Project#39: Tutorial to use K8S Cluster

Qiskit

Qiskit Advocate Mentorship Program: Spring 2021

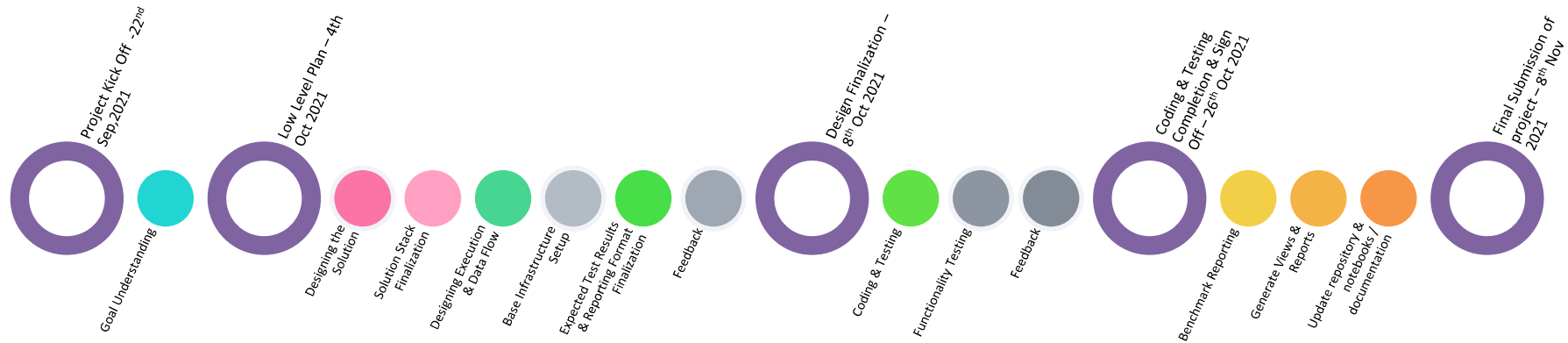Hiroshi Horii: Mentor - IBM Research, Japan

Anuj Mehrotra: Mentee - India

Background

✓ Simulation of quantum applications like quantum chemistry, materials science, quantum biology, generates quantum systems which are much larger than computational NISQ devices, this gap can be handled by parallelizing the quantum simulation.

✓ Qiskit Aer is a Noisy quantum circuit simulator backend and runs simulation jobs on a single-worker Python multiprocessing ThreadPool executor so that all parallelization is handled by low-level OpenMP and CUDA code.

✓ VQE (and the other variational algorithms) already support generating circuits together for parallel gradient computation. In order to customize job-level parallel execution of multiple circuits, a custom multiprocessing executor can be specified, which controls the splitting of circuits using the **executor** and **max_job_size** backend options of **Qiskit AerSimulator**.

✓ For large scale job parallelization on HPC clusters, Qiskit Aer executors support the distributed Clients from the **DASK** (parallel computing library for Python). DASK natively scales Python and suitable for applications which require a distributed, auto scaling compute environment that is completely independent of application.

✓ Using Kubernetes clusters, DASK worker environment can be either scaled up manually, or can be scaled as the need arises by creating auto scaling rules in Kubernetes configuration, which means DASK only need to manage scheduling across workers in Kubernetes Cluster , as work go up or down.

Goal

✓ Publish a tutorial to use DASK on Kubernetes(K8S) cluster as a (distributed) computation engine of Qiskit-Aer.

Implementation Plan

Qiskit

Timeline milestones:
- Project Kick Off – 22nd Sep.2021
- Goal Understanding
- Low Level Plan – 4th Oct 2021
- Designing the Solution
- Solution Stack Finalization
- Designing Execution & Data Flow
- Base Infrastructure Setup
- Expected Test Results & Reporting Format Finalization
- Feedback
- Design Finalization – 8th Oct 2021
- Coding & Testing
- Functionality Testing
- Feedback
- Coding & Testing Completion & Sign Off – 26th Oct 2021
- Benchmark Reporting
- Generate Views & Reports
- Update repository & notebooks / documentation
- Final Submission of project – 8th Nov 2021

Deliverables:

✓ Github Repository (with sample codes ,Tutorial, report)

✓ Project Report (with test results)

✓ Contribute to Qiskit Aer repository (raising issues /bug reporting, share resolutions)

Current Status:

✓ Project Goal Understanding Complete.

✓ Base Infrastructure Design & Setup Complete using DASK, Minikube, Dockers, Python 3.8 on Ubuntu 20.04 LTS OS platform.

✓ Proof of Concept Testing Initiated

✓ Contributed to Qiskit Aer repository : Issue #1364 : Serialization Error while submitting Quantum Objects to compute engine (DASK workers) of Qiskit Aer.

Architecture of Clustered Backend for Aer Simulator