OAK RIDGE National Laboratory | LEADERSHIP COMPUTING FACILITY

August 6, 2025

An Introduction to Quantum Computing using Linear Solvers

2025 OLCF User Meeting Hands-On Quantum Computing Training

OLCF Staff

Oak Ridge National Laboratory

U.S. DEPARTMENT OF ENERGY

ORNL IS MANAGED BY UT-BATTELLE LLC
FOR THE US DEPARTMENT OF ENERGY

*This research used resources of the **Oak Ridge Leadership Computing Facility (OLCF) at the Oak Ridge National Laboratory**, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.*

Agenda

Who are we?

Team Members

What is quantum computing?

A brief Intro

What is QLSA?

A brief Intro/Catchup

What quantum resources?

QCUP, IQM & IonQ

Demo Time!

OLCF Staff

Michael Sandoval

Suzanne Parete-Koon

MacKenzie Boyd

Murali Gopalakrishnan Meena

ORNL QCFD team and collaborators

Group website


Kalyan Gottiparthi


Chao Lu


Toño Coello Pérez


Amir Shehata


Michael Sandoval


Seongmin Kim


Pooja Rao
NVIDIA


Justin Lietz
NVIDIA


Xinfeng Gao
UVA


In-Saeng Suh


Antigoni Georgiadou


Alessandro Baroni


Ryan Landfield


Matt Norman


Tom Beck


Paul Lin
NERSC-LBNL


Yu Zhang
LANL

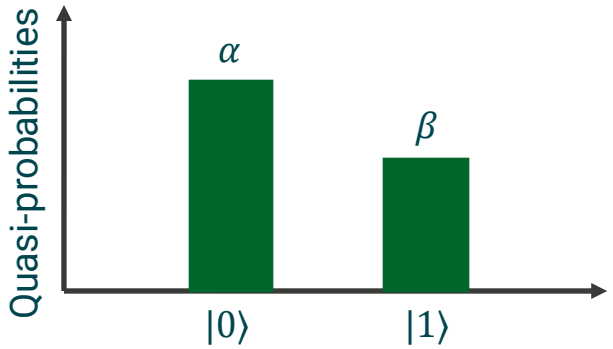
ORNL - OLCF

Quantum Computing

A very brief Intro to QC

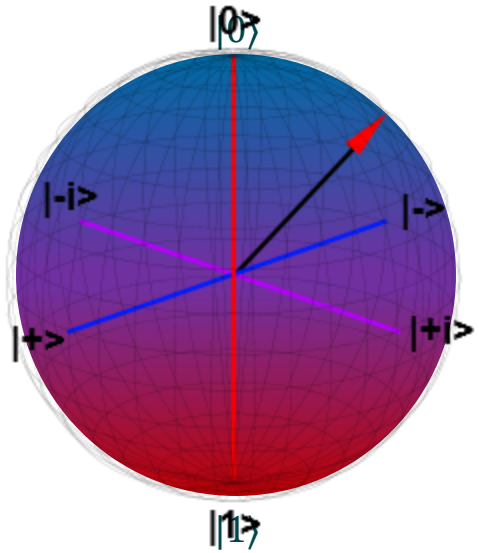
Classical vs Quantum Computing

- Bit vs Qubit – information states
- State of superposition – a combination of all possible configurations
  - Store more information!
  - Measuring will lead to collapse to a binary state
- Types of qubits:
  - Superconducting
  - Trapped ion
  - Photons
  - Neutral atoms
  - Quantum dots



Classical computing - **Bit**

Quantum computing - **Qubit**



$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$

https://www.ibm.com/think/topics/quantum-computing
Michael Sandoval's Quantum Computing Primer
Rick Arthur's Quantum Computing 101 briefing

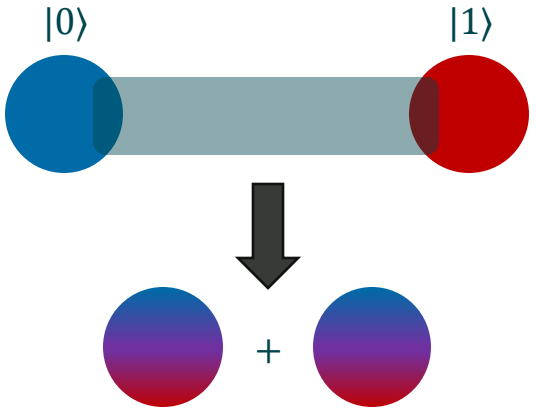
Key principles of quantum computing

1. **Superposition**

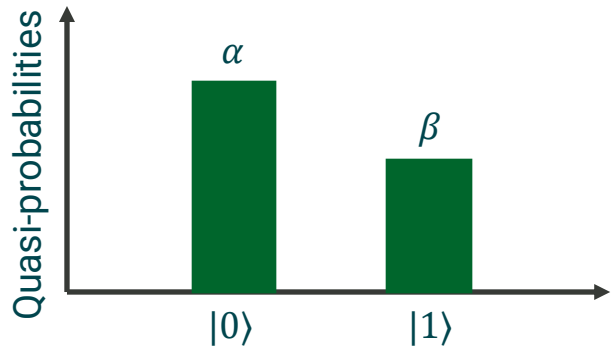
2. **Entanglement** – ability of qubits to correlate with each others' states
   - Store even more information!

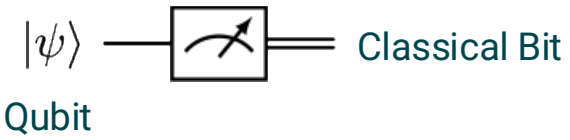
3. **Interference** – information is structured like waves with amplitudes
   - Waves can amplify or cancel each other
   - Amplitudes: Probabilities of the outcomes of measurement

4. **Decoherence** – collapse from quantum to nonquantum state
   - Intentionally (measurement)
   - Allows quantum computers to interact with classical computers
   - Unintentionally (interaction with environment)



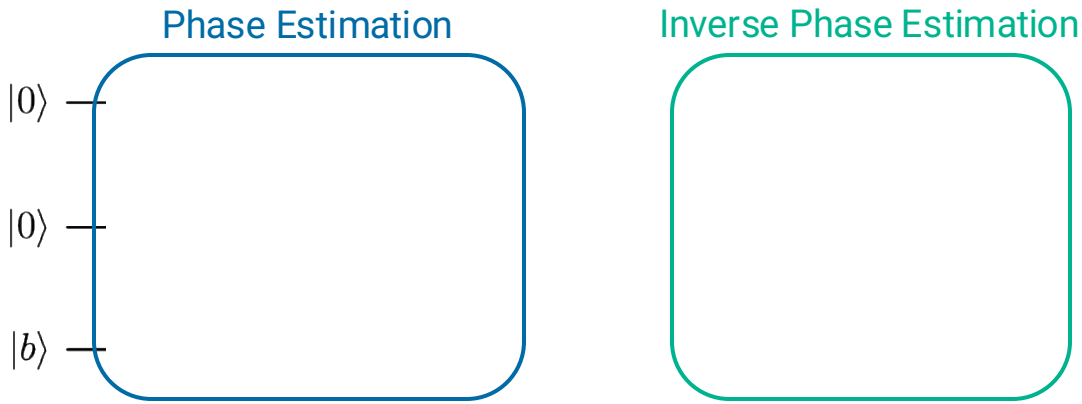
$$|\psi\rangle = \alpha_1|00\rangle + \alpha_2|01\rangle + \alpha_3|10\rangle + \alpha_4|11\rangle$$





