

Summer student report

31 August 2018

marc.benedi.san.millan@cern.ch

# Construction of a Kubernetes cluster in the CERN Technical Network and automating its deployment with Ansible

Author: Marc Benedi
Supervisor: Andrea Calia
BE-OP-LHC
CERN, CH-1211 Geneva, Switzerland

Keywords: Cluster, Kubernetes, Docker, Ansible, Technical Network

This report describes how the cluster deployed in the Technical Network works and how to deploy it in other machines. Firstly, the base technologies such as Docker, Kubernetes, and Ansible are explained. Secondly, the methodology followed during the development of the project is described. Thirdly, how it works and the Ansible play-books structure, and fourthly, the manual in which how to install the cluster and do some usual operations is explained. Finally, the results showing the improvement in time, and the difficulties found during the project.

## Contents

**1 Introduction** **2**
1.1 Background . . . . . 2
1.2 Scope . . . . . 2

**2 Technologies** **3**
2.1 Docker . . . . . 3
2.2 Kubernetes . . . . . 3
2.3 Ansible . . . . . 3

**3 Results** **3**

# 1 Introduction

The CERN Summer Student Programme offers undergraduate students the opportunity to join research teams participating in experiments at CERN between 8 to 13 weeks in summer. The goal of the programme is give experience and knowledge to the students and put them in a multidisciplinary and multicultural environment.

This report summarises a Summer Student Programme project and be a guide to how to build, configure, and use the cluster.

## 1.1 Background

Recently, new technologies, such as Docker[1], present a new way of deploying applications which improves certain use cases. Along with cluster technologies, the CERN team BE-OP-LHC saw an opportunity to improve some of its services.

They wanted a cluster for operations to improve the following characteristics of its services:

- Scalability
- Response time
- Availability

In addition, to improve the usability of the cluster, a GUI/Dashboard was required. Using the Dashboard, the management actions are more comfortable to perform.

## 1.2 Scope

The scope of the project is to configure a high performance cluster for the deployment of different kind of services and applications that are vital for the operations of the LHC. At CERN, all the accelerators devices and machines are inside a so-called Technical Network (TN) for improved security. The cluster will also be part of the TN network due to the fact that the applications that will run on it will have to access accelerator specific devices and servers. This poses a challenge on the design of the configuration since the TN is not connected to the Internet, so the configuration cannot be fetched from outside CERN TN.

The LHC Coupling Analysis Service will be used as pilot project to assess the performance and availability of the cluster. The LHC Coupling Analysis Service calculates the coupling values of a LHC beam that is then used to make corrections to avoid beam instabilities. This service is used on a daily basis during LHC operations and will benefit from the high performance and parallelism of the cluster.

The final results of the project are the scripts and configurations for creating an operational cluster that will be available on the CERN TN.

# 2 Technologies

## 2.1 Docker

Docker[1] is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow developers packaging their applications along with all its dependencies, which assures that the application will work in any environment.
Developers package their applications into *Docker images* which are defined in the *Dockerfile*[2].
*Docker images* run in *Docker containers* which are the materialisation of the environment defined in the Dockerfile.

## 2.2 Kubernetes

Kubernetes[3], also known as K8s, is a platform for managing containerised workloads and services, the facilities both declarative configuration and automation. In other words, it is a layer of abstraction over container engines such as Docker.

## 2.3 Ansible

Ansible is an automation engine which can be executed remotely to configure machines. It allows defining states/conditions which have to be satisfied by the machine, and if they are not, it executes the required actions to carry out them.

For example, the following part specifies that Docker has to be installed in the machines. If it is not installed Ansible installs it automatically.

```
- name: Installing docker
  yum: name=docker state=present
```

Ansible is used in this project to configure the machines in the cluster and deploy K8s and Docker easily.

# 3 Results

In this section, the experiments and their results are explained.
The following experiments were done deploying the LHC Coupling Analysis Service with different configurations of: number of pods, memory limit, and CPU limit.

As shown in figure 1, with 2 CPUs and 2GB of memory, the time per analysis is