OCRE Cloud Benchmarking Validation Test Suite

Ignacio Peluaga (CERN / Universidad de Sevilla) - João Fernandes (CERN)

OCRE Open Clouds for Research Environments

Context

OCRE (Open Clouds for Research Environments) aims to accelerate cloud adoption in Europe, bringing together cloud providers and the research community in the context of the EOSC (European Open Science Cloud).

Problem

In order to compare services performance and adequacy for multiple scientific domains (HEP, Life Sciences, Photon-Neutron Sciences, Astronomy), a framework to run a set of benchmarks across all cloud stacks is needed.

Solution

Assemble and package a cloud benchmarking suite with contributions from multiple research use cases and scientific domains.

Motivation

From 2016-2018, ten leading European research organisations participated in the Helix Nebula Science Cloud project (HNSciCloud), providing a hybrid cloud platform for the European research community, expressing the need to increase analysis capability and capacity for their scientific program.

A major result of HNSciCloud has been assembling a test-suite of more than 30 tests, covering all the award criteria described in the technical specifications. The OCRE project will extend the HNSciCloud test-suite, the validation tool entirely conceived and developed by the research community, to assess cloud vendors to serve the scientific use cases.

Current status

- » Provisioning with **Terraform**, supported by most of the providers
- » Abstraction layer by **Docker** and **Kubernetes**. Tests run on containers
- » Written in **Python** and **Shell**. Usage of simple **YAML** files for configuration.
- » Using **S3** to store results. Results in **JSON** format for simplicity and future processing.
- » Adding verification system: **NodeJS** service deployed at CERN's cloud that works as a proxy to receive and submit runs. Once completed, uploads harvested results to the S3 bucket. The verification step is essential to avoid faking results.
- » Cataloguing several tests to validate a variety of cloud services: compute, storage, HPC, GPUs, end to end network connectivity and advanced containerised cloud applications. Each test consists on a containerised application that returns a PASS/FAIL status based on pre-defined criteria. Tests are described below.

Project plan

2018 — Gathering of a set of independent tests covering cloud services across the stack from multiple scientific domains.

Selection of the overall framework and design of the software stack: how to achieve the needed abstraction layer to overcome heterogeneousness?

2019 — Implementation of the first version: provision the abstraction layer.

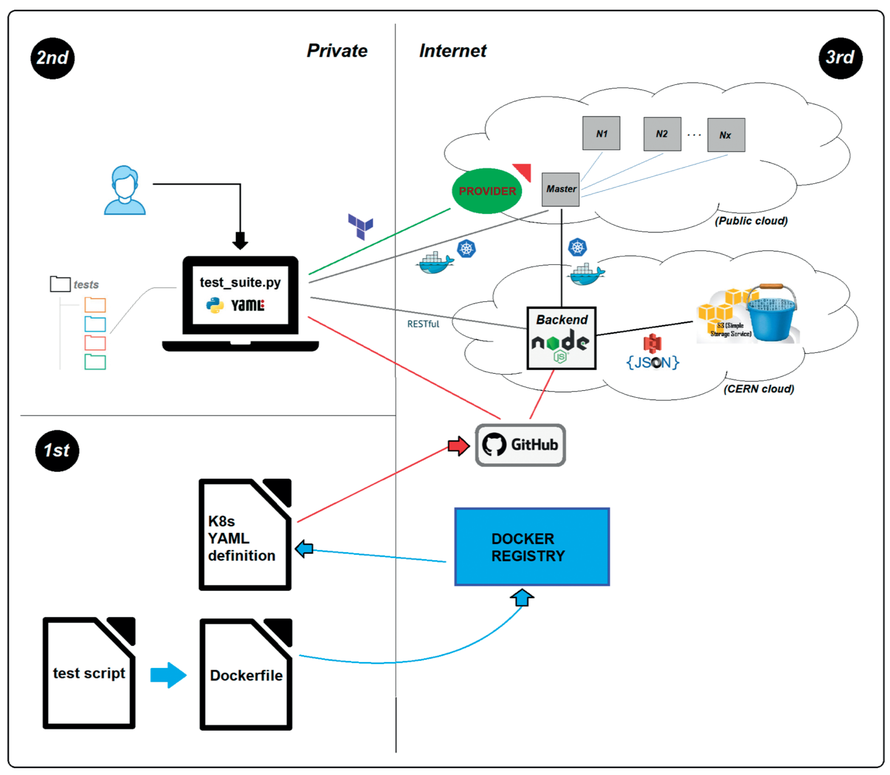
Tailoring and integration of the tests developed by the scientific institutes for HNSciCloud.