Working of a quantum computer

1. **Qubits** are prepared as superposition of states
2. **Gates** used to operate on qubits and entangle them
   - Unitary operations
   - Reversible
   - Can operate on a single qubit or multiple (entanglement)
3. **Circuits** are collection of gates
4. **Quantum algorithms** are collection of circuits to create desired interference between states
5. **Measurement** (amplified outcomes) gives solution



Running a quantum algorithm

Backend	Mechanism	Functionality
Simulator	Classical	Classical program modeling a quantum system in an ideal scenario
Emulator	Classical	Classical program modeling actual behavior of a quantum system
Real	Quantum	Physical hardware performing real quantum computations

Quantum Linear Solvers

A brief Intro/Catchup

Governing equations are discretized to create a set of algebraic equations & assembled into $Ax = b$

$$\nabla \cdot \boldsymbol{u} = 0,$$

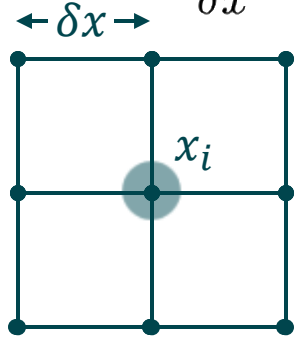
$$\frac{\partial \boldsymbol{u}}{\partial t} = -\underbrace{\boldsymbol{u} \cdot \nabla \boldsymbol{u}}_{\text{nonlinear term}} \underbrace{-\nabla p + \frac{1}{Re}\nabla^2 \boldsymbol{u}}_{\text{linear terms}} + \boldsymbol{s}$$

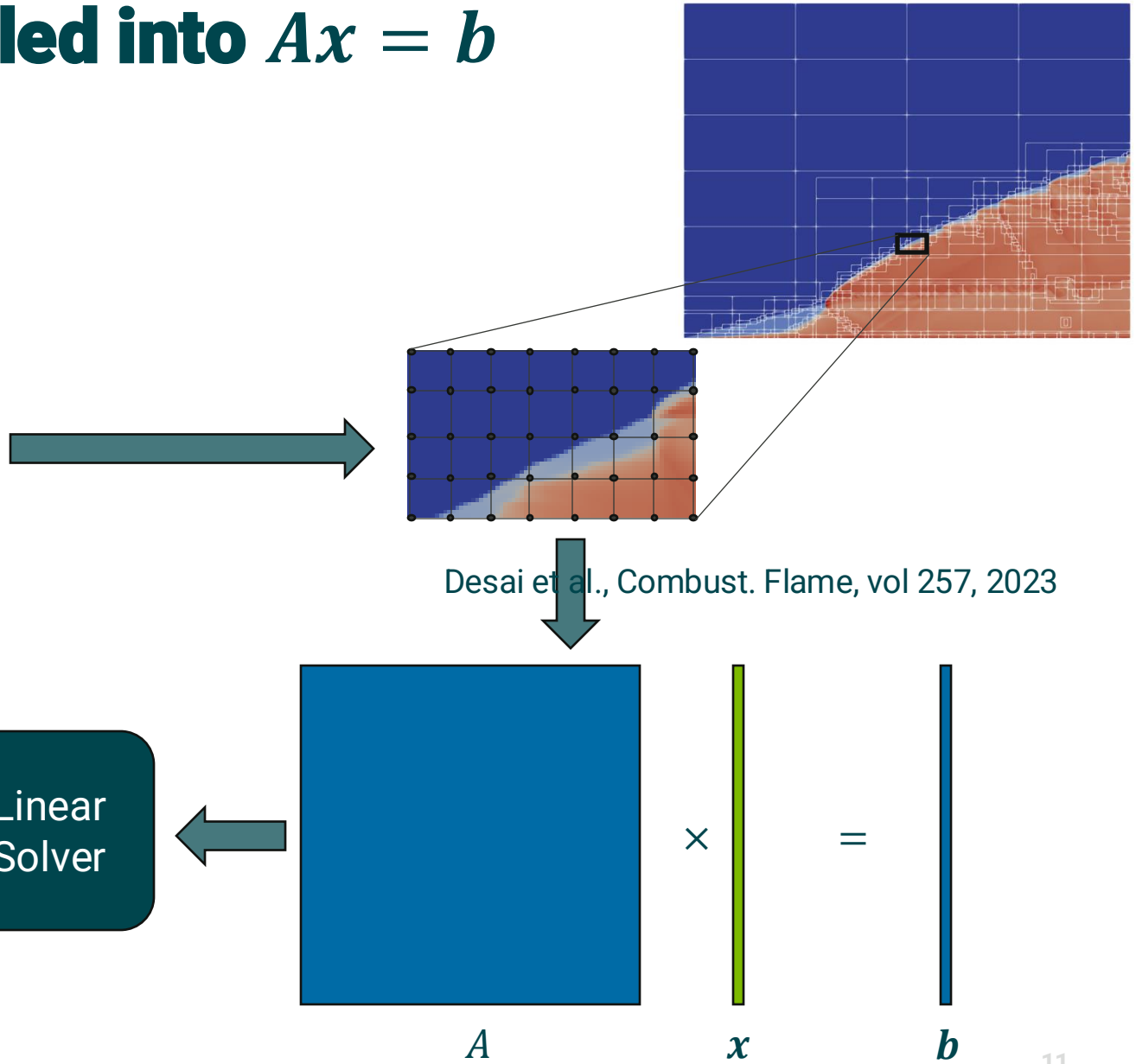
$$\nabla^2 p = -\underbrace{\nabla \cdot (\boldsymbol{u} \cdot \nabla \boldsymbol{u})}_{\text{nonlinear term}}$$

Finite difference approximation

$$\frac{\partial u_\alpha(x_i)}{\partial x} \simeq \frac{u_\alpha(x_i + \delta x) - u_\alpha(x_i - \delta x)}{2\delta x}$$

$$\frac{\partial^2 u_\alpha(x_i)}{\partial x^2} \simeq \frac{u_\alpha(x_i + \delta x) + u_\alpha(x_i - \delta x) - 2u_\alpha(x_i)}{\delta x^2}$$



