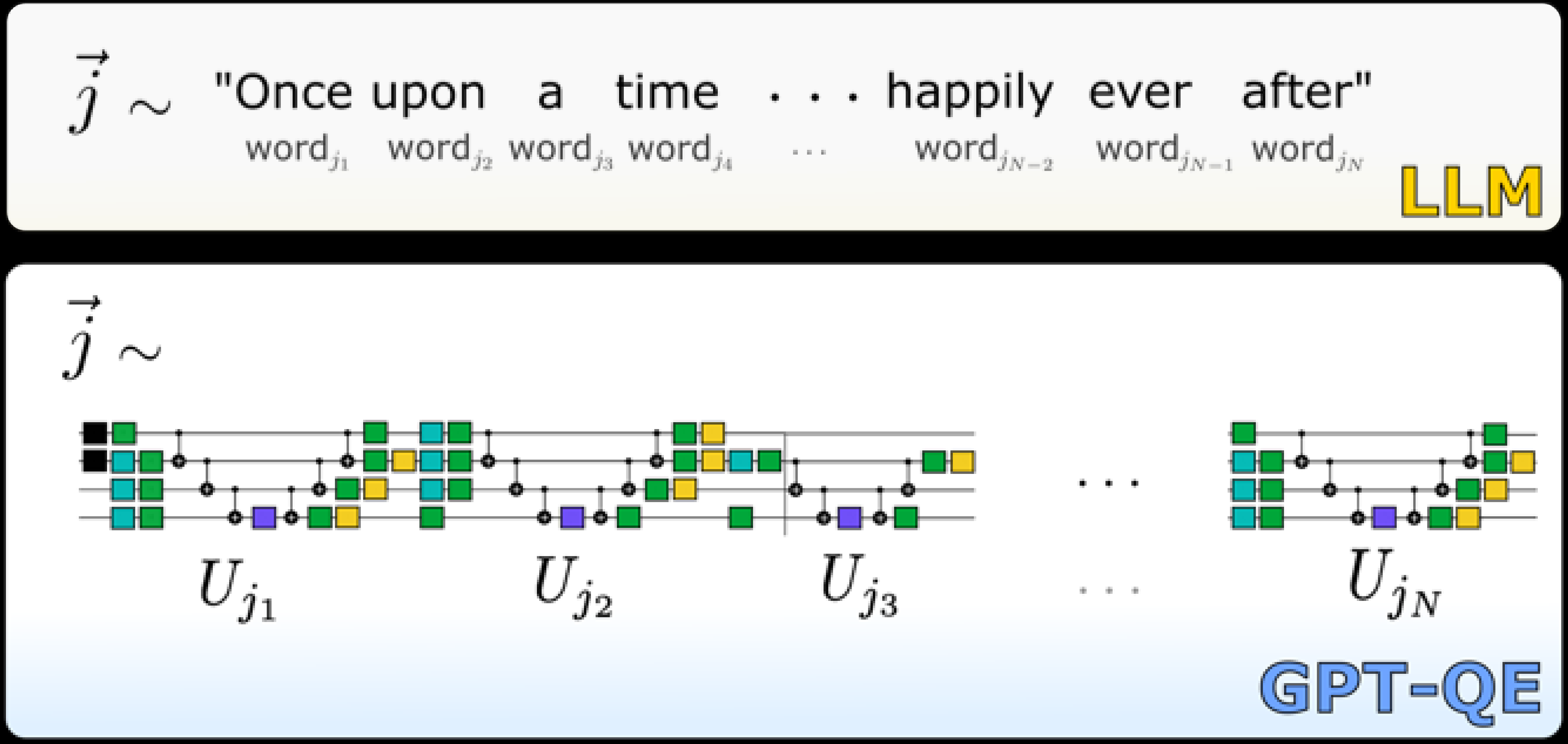
Speedup over CPU when running GQE on GPU

The Generative Quantum Eigensolver

First demonstration of GPT-generated circuits in the literature

Goal: Find a circuit producing e.g. ground state
Optimization of some possible space of circuits for desired output

- **GPT-QE specifically employs a GPT model**
  - Recent advances in attention-based transformer models can be leveraged
- **Analogy to Large Language Models (LLMs)**
  - Quantum operations are analogous to words
  - Quantum circuits analogous to sentences
  - When trained, GPT-QE generates a sequence of operations to form a circuit
- **Training learns weights in GPT model**
  - Cost function compares current prediction to measure of desired circuit output



