Open Clouds for Research Environments

Test and validation suite for cloud services

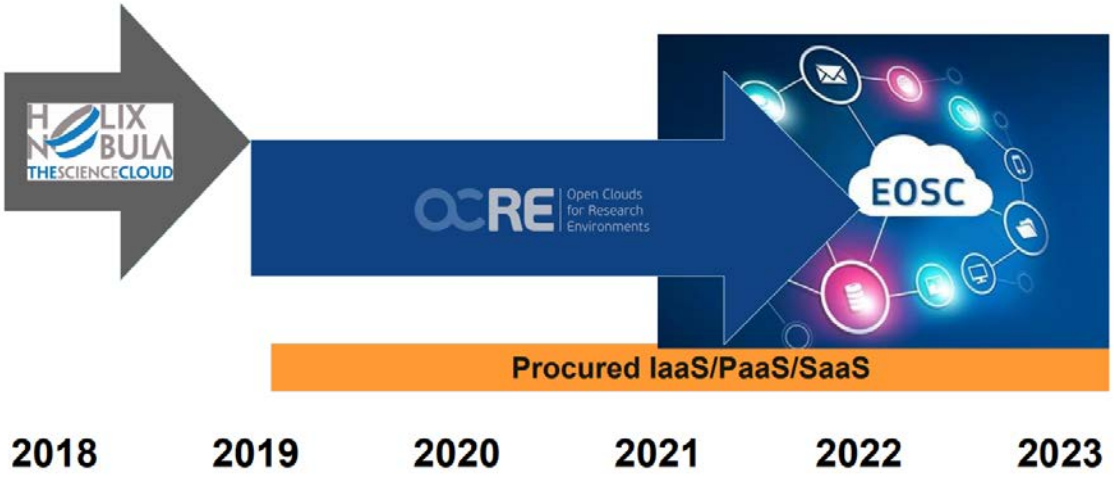
First European perfSONAR User Workshop – London UK, 5 June 2019

Ignacio Peluaga

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

The OCRE Project

- Lead by GEANT, the Open Clouds for Research Environments (OCRE) project aims to stimulate the usage uptake of commercial cloud services by the European research community
- Through a tender resulting in framework agreements with suppliers, which research institutions can use to buy commodity cloud
- By making available 9.5 million € in cloud adoption funds from the EC, for the research community to use
- Cloud providers bid as part of a public tender procedure
- Use HNSciCloud tests to assess providers' technical readiness level



GÉANT
IaaS Framework Provider

More info: ocre-project.eu

The HNSciCloud Project

- Helix Nebula – The Science Cloud
  - Provided an Hybrid cloud platform for the European research community
  - Collective effort of 10 procurer Research Organisations forming the Buyers Group
  - Expressing the need to increase the analysis capability and capacity offered to their users
  - Europe based providers of commercial cloud services bid to participate in the project. Those meeting the requirements and technically validated develop the system

HNSciCloud testing activity

- ***CERN***
  - CERN Batch service
  - GPU (Machine learning)
  - PerfSONAR
  - VM Provisioning and Personalisation
  - Dockerized deployment of E0S+CERNBOX+SWAN
  - Security assessments
- ***CNRS***
  - HTCondor dynamic expansion
  - LHC_VO MC
- ***EMBL***
  - Marine Metagenomics
- ***ESRF***
  - Functionality and user-friendliness of cloud interfaces
- ***INFN***
  - Bellell_SIM_PROD
  - Dynfarm_scale
  - DODAS
- ***KIT***
  - Batch system extension
  - CPU benchmarks
  - Fed auth with ECP SAML
- ***SURFsara***
  - BBMRi import and testing of images/recipes
  - LOFAR/BBMRI data transfer
  - LIGO/VIRGO
- ***IFAE***
  - MAGIC hybrid cloud
  - CTA_dirac data fetching
- ***DESY***
  - HPL and HPCG (HPC)
  - SIMEX (photon science)
  - Serial X-Ray (Crystallography)

