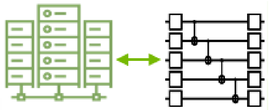
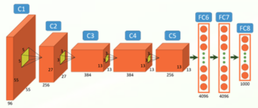
Introduction to CUDA-Q

Naruhiko Tan, HPC Solutions Architect

Accelerated Computing





Scientific Computing

Deep Learning

Generative AI

Quantum Computing

NVIDIA Quantum

Powering the Global Quantum Computing Community

160+

Quantum Partners

>90%

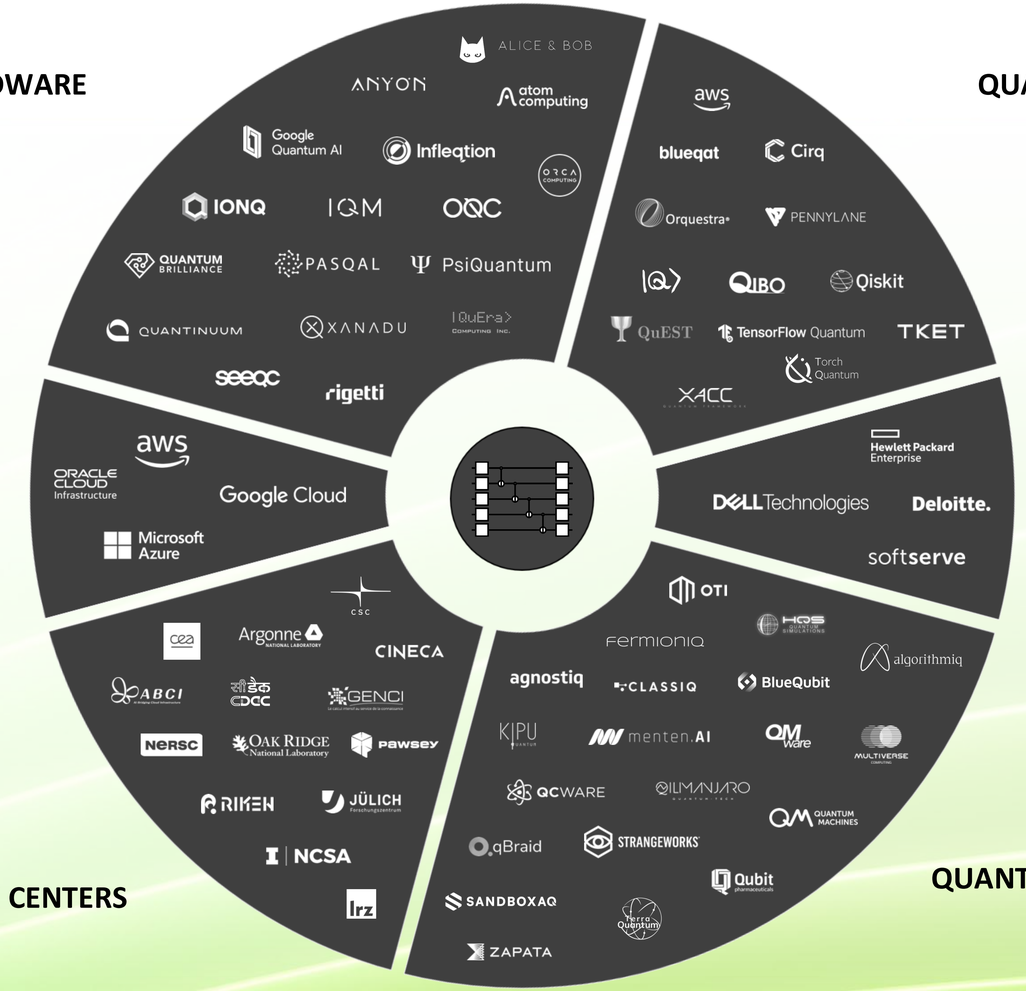
Largest Startups

>78%

QPUs Integrating CUDA Q

15/17

Leading Frameworks Accelerated



Quantum Accelerated Supercomputing

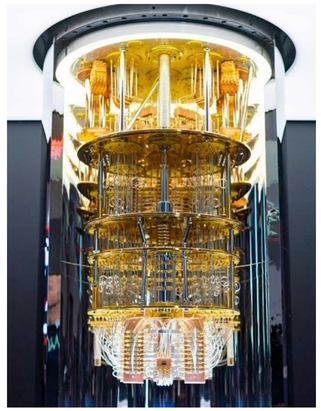
Supercomputers are the foundation of Quantum R&D