2020 — Development of newer and wider tests to be added to the validation tool.

2021 — Public Release: make it available to the scientific community including individual researchers (Long Tail of Science)

Area	Test description	Status
GPU	Deep Learning for **fast detector simulation** in High Energy Physics: distributed training of a Generative Adversarial Network. Tests **MPI-based** communication across nodes and Kubeflow deployment. For this specific test, GPU-flavoured machines are required, as it was defined on its use case itself.	✓
Storage	S3 protocol endpoint and CRUD methods: Testing of different **Simple Storage Service functionalities**. This test first tries to create an S3 bucket on the provider side. On success, it tries the following methods: PUT, simply GET, GET with prefix matching, GET chunk and GET chunks.	✓
Networking	Networking tests using the open source perfSONAR toolkit: includes traceroute, tracepath, iperf, nuttcp, ping and owamp. A perfSONAR **endpoint** is created **on the provider side** and from there, the networking test are launched against another perfSONAR endpoint the user choses. perfSONAR is a mature project developed by the University of Michigan among other institutions and offers endpoints deployed **all around the globe**.	✓
CPU	CPU benchmarking in a containerised way. This test runs **benchmarking** tools inside containers deployed on the cloud provider to benchmark the underlaying **physical machine**. The tests it contains are mainly simulations of scientific experiments which are very demanding in terms of **compute power**.	✓
Data Repatriation	Data exportation: take data out of the cloud onto Zendo. In the chase of making the scientific community embrace a **vouchers usage** of cloud computing, there is always a drawback: once my cloud credits are over, what can I do with the data I have on the cloud? **Zenodo** is a tool developed by CERN useful in this case as it keeps data for the **long term** and it is **free**. What this test does is to verify that data can be taken from the private cloud being tested onto Zenodo.	✓
HPCaas	Simulation of x-ray spectroscopies. **ESRF** provided the FDMNES suite for the suite. The aim of the FDMNES project is to supply to the community a user-friendly code to **simulate** x-ray spectroscopies, linked to the real absorption (XANES, XMCD) or resonant scattering (RXD in bulk or SRXRD for surfaces) of the synchrotron radiation. This test will validate that a provider meets the requirements to supply **HPC features** as FDMNES runs on a cluster of **huge-flavoured** VMs making use of **MPI**.	Under Development

For each test deployed, a Docker image is created containing the respective software: normally python or shell scripts or third-party tools. These images are used on Kubernetes resource definition YAML files to deploy pods on the cluster. Once the test inside the container running on the Kubernetes cluster is completed, results are harvested, resources released and the next test is deployed.



Usage in 4 steps

1. Clone repository and configure the tool: Simple setting of boolean variables on a YAML file to select tests to be run.
2. Run the tool from command line.
3. Eventually results are available locally.
4. Optional: use the verification system which after running the suite pushes the harvested results to the S3 bucket.

Get involved

The project developers seek for contributions and feedback from everyone. The code and all information about the project have been made available publicly.

Please refer to the following links in order to get more information. Feel free to test it and provide your feedback or let us know if you want to integrate a test.

HELIX NEBULA THESCIENCECLOUD
www.hnscicloud.eu

OCRE Open Clouds for Research Environments
www.ocre-project.eu

Repository: https://github.com/cern-it-efp/ocre-testsuite
Documentation: https://ocre-testsuite.readthedocs.io/en/latest/

For more information please contact:

Ignacio Peluaga: ignacio.peluaga.lozada@cern.ch

OCRE receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 824079.

OCREproject in/in/OCRE

CERN - 2019