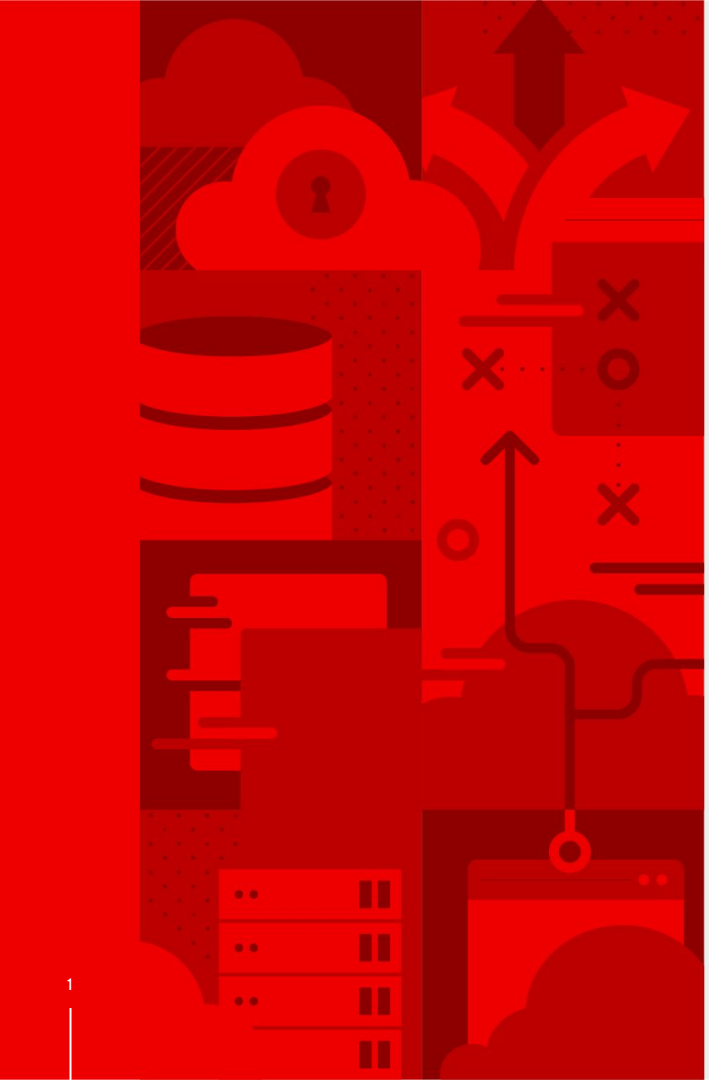


Quantum Computing on OpenShift

Parul Singh
Software Engineer
Red Hat

Ismael Faro
Tech Lead Cloud
IBM Quantum

Luciano Bello
Qiskit Dev team
IBM Quantum

IBM Quantum

Red Hat

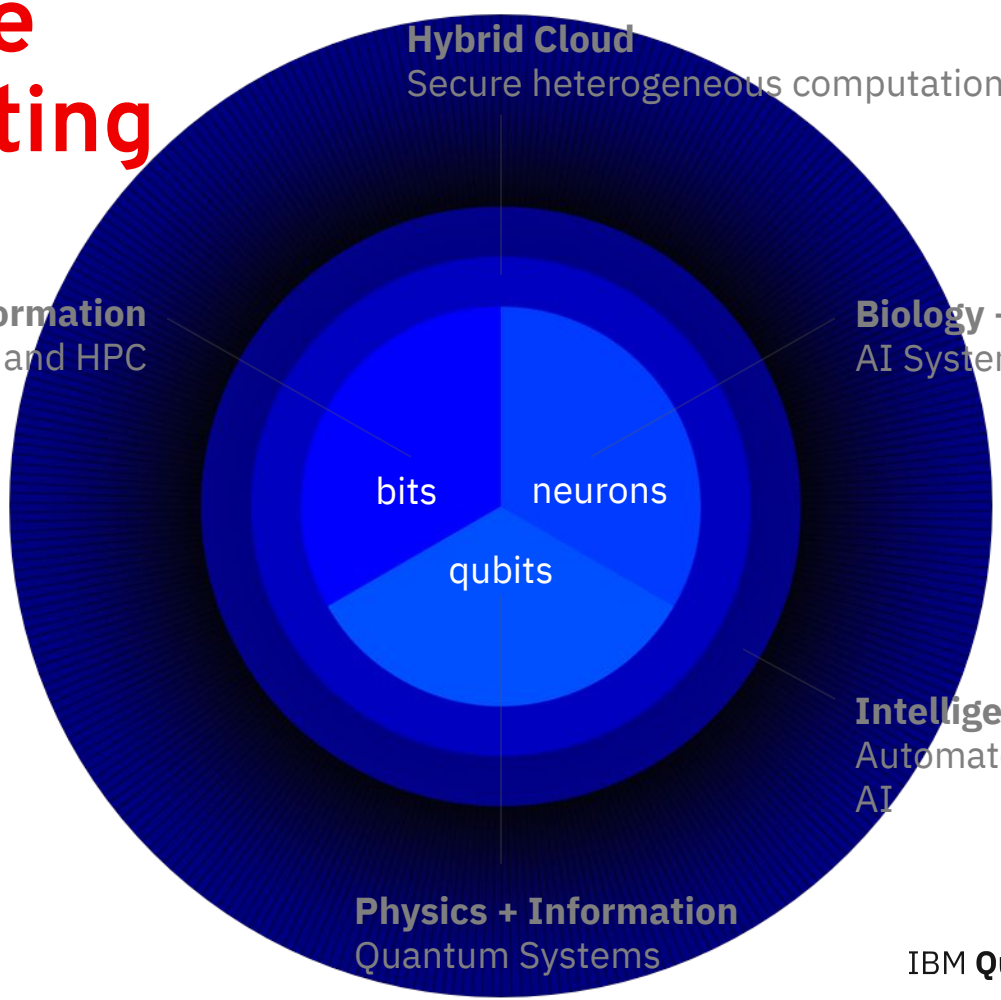
Agenda

Introduction to Quantum Computing
Future of computing
Key Concepts

Qiskit Operator
Demo how to launch a development env for implementing quantum algorithm using Qiskit & OpenShift