Lessons learned

- ***Design and implement a tool that packages the tests from HNSciCloud, to be used to technically validate public cloud providers biding to the tender***
- ***Added value:***

Consistency

Faster testing

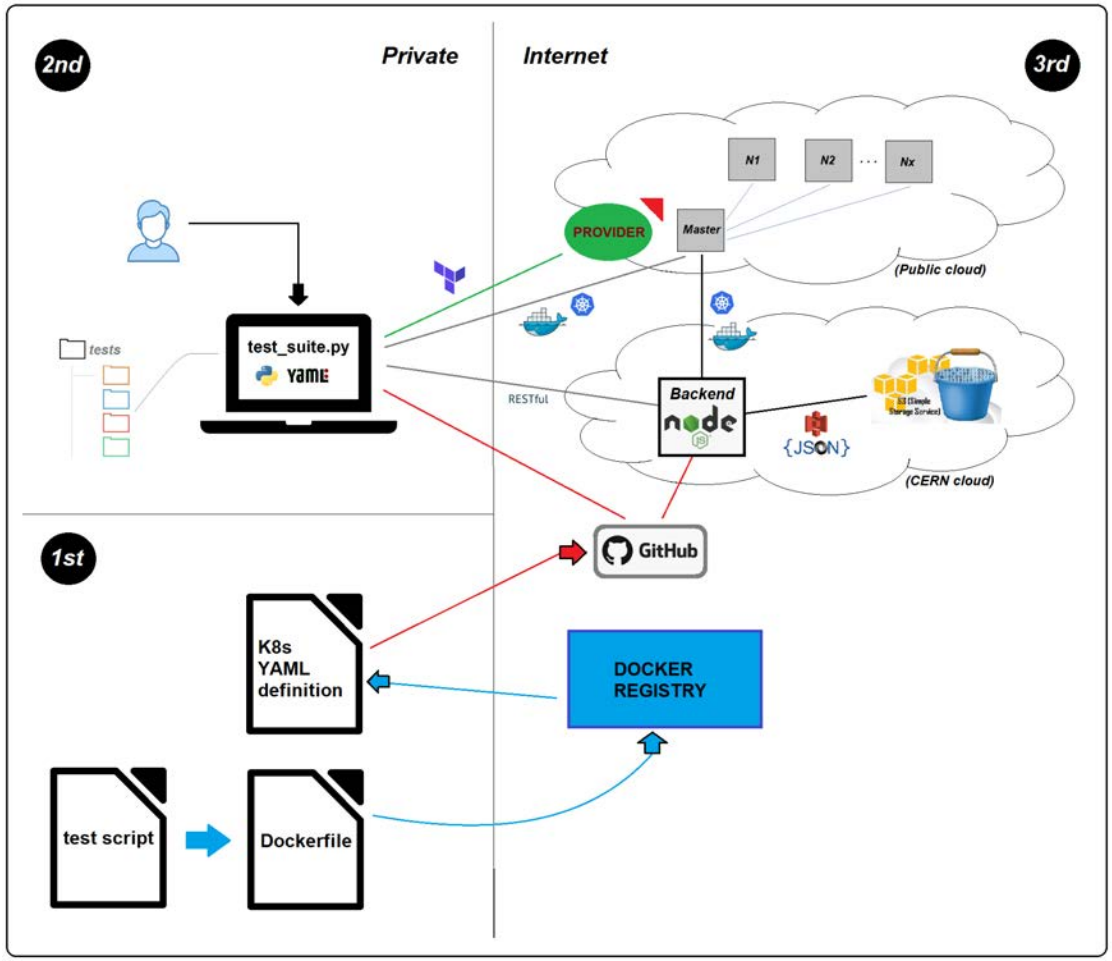
Cost effectiveness

Efficient testing

The Solution

OCRE Test Suite

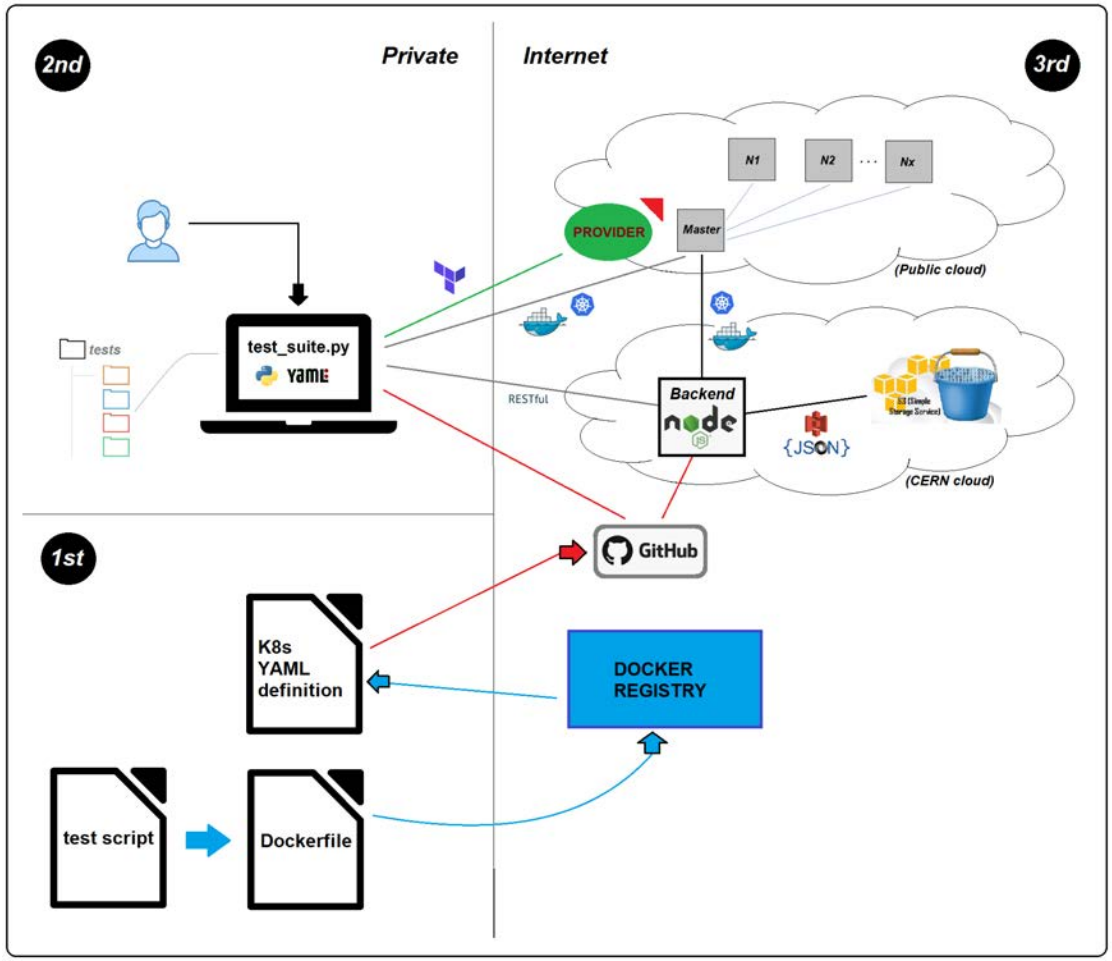
- ***To technically validate all cloud services selected in the OCRE Framework, based on tests assembled by HNSciCloud***
  - Written in **Python**
  - Provision of the stack with **Terraform**
  - Deployment of tests on **Docker** containers to **Kubernetes** cluster (abstraction layer)
  - Simple **YAML** configuration
  - Results: **JSON** on **S3** bucket on CERN cloud
  - **NodeJS** service for verification (under development and testing)



OCRE Test Suite

- **Process**

1) User clones the public repository
2) Configuration by filling *configs.yaml* and *testsCatalog.yaml*
3) The test-suite, according to *configs.yaml*, will provision raw VMs and then bootstrap a Kubernetes cluster on them
4) Once the cluster is ready, the test-suite will deploy the tests according to *testsCatalog.yaml*: these run on Docker containers