Simulation

- Quantum computers are small and error-prone -> simulation is an essential tool
- **Today:** Powerful simulators enable algorithm and application R&D – new approaches (e.g. tensor networks)
- **Future:** Digital twins of quantum computers for design and architecture optimization

HPC Quantum Integration

- Useful quantum computing will be hybrid
- **Today:** Enable domain scientists to start developing for QPUs, enable quantum researchers to use accelerated computing
- **Future:** quantum computers will integrate tightly with supercomputers as accelerators and be co-programmed

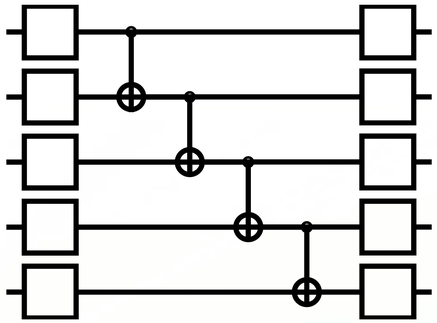


AI for Quantum

- Error correction, calibration, control, compilation are challenging computationally, real-time compute often needed
- Accelerated computing and AI can solve these problems
- **Today:** Enable AI research for all of the above
- **Future:** Hybrid Quantum+AI supercomputer with low-latency link

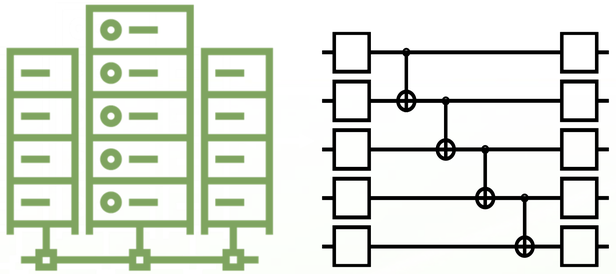
NVIDIA Quantum

Powering the Global Quantum Computing Community

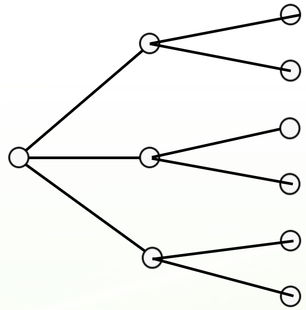


Simulation

Algorithm Design, Resource Estimation, QPU Design



HPC Quantum Integration



AI for Quantum

QEC, Calibration, Algorithms

CUDA-Q

Libraries | Programming Model | Tools | Infrastructure

cuQuantum

Quantum Simulation

DGX-Q

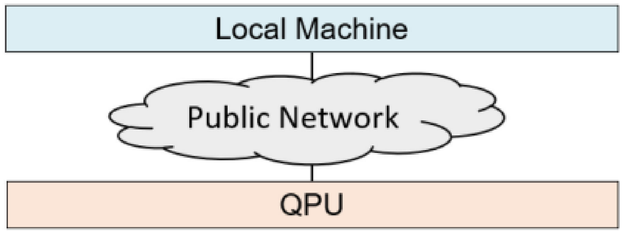
Quantum Integrated Computing

GPU Supercomputing

Motivation behind CUDA-Q

Integrate quantum computers seamlessly with the modern scientific computing ecosystem

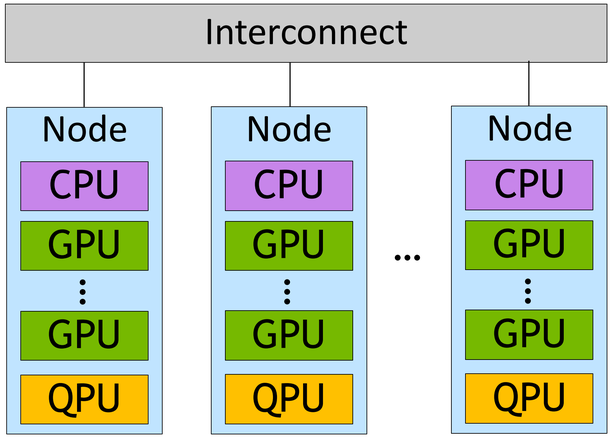
- Quantum computers will accelerate some of today's most important computational problems and HPC workloads
  - Quantum chemistry, Materials simulation, AI
- Want to enable researchers to seamlessly integrate CPUs, GPUs, and QPUs
  - Develop new hybrid applications and accelerate existing ones
  - Leverage classical GPU computing for control, calibration, error mitigation, and error correction
- We also expect CPUs and GPUs to be able to enhance the performance of QPUs
  - Classical preprocessing (circuit optimization) and postprocessing (error correction)
  - Optimal control and QPU calibration
- Research Centers worldwide are focused on integration of quantum computers with classical supercomputers