Example circuit
Qiskit developer's from IBM will demo how to implement circuits using Qiskit

The future of computing



IBM Quantum Red Hat

The world's most powerful "bits + neurons" system

Oak Ridge National Laboratory
US Department of Energy

Summit supercomputer specs

200,000
trillion calculations
per second

9216
IBM Power 9 processors

27,648
NVIDIA GPUs

250 PB
File System

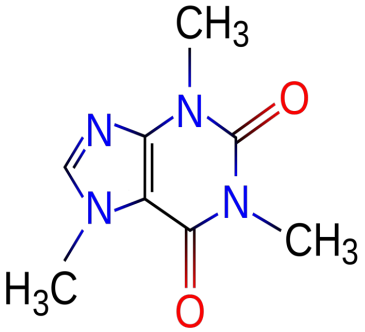
IBM Red Hat Enterprise Linux (RHEL) v 7.4
Operating System

IBM Quantum Red Hat

Are there still intractable problems?

Bits & Qubits

It's *impossible* to completely represent the molecular configuration of caffeine on *today's most powerful supercomputers*, but we could represent it using *160 qubits*



$C_8H_{10}N_4O_2$

6

How many bits do we need for caffeine?

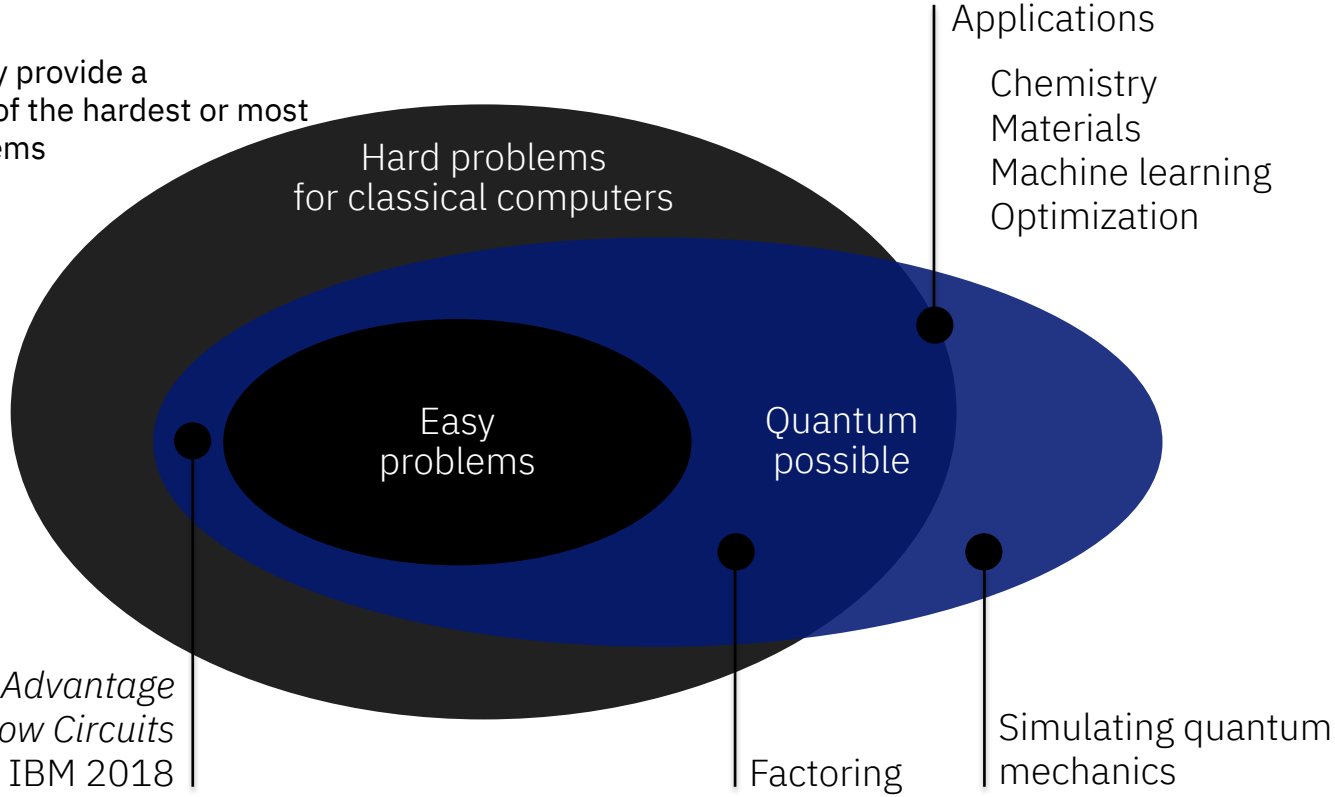
We need approximately 10^{48} bits to represent the energy configuration of a single caffeine molecule at a single instant.