OCRE Open Clouds for Research Environments

Tests catalog

Tests Catalog

- **Existing test**
  - Compute: CPU benchmarking with containers (Domenico Giordano – CERN)
  - Storage: S3 endpoints (Oliver Keeble – CERN)
  - GPUs: Distributed training and optimisation of Deep Learning models (Sofia Vallecorsa – CERN)
  - Network: Performance measurements with perfSONAR (Shawn McKee – University of Michigan & Marian Babik – CERN)
  - Data Repatriation: From the commercial cloud provider to Zenodo (Ignacio Peluaga – CERN)
- **Tests under development**
  - HPC: FDMNES - Simulation of X-ray spectroscopies (Rainer Wilcke – ESRF)
  - DODAS: Emulate CMS jobs to verify that the node is able to run real workflows as in DODAS HTCondor environment (Daniele Spiga & Diego Ciangottini – INFN)
- **More to come**
  - Disk access stress, with non-streaming I/O patterns (EMBL)
  - HDF5_io (DESY)
  - Data isolation test (SURFsara)
  - SLURM (SURFsara)

OCRE Open Clouds for Research Environments

Network test

K8s pod deployment

testsCatalog.yaml

```
. . .
cpuBenchmarking:
  run: true
perfsonarTest:
  run: true
  endpoint: psb01-gva.cern.ch
hpcTest:
  run: true
  nodes: 3
. . .
```

```
---
apiVersion: v1
kind: Pod
metadata:
  name: ps-pod
spec:
  hostNetwork: true
  containers:
  - name: ps-cont
    image: perfsonar/testpoint:latest
    securityContext:
      privileged: true
    imagePullPolicy: Always
    env:
    - name: ENDPOINT
      value: psb01-gva.cern.ch
```

K8s pod deployment

- **Process**
  1) Test-Suite completes YAML file according to configuration
  2) Deploys the pod using kubectl and the YAML file
  3) Container on the pod uses perfSONAR testpoint image, runs pScheduler
  4) Once pod is ready: Test-Suite runs pScheduler tasks remotely using kubectl

pScheduler tasks run

```
#!/bin/bash

pscheduler ping $ENDPOINT || exit 1

pscheduler task --format=json throughput --dest=$ENDPOINT > throughput.json

pscheduler task --format=json rtt --dest=$ENDPOINT > rtt.json

pscheduler task --format=json trace --dest=$ENDPOINT > trace.json

pscheduler task --format=json latency --dest=$ENDPOINT > latency.json
```

Output of commands is sent to json files

Harvesting results and completion

1) Test-Suite harvests the resulting JSON files from the pod using kubectl

2) Once all results are harvested, the perfSONAR pod is killed

3) Verification System(optional): launch the test-suite –skipping provisioning phase- from a server running on the CERN cloud. Harvested results are then pushed to an S3 bucket

Testing the test suite

- **Ran on**
  - CERN Cloud (OpenStack), Exoscale (CloudStack), CloudFerro (OpenStack)
  - Detailed and processed results will come later

- **More to come**
  - T-Systems (OpenStack), Microsoft Azure, Google Cloud, AWS, …

Next steps

Network test: perfSONAR

- Use pScheduler's API instead of CLI
- Run latencybg task instead of latency
- Manage, process and homogenize pScheduler results

Test-suite

- New release
  - Multiple, parallel clusters
  - NAT support
  - Support of non ROOT access
  - Improved logs
  - Implement proper results verification: how to avoid fake results?
- Develop a dashboard to show results
- Add more tests
  - Contributions?

Test contribution

- **Step 1**
  - Contact the test-suite development team
- **Step 2**
  - Assessment of the work to be done and set up requirements
- **Step 3**
  - Info about documentation and contact of the person providing the test plus the license governing it

Get involved

PUBLIC REPOSITORY

https://github.com/cern-it-efp/ocre-testsuite

DOCUMENTATION

https://ocre-testsuite.readthedocs.io

OCRE WEBPAGE

https://ocre-project.eu

OCRE

Open Clouds for Research Environments

Thank you

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