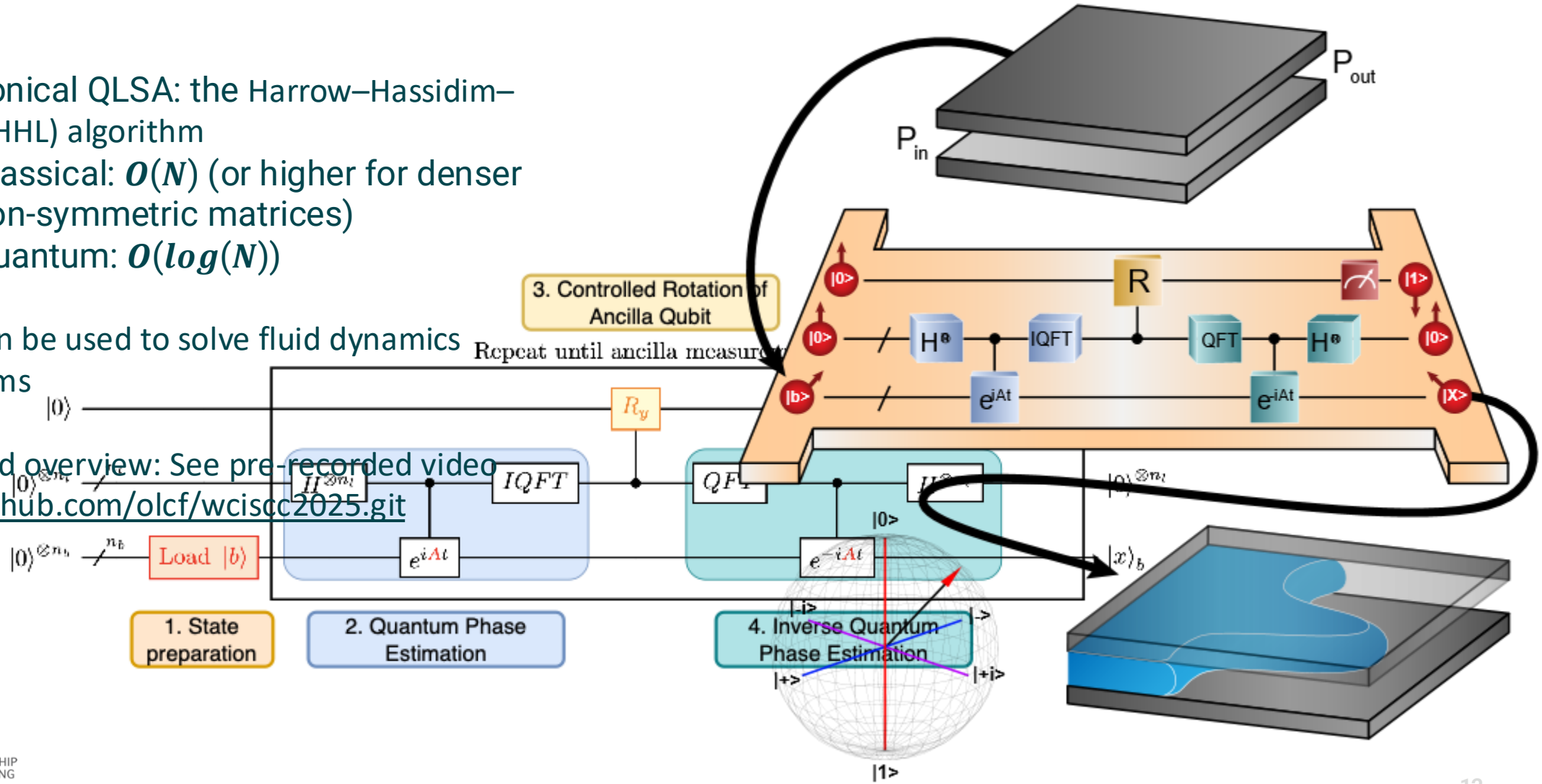


Quantum Linear Systems Algorithms (QLSA) solve $Ax = b$ problems using quantum computers

- A canonical QLSA: the Harrow–Hassidim–Lloyd (HHL) algorithm
  - Classical: $O(N)$ (or higher for denser non-symmetric matrices)
  - Quantum: $O(log(N))$

- HHL can be used to solve fluid dynamics problems

- Detailed overview: See pre-recorded video https://github.com/olcf/wciscc2025.git



Quantum Hardware Access with QCUP

IQM Resonance & IonQ Cloud

Quantum Computing User Program

Enable Research

Provide a broad spectrum of user access to the best available quantum computing systems

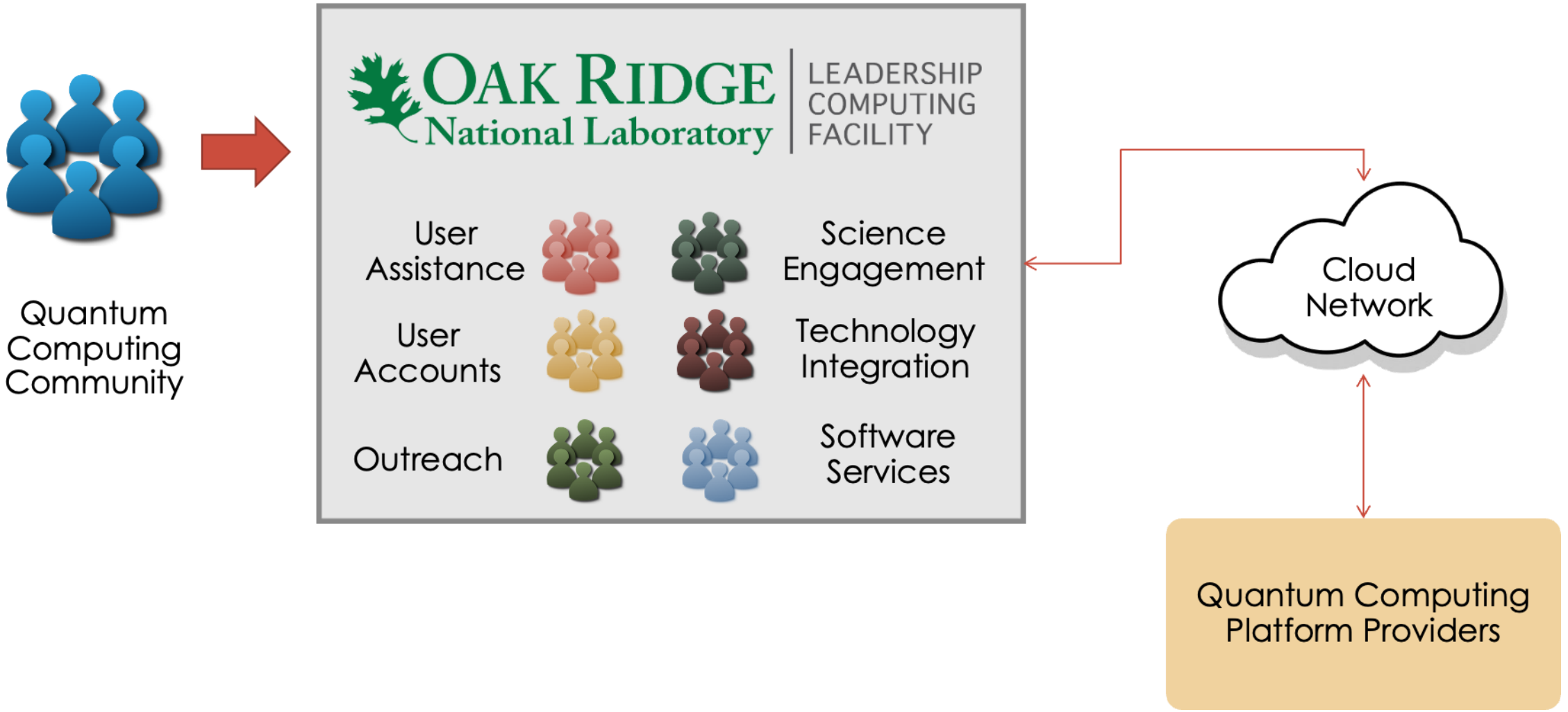
Evaluate Technology

Monitor the breadth and performance of early quantum computing applications

Engage Community

Support growth of the quantum ecosystem by engaging with users, developers, vendors, and providers