This is 1 to 10% of the total number of atoms in the Earth.

10^{48} = 1,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000

Hard versus easy problems

Quantum computing may provide a new path to solve some of the hardest or most memory intensive problems in business and science.

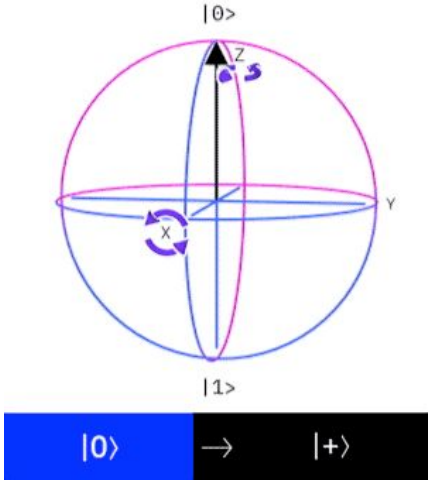


IBM **Quantum** Red Hat

Quantum Computing Key Concepts

Red Hat

Key Concepts

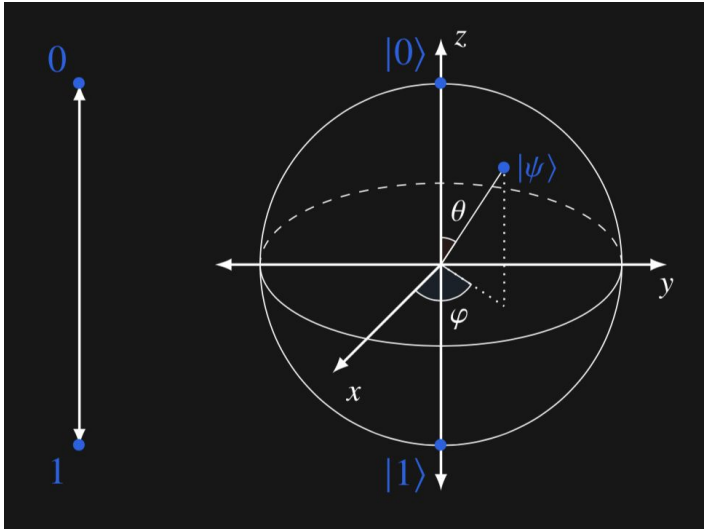


Bits can be **0** or **1**

Quantum bits, or *qubits*, can take on those values but can represent a combination of **|0>** and **|1>** while we are computing

Superposition is creating a quantum state that is a combination of |0> and |1>

A Hadamard gate H on qubit 0, which puts it into a superposition state

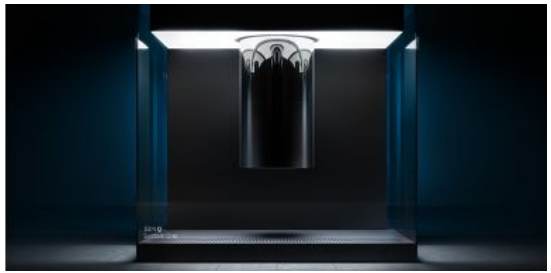


Entanglement strongly connects two or more qubits so that their quantum states are no longer independent

A controlled-Not operation (CX) on control qubit 0 and target qubit 1, putting the qubits in an entangled state

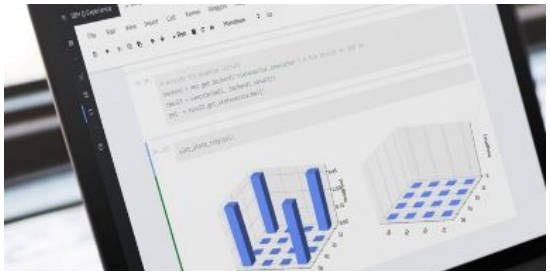
IBM Quantum

Quantum Systems



- Lead the world in application systems (29 Devices)
- Lead the world in quantum research