UNIVERSITY OF TORONTO

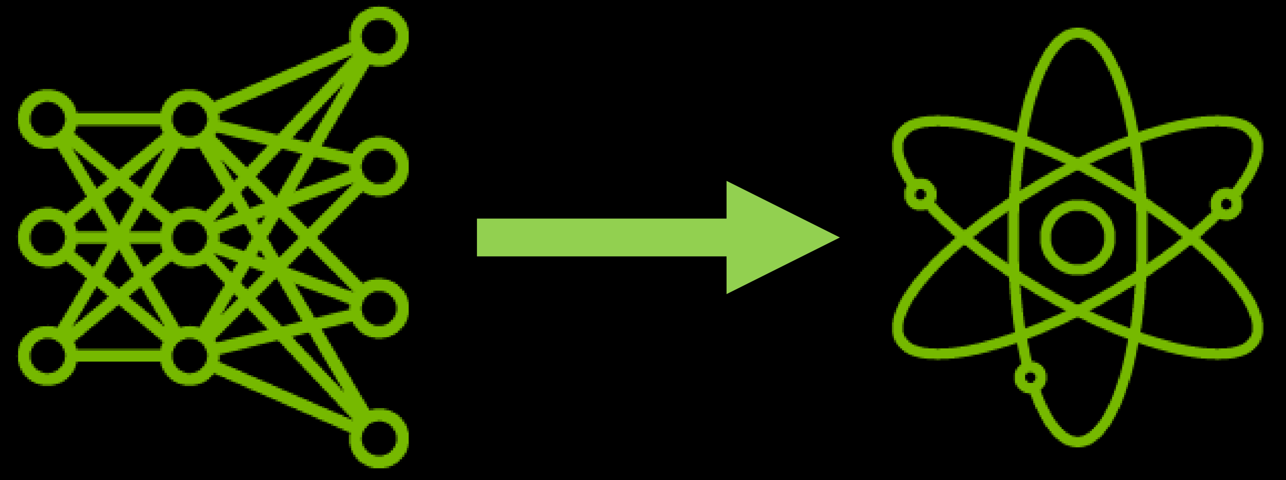
St. Jude Children's Research Hospital
Finding cures. Saving children.

https://arxiv.org/pdf/2401.09253.pdf

NVIDIA

Extension to other domains

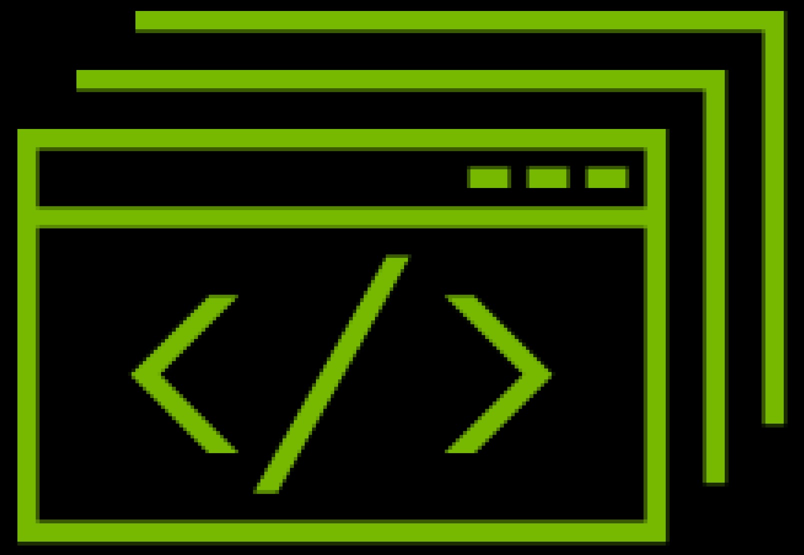
Pretraining databases can be extended to other applications



Results of current GPT-QE work is first step to understanding how GPT can accelerate QC

$$\{U_1, U_2, \ldots U_L\} \downarrow \{\tilde{U}_1, \tilde{U}_2, \ldots \tilde{U}_L\}$$

Hamiltonian and operator pool (GPT gate-set) can be switched out to explore applications beyond Chemistry and also accommodate different hardware



GPT-QE code available online at
https://github.com/cudaq-libraries/cudaqlib/tree/main/examples/python

NVIDIA Quantum

NVIDIA Quantum
https://www.nvidia.com/en-us/solutions/quantum-computing/

CUDA-Q v0.12 Now Available
Python – `> pip install cudaq`
C++ – https://github.com/NVIDIA/cuda-quantum/releases

NVIDIA