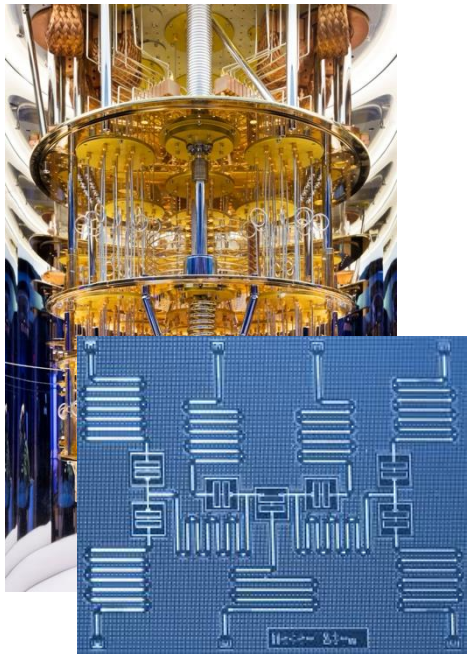
QCUP Operations Model: Cloud Access



Multiple Quantum Computing Resources

IBM Quantum

General-purpose trasnmon systems provide up to 156 qubits

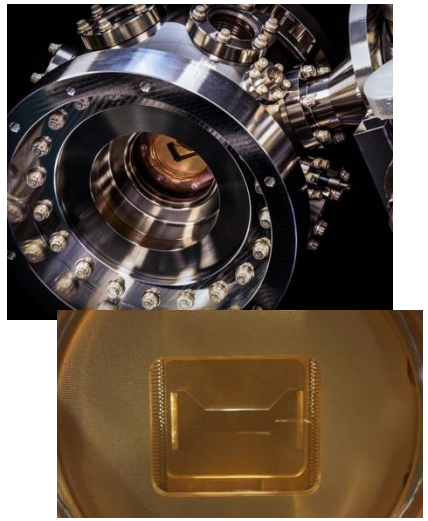


IQM

General-purpose trasnmon systems provide up to 56 qubits

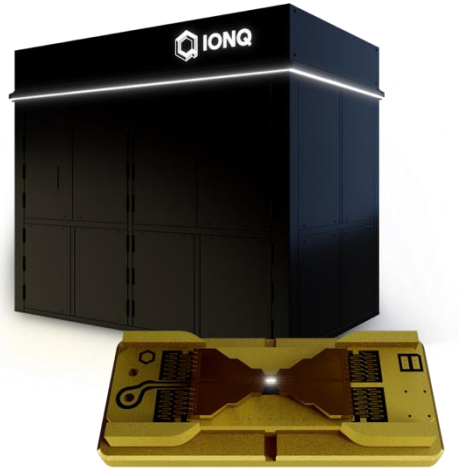
Quantinuum

General-purpose ion trap systems provide up to 56 qubits



IonQ

General-purpose ion trap systems provide up to 36 qubits

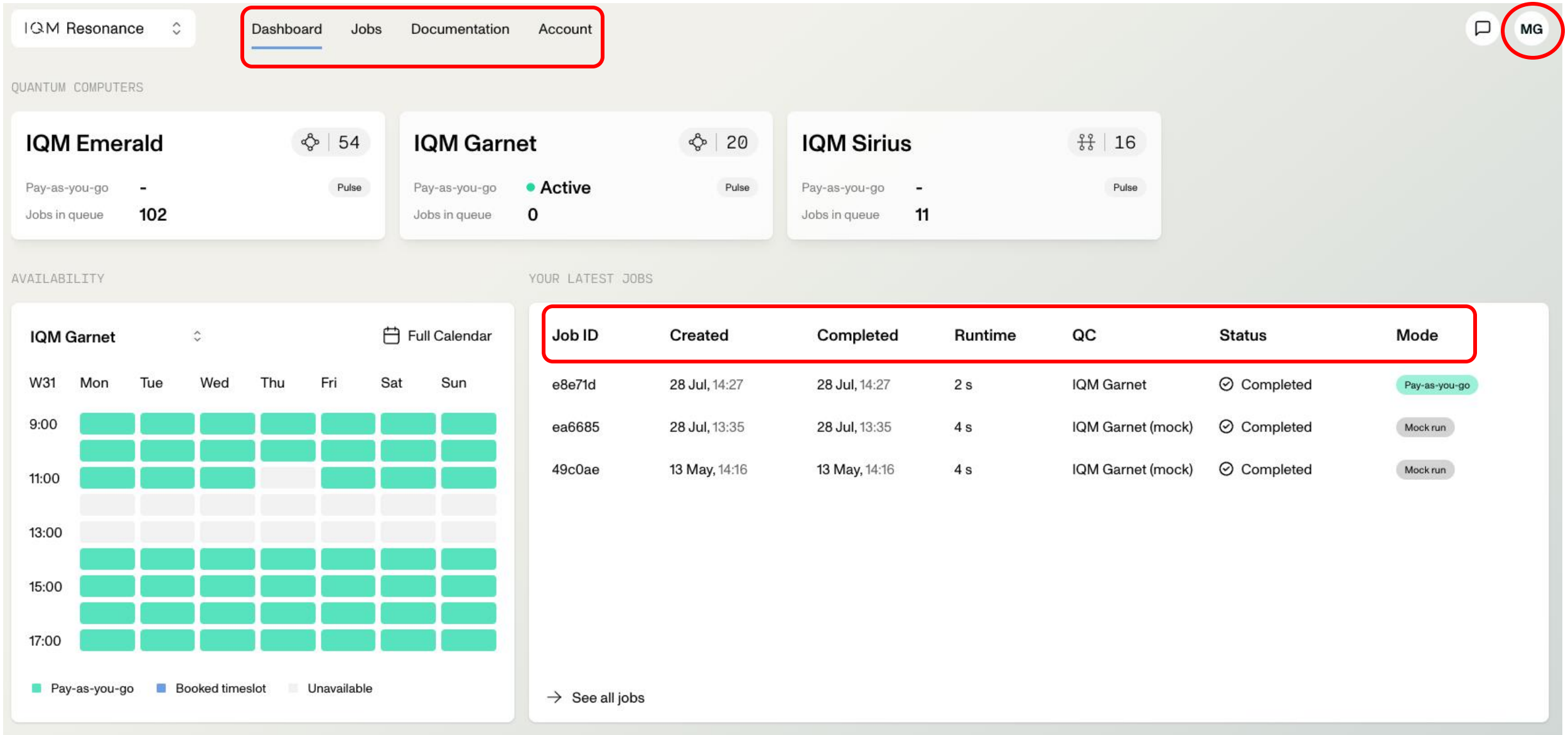


Current system count is 16

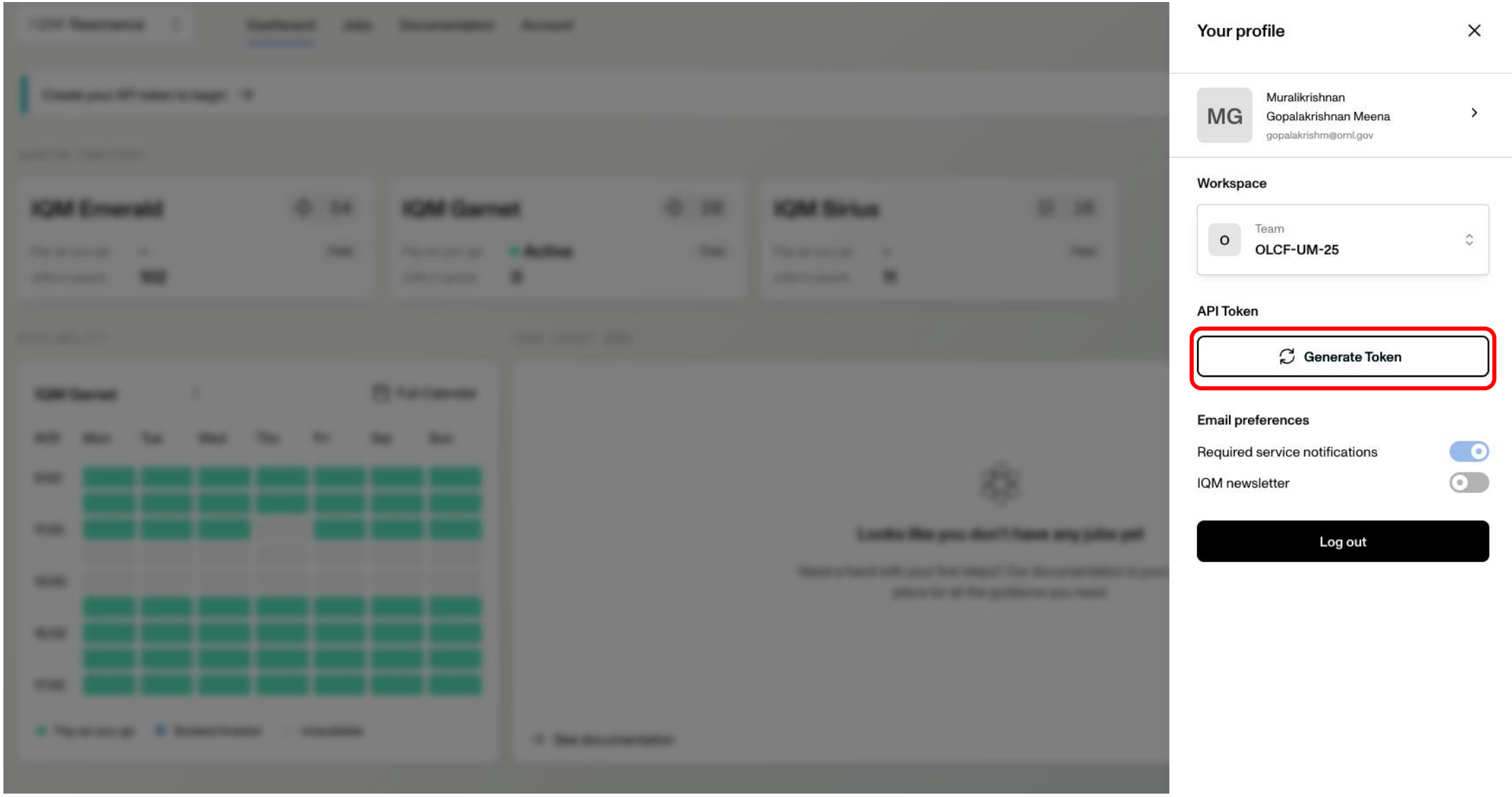
IQM Resonance

- IQM provides access to quantum computing resources through their cloud platform, IQM Resonance.
  - The 20 qubit **Garnet,** 16 qubit Sirius, and 54 qubit Emerald systems.
  - IQM's quantum processors are made up of superconducting transmon qubits, utilizing multiple topologies.
  - User dashboard: https://resonance.meetiqm.com/
- OLCF Documentation: https://docs.olcf.ornl.gov/quantum/quantum_systems/iqm.html
- We'll be submitting jobs to Garnet via Python from Frontier
  - "Pay-as-you-go" mode
    - Certain windows where a queue is open
  - "Timeslot" mode
    - Reserved for today from 2-3 PM (Eastern)
- Should've received an invite from me to create your account (**not the same account as Frontier!!**)