Quantum Cloud and Software

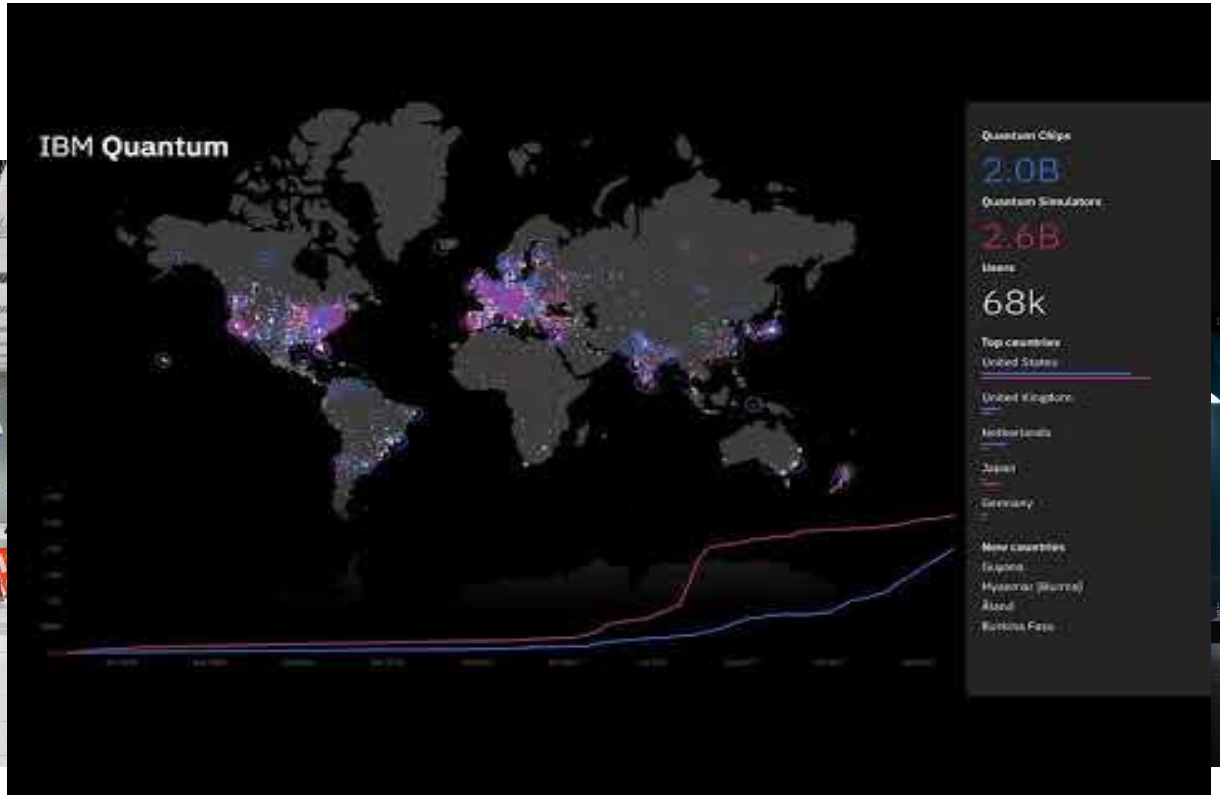


- Open source projects (Qiskit)
- IBM Quantum Services, Quantum Experience, Quantum Lab, Systems access

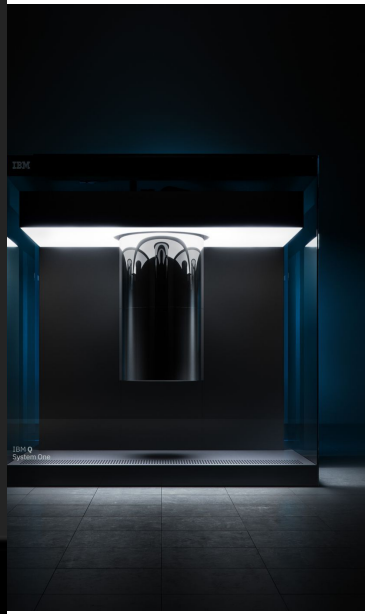
Quantum Community

- Education, Researcher, Developers, Business
- Lead quantum software ecosystem development

Research



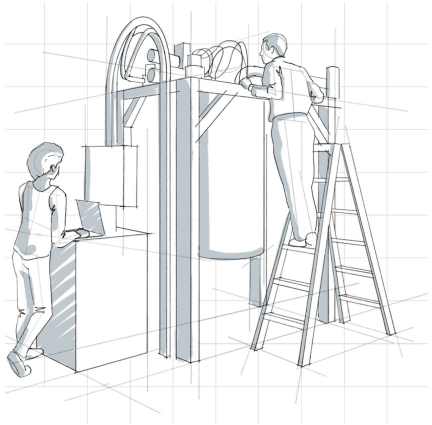
Production



IBM Quantum Red Hat

Qiskit

Qiskit [quiss-kit] is an open source SDK for working with quantum computers at the level of pulses, circuits and algorithms.



- Application, Programs
- Patterns and Circuits Library
- Transpiler (Analyze, Synthesize, Map, Optimize)
- Experimentalist tools and Pulses tools
- Simulators
- IBMQ Provider

