Baseline test suite for an HPCC Systems cluster on Kubernetes

The proposal period for 2022 internships is now closed
The proposal period for 2023 internships will open in November 2022

Student work experience opportunities also exist for students who want to suggest their own project idea. Project suggestions must be relevant to HPCC Systems and of benefit to our open source community.

Find out about the HPCC Systems Summer Internship Program.

Project Description

This project requires at least some basic knowledge of HPCC Platform and test methodology. Current HPCC Platform has regression test suite https://github.com/hpcc-systems/HPCC-Platform/tree/master/testing/regress and performance test https://github.com/hpcc-systems/PerformanceTesting on bare-metal setup. This project is to adopt these tests to cloud environment mainly focus on benchmark type of measurement of Roxie and Thor jobs in various setup, such as cloud environment, storage types, Roxie and Thor targets size, Kubernetes Node size as well etc.

The code can be developed and tested in local Kubernetes and real measurement will be conducted primarily on Azure and optionally on AWS.

Here are some dimensions for the test:

- **Cloud:**
  Azure AKS
  AWS EKS
  Google GKE (optional)
- **Storage types:**
  Azure: diskfile, blob
  AWS: csi-efs, efs, FSx for Lustre (depends the implementation), s3
  GCP: nfs, cloud storage, file storage
- **Encryption** (storage/volume): on/off
- **Datasets:** various datasets will be provided for the project
- **Target:** thor (data process), Roxie (query)
- **Cluster size:** various thor and roxie size will be used in the benchmark.
- **Network speed**:
- **Caching:**

Additional considerations about the project

- A good grasp of HPCC Systems, and how it differs from a relational database (or from something that processes streamed data)
- extend performance test include other activities, and aspects of the system (e.g. lots of tiny subgraphs, lots of small graphs, workflow dependencies)
  monitor and analysis test results variation is from cloud noise or HPCCSystems Platform code changethe new test should be informative to be representative of the work that is done on the platform
- We want to reduce costs which for the cloud are a combination of time taken and machine type. It would be interesting to know how performance of different activities changes with different constraints e.g. number of cpus, memory, network bandwidth, and even better if it highlighted areas in the platform that would give a significant reduction in cost for little work.
- Focus on improving the performance suite, and gathering and analyzing the stats already generated by the platform (rather than using other benchmarking tools). Work closely with our performance test team to produce graphs and look at trends for the performance suite.
- Should not try to directly and exactly compare some RDBMS benchmark to Roxie, but as a follow-on to performance suite work - come up with a few types of queries (or select some from the existing performance suite) and then run those at various loads and AKS cluster sizes to show performance and also perhaps HPCC scaling.

A github project should be created to host all files and documentation.

Student will work closely with our build and test team.

If you are interested in this project, please contact Contact Details

Completion of this project involves:

- Measurement on Azure AKS and optionally AWS.
- A complete github project with Documentation

By the mid term review we would expect you to have:

- A github project with design and initial code implementation
- Basic setup and measurement on Azure.

Mentor	Xiaoming Wang Contact Details **Backup Mentor:** Godson Fortil Godson.Fortil@lexisnexisrisk.com, Turlapathi, Krishna Krishna.Turlapathi@lexisnexisrisk.com Contact Details

Skills needed	• General Cloud Environment knowledge such as Azure, AWS and GCP, Kubernetes and Docker • Unix Shell, Python and PowerShell • Ability to write test code. Knowledge of ECL is not a requirement since it should be possible to re-use existing code with minimal changes for this purpose. Links are provided below to our ECL training documentation and online courses should you wish to become familiar with the ECL language.
Deliverables	**Midterm** • A github project with design and initial code implementation • Basic setup and measurement on Azure. **End of project** Complete github project with documenatation. Finish measurements for Azure and AWS.
Other resources	• HPCC Systems website • HPCC Systems Cloud native Platform resources • Docker Hub: https://github.com/hpcc-systems/docker-hpcc • Learning ECL documentation and on-line training courses and any Kubernetes tutorial • https://hpccsystems.com/blog/persisting-data-cloud2 • https://hpccsystems.com/blog/persisting-data-cloud