The IQM Resonance Dashboard - I



The IQM Resonance Dashboard - II

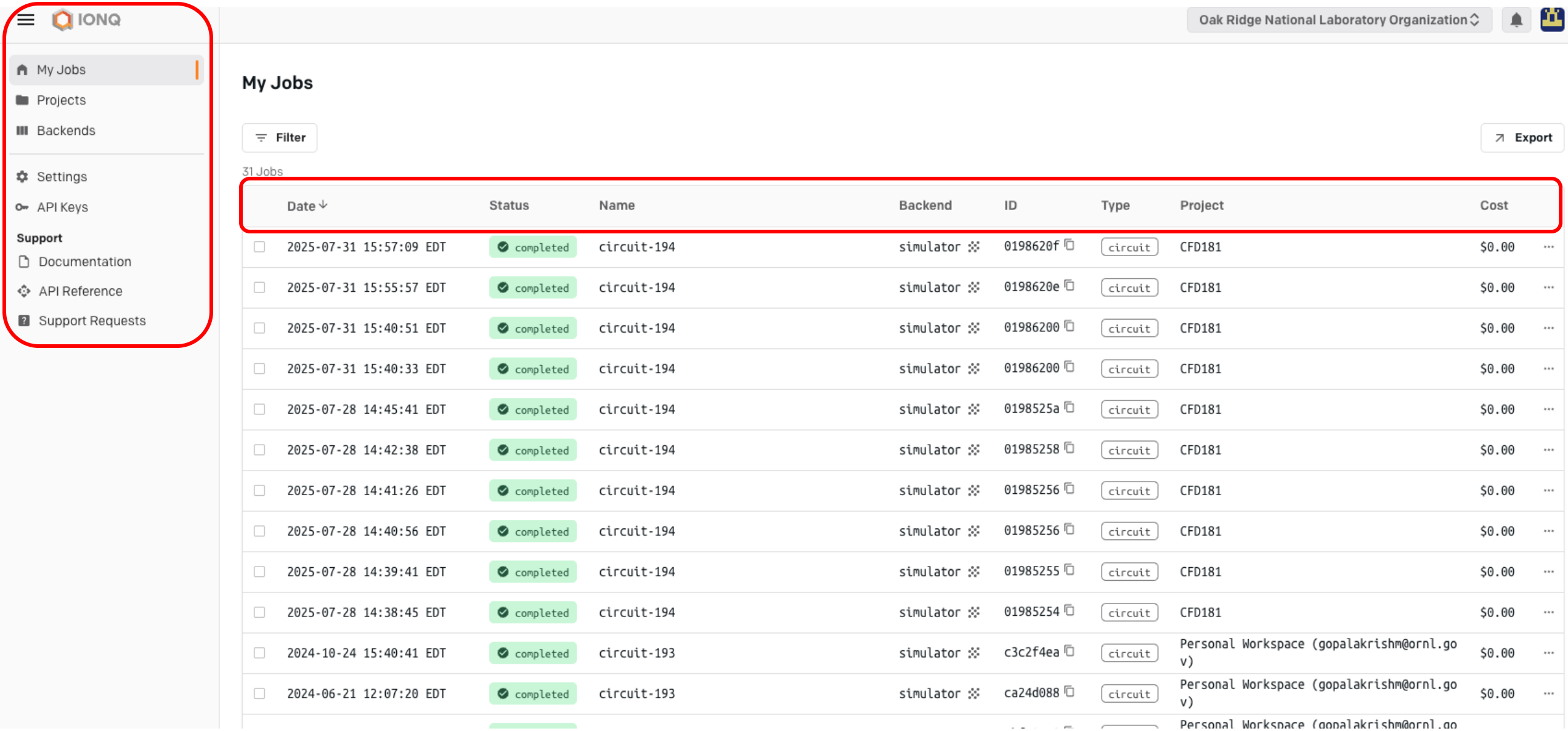


The IQM Resonance Dashboard - III

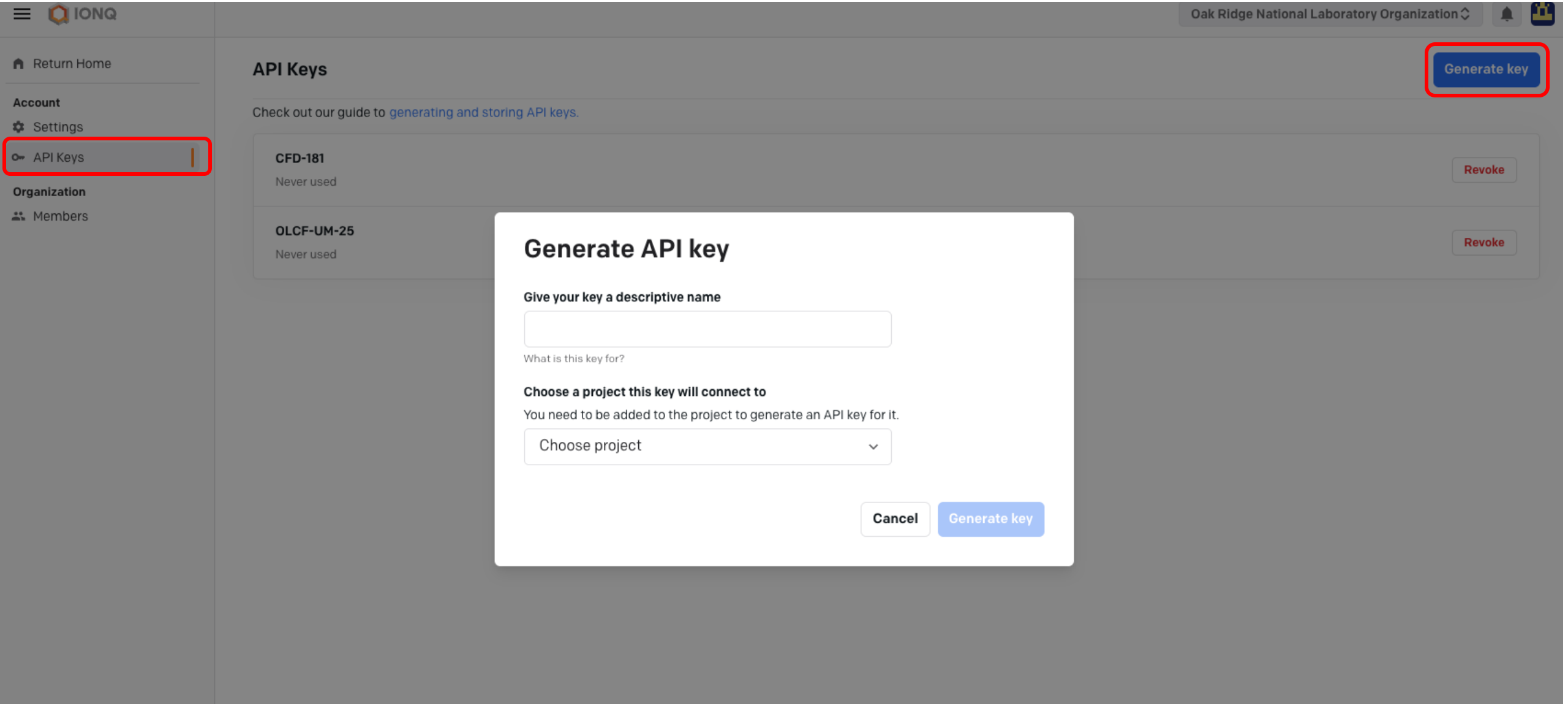
IonQ Cloud Console

- IonQ provides access to quantum computing resources through their cloud platform, IonQ Quantum Cloud.
  - The 25 qubit **Aria** and 36 qubit Forte systems.
  - IonQ's quantum processors are made up of trapped-ion qubits, utilizing multiple topologies.
  - User dashboard: https://cloud.ionq.com/
- OLCF Documentation: https://docs.olcf.ornl.gov/quantum/quantum_systems/ionq.html
- We'll be submitting jobs to Aria via Python from Frontier
  - Run using credits
    - Priority window for today from 2-3:30 PM (Eastern)
    - Limited credits for this tutorial – **only run up to 500 shots**
- Should've received an invite from me to create your account (**not the same account as Frontier!!**)