Quantum Programming Today

Great for early experimentation.

vs…



Where we need to get…

Application-level Quantum Programming

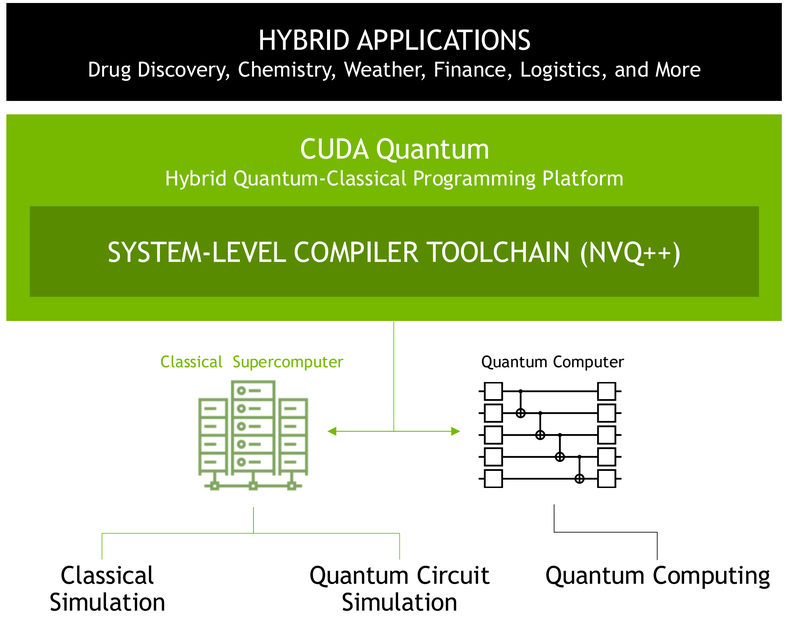
Hybrid quantum-classical applications at scale.

Figure adapted from:
Quantum Computers for High-Performance Computing.
Humble, McCaskey, Lyakh, Gowrishankar, Frisch, Monz.
IEEE Micro Sept 2021. 10.1109/MM.2021.3099140

Introducing CUDA Quantum

Platform for unified quantum-classical accelerated computing

- Programming model extending C++ and Python with quantum kernels
- Open programming model, open-source compiler
  - https://github.com/NVIDIA/cuda-quantum
- QPU Agnostic – Partnering broadly including superconducting, trapped ion, neutral atom, photonic, and NV center QPUs
- Interoperable with the modern scientific computing ecosystem
- Seamless transition from simulation to physical QPU




```
auto ansatz = [](std::vector<double> thetas) __qpu__ {
  cudaq::qreg<3> q;
  x(q[0]);
  ry(thetas[0], q[1]);
  ry(thetas[1], q[2]);
  x<cudaq::ctrl>(q[2], q[0]);
  x<cudaq::ctrl>(q[0], q[1]);
  ry(-thetas[0], q[1]);
  x<cudaq::ctrl>(q[0], q[1]);
  x<cudaq::ctrl>(q[1], q[0]);
};

cudaq::spin_op H = ...;
double energy = cudaq::observe(ansatz, H, {M_PI, M_PI_2});
```

CUDA-Q Academic

Quantum Curriculum and Workforce Development Partnerships

- Coursework designed in partnership with academic institutions
- Access workshops from anywhere with GPU-acceleration in the cloud
- Active learning in distributed, quantum-accelerated computing and CUDA-Q
- Interactive Jupyter notebooks feature lectures, explanations, exercises, and assessments

Arizona State University
Carnegie Mellon University
Dartmouth Engineering
QC Quantum Center
Fordham University
Mälardalen University
Northwestern
Pittsburgh Supercomputing Center
Universitat Politècnica de València
Princeton University
Purdue University
RMU Robert Morris
TUM Chair for Design Automation
Technion Israel Institute of Technology
UC Davis
University of Cambridge Physical Computation Laboratory
The University of Chicago
UF University of Florida
University of Illinois Urbana-Champaign
University of Pittsburgh
Universitat Politècnica de Catalunya BarcelonaTech

Resources

Links

- CUDA-Q Repo for issues and contributions: NVIDIA/CUDA-Quantum (github.com)
- CUDA-Q documentation: CUDA-Q — NVIDIA CUDA-Q documentation
- Quantum computing technical blogs: Tag: Quantum Computing | NVIDIA Technical Blog
- CUDA-Q marketing page: CUDA-Q for Hybrid Quantum-Classical Computing | NVIDIA Developer

Documentation Reference

Quick Start

Multi-GPU Workflows

Simulator backends

Hardware backends

Python code examples

C++ code examples

Applications