Qiskit.org

@qiskit

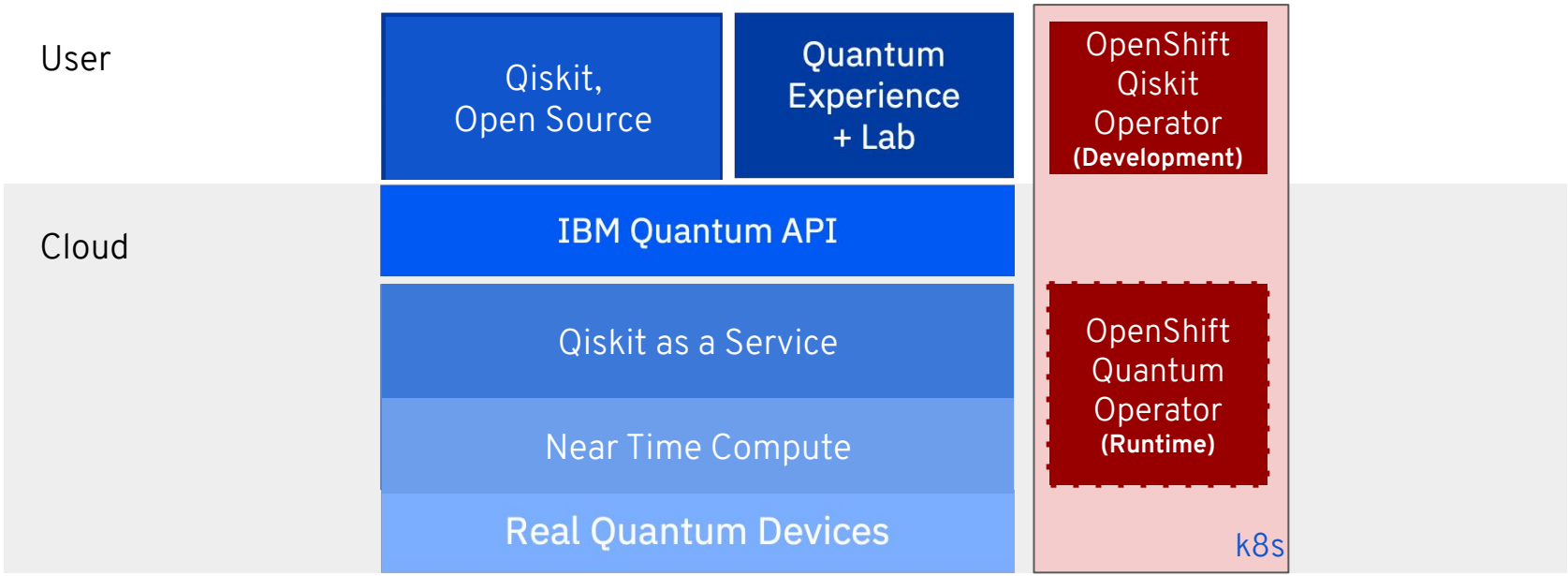
Red Hat Quantum Computing Roadmap

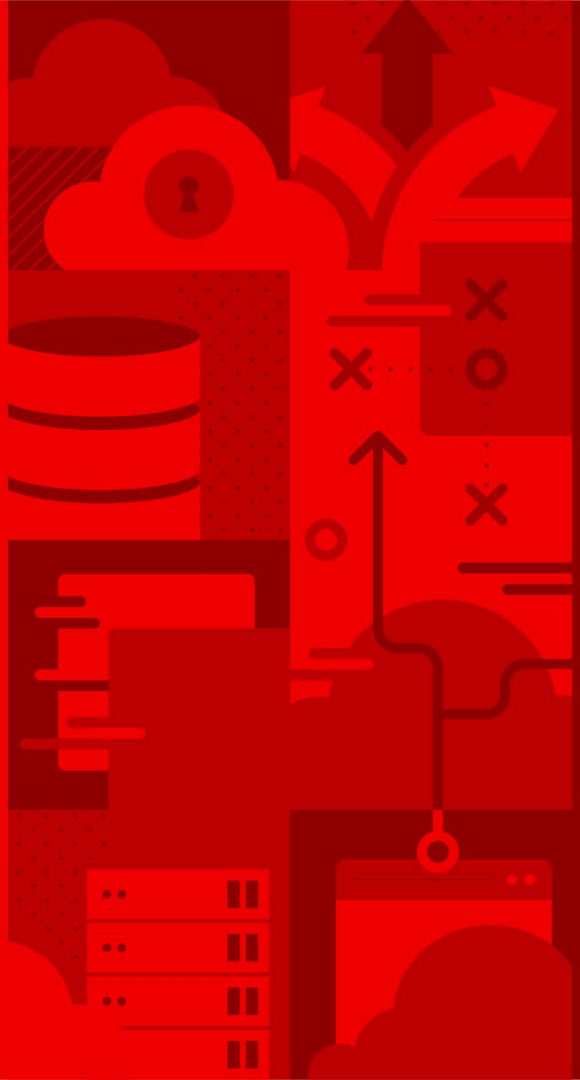
Our general goal

- Define how the **classical and the quantum can be connected together**
- **OpenShift** provides best of classical computing environments
- IBM Research has developed quantum computers based on superconducting qubits

Prototyping to define the best practices for running heterogenous workflows in a co processor model using OpenShift and IBM Quantum Services