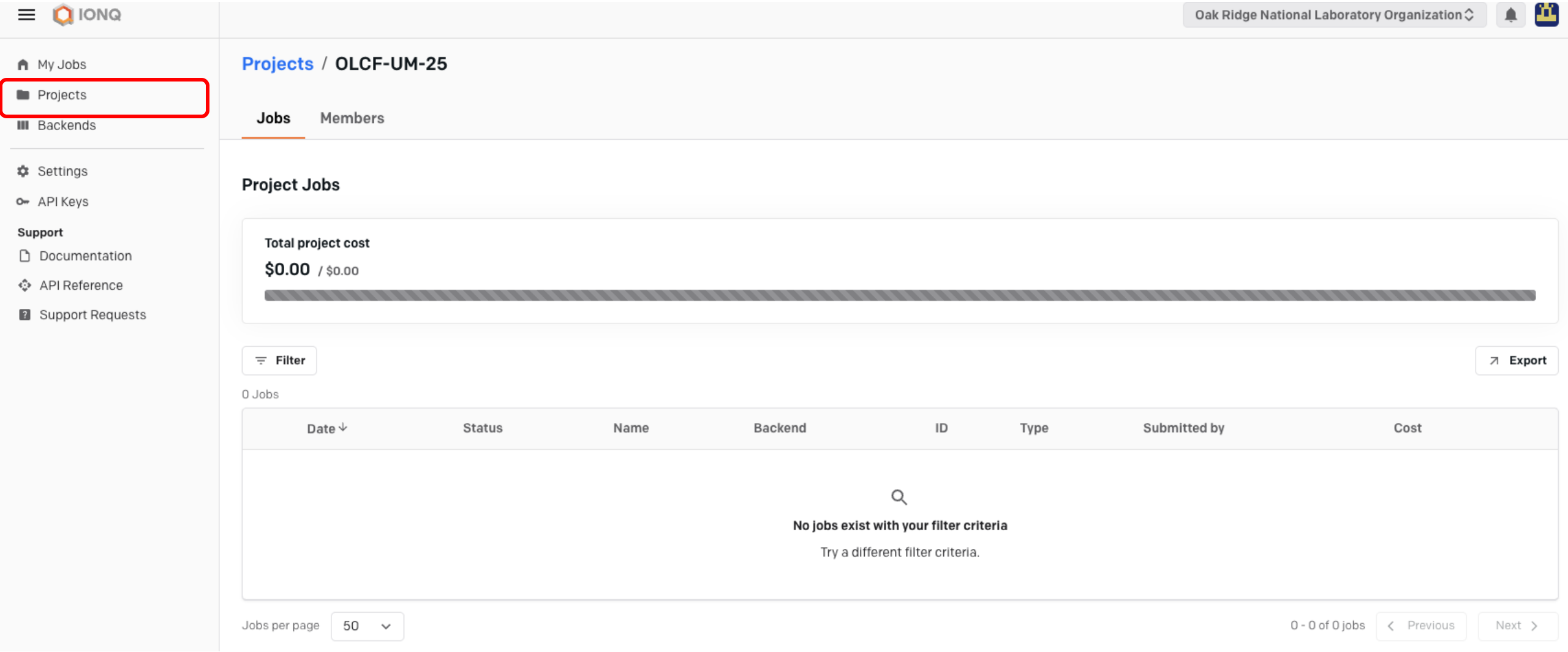
The IonQ Cloud Dashboard - I



The IonQ Cloud Dashboard - II



The IonQ Cloud Dashboard - III



The Hands-On Tutorial

Demo & Challenge

Run the HHL circuit on simulators, emulators, and real devices

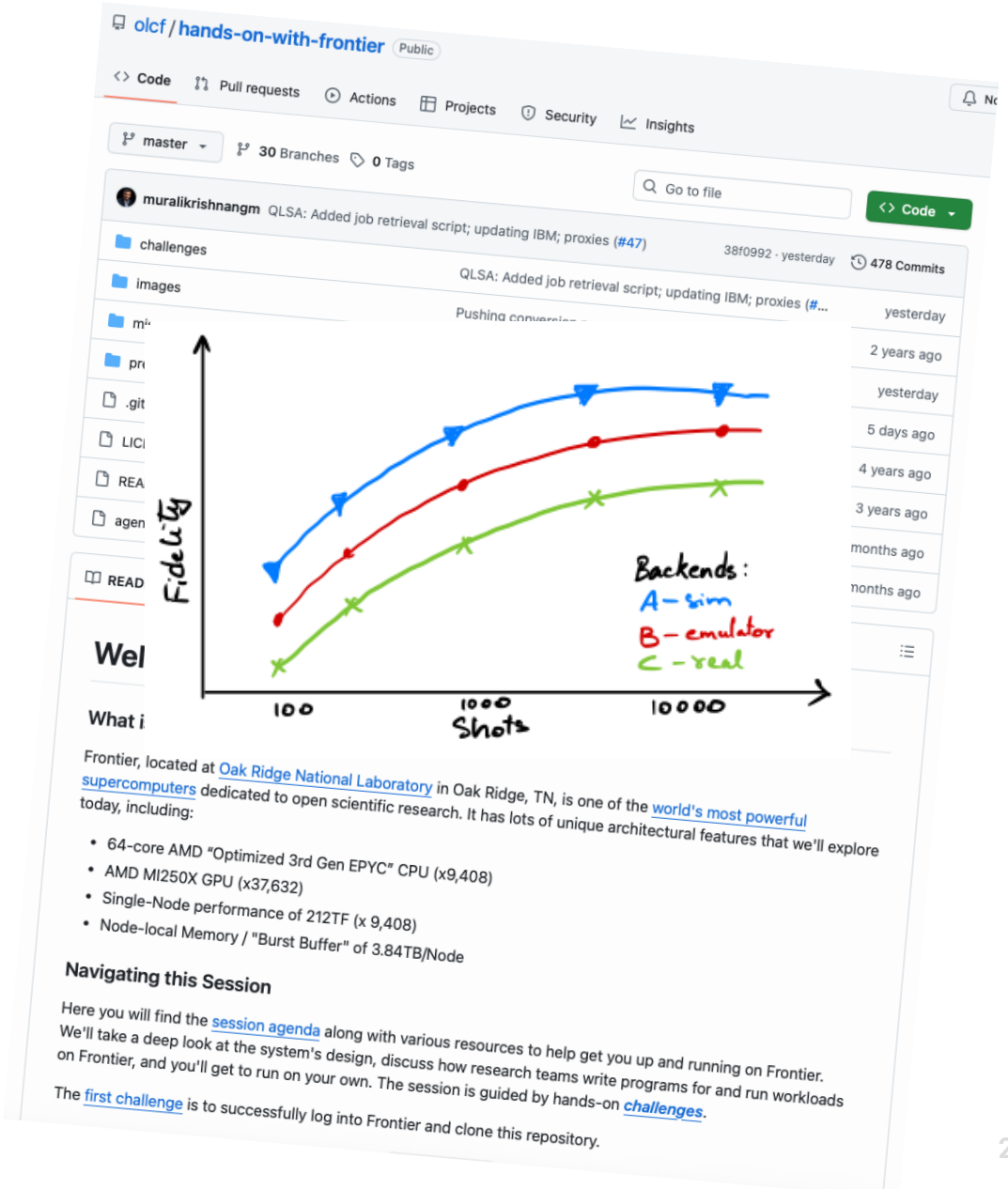
Demo Time!

https://github.com/olcf/hands-on-with-frontier

- Go to: challenges/Python_QLSA

Challenge Time! Two Tasks:

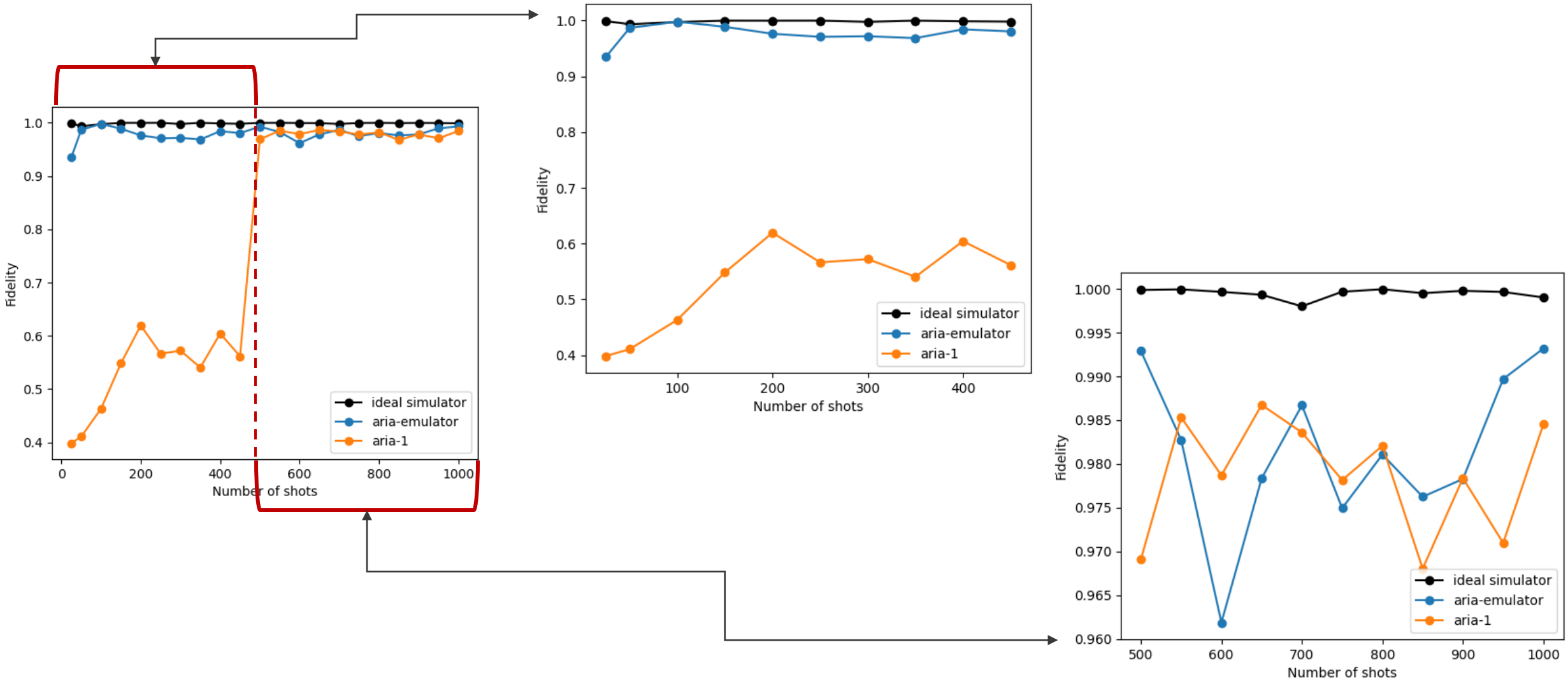
1. Shots-based study
2. Backend evaluation



Logistics

1. When requesting Frontier node,
    1. Use the reservation flag: **--reservation=usermeeting2**
    2. Please request only **1 node** per user
2. Due to limited resources, **IonQ runs are capped at 500 shots** – try to use the resource sparingly

Sample results of running on IonQ



Run the HHL circuit on simulators, emulators, and real devices

https://github.com/olcf/hands-on-with-frontier

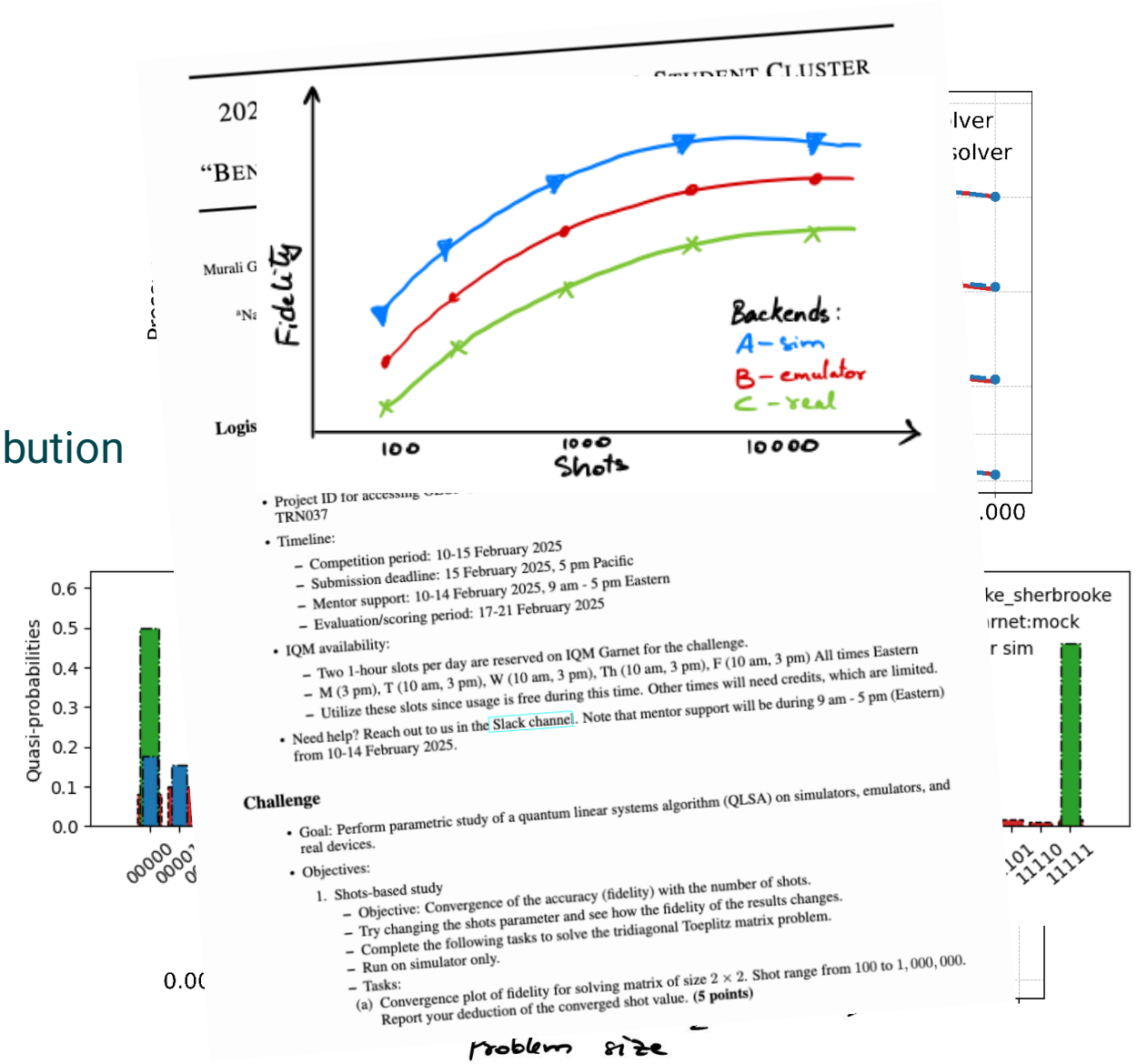
- Go to: challenges/Python_QLSA

Explore more!

1. Shots-based study – uncertainty in results
2. Backend evaluation – quasi-probability distribution
3. *Bonus: Solve the Hele–Shaw flow*

Try out our 2025 WCISCC student challenge!

https://github.com/olcf/wciscc2025.git




Questions?
https://github.com/olcf/hands-on-with-frontier