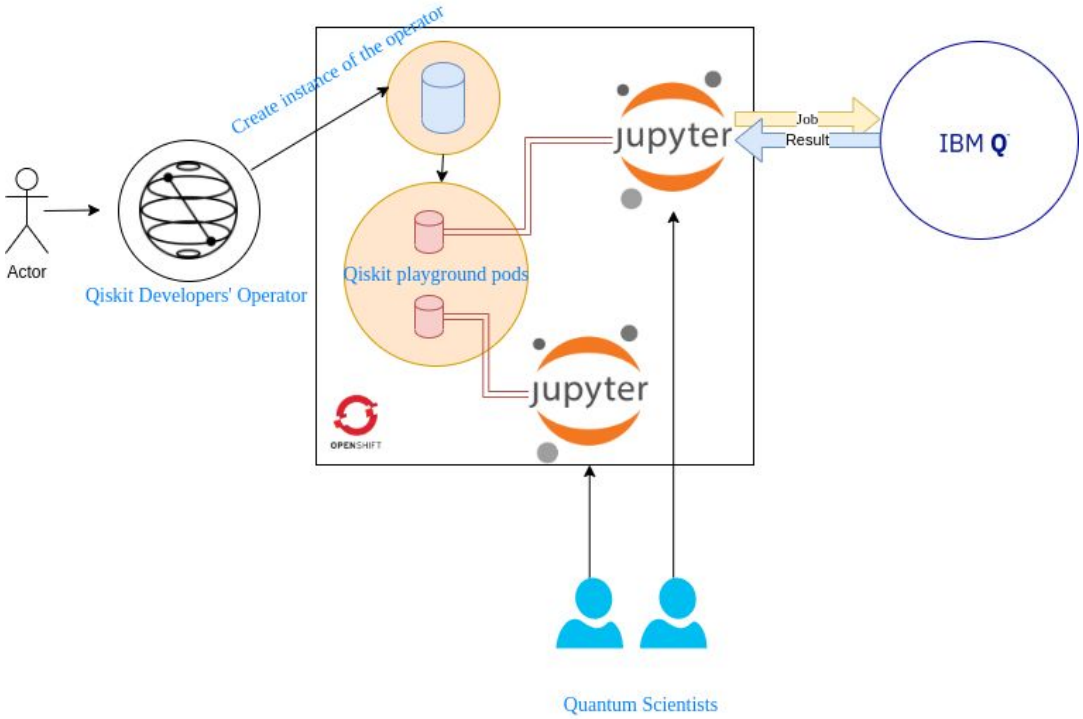
IBM Quantum software Stack





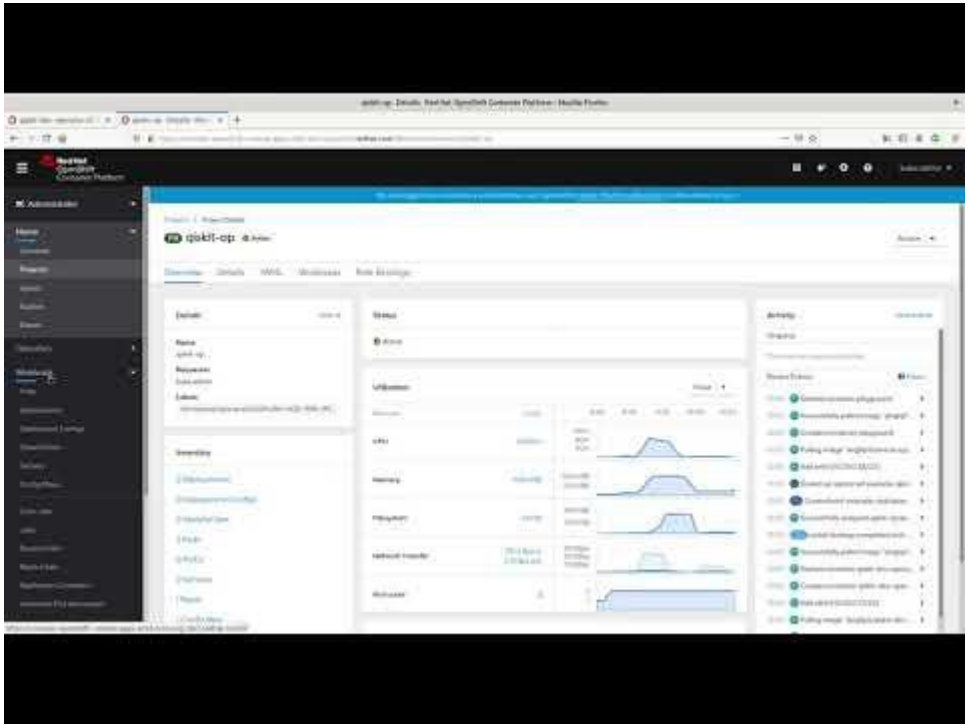
Demos -OpenShift Qiskit operator

Developing Quantum Circuits on OpenShift using Qiskit Operator



Launches development environment with all the dependencies pre installed

OpenShift Qiskit Operator Demo



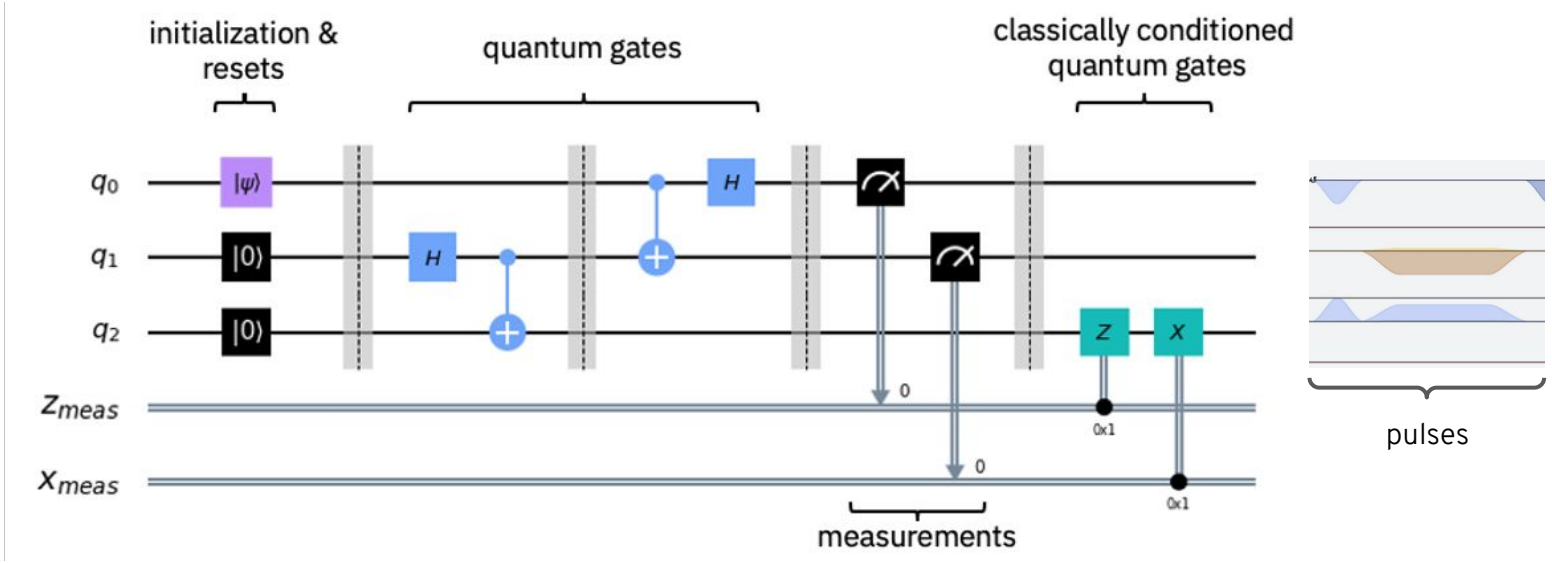
IBM Quantum Red Hat

Quantum circuits

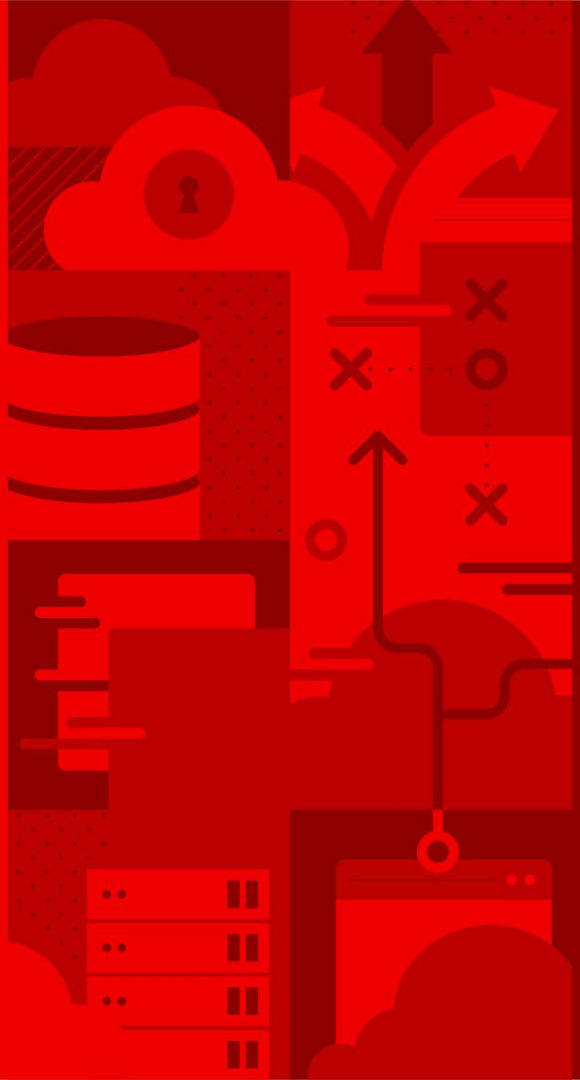
The quantum building blocks

IBM Quantum Red Hat

Pulses, Gates and Circuits

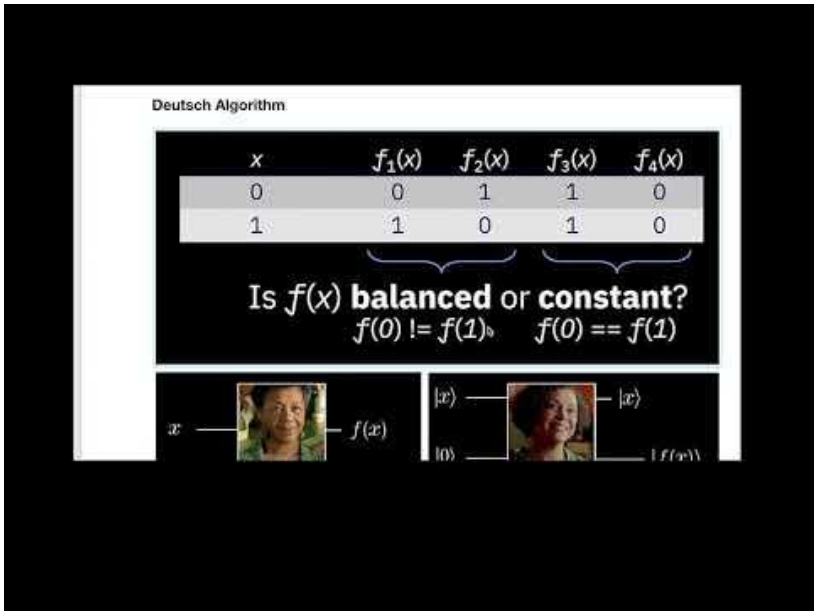


A quantum circuit is a computational routine consisting of coherent quantum operations on quantum data, such as qubits, and concurrent real-time classical computation.



Developing circuits in Qiskit

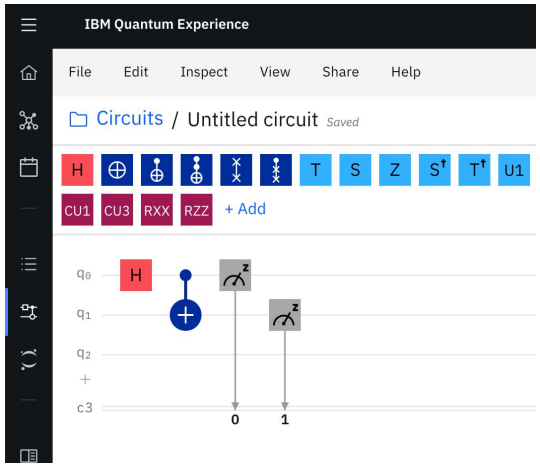
Learn Quantum



Learn Quantum

Quantum Experience

quantum-computing.ibm.com



Qiskit

qiskit.org

```
import numpy as np
from qiskit import(
  QuantumCircuit,
  execute,
  Aer)
from qiskit.visualization import plot_histogram

# Use Aer's qasm_simulator
simulator = Aer.get_backend('qasm_simulator')

# Create a Quantum Circuit acting on the q register
circuit = QuantumCircuit(2, 2)

# Add a H gate on qubit 0
circuit.h(0)

# Add a CX (CNOT) gate on control qubit 0 and target qubit 1
circuit.cx(0, 1)

# Map the quantum measurement to the classical bits
circuit.measure([0,1], [0,1])

# Execute the circuit on the qasm simulator
job = execute(circuit, simulator, shots=1000)
```

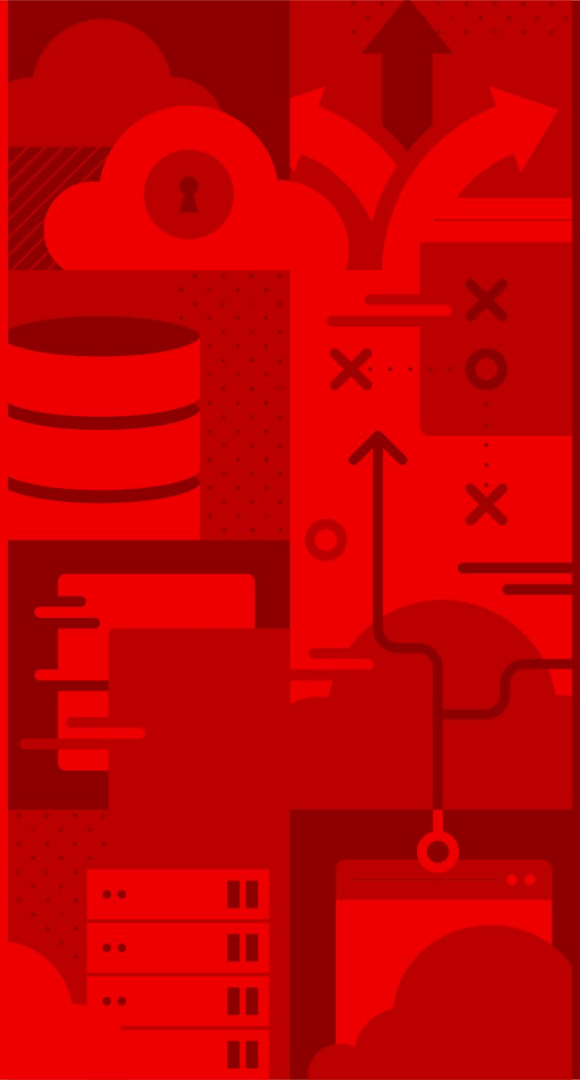
Resources

Project Repo : https://github.com/qiskit-community/openshift-quantum-operators

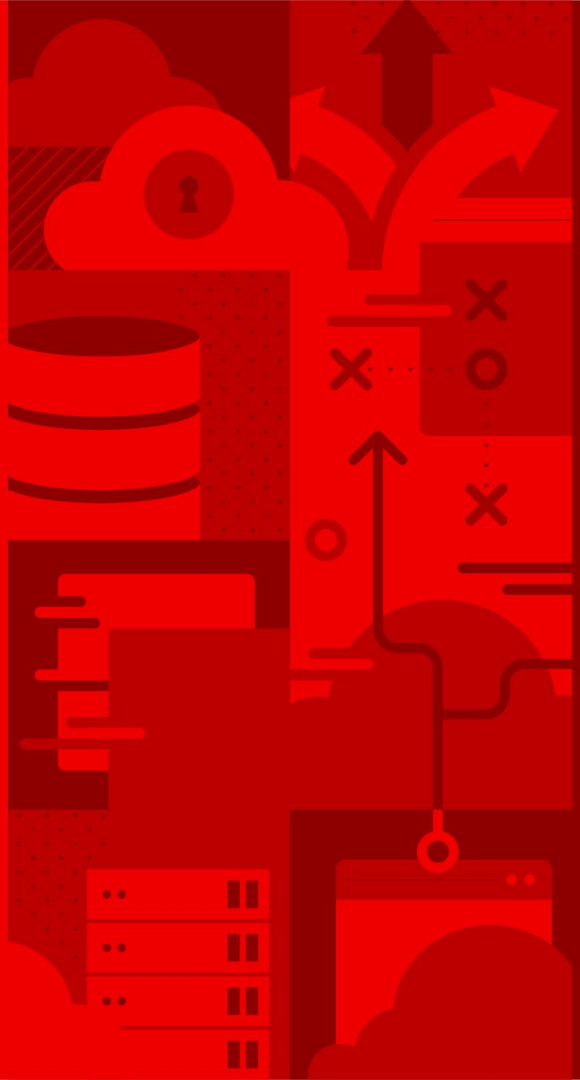
IBM Quantum Experience and Account: https://quantum-computing.ibm.com

Qiskit: qiskit.org

Presentation and Demo Notebook:
https://github.com/qiskit-community/qiskit-presentations/tree/master/2020-09-15_DevNation



Questions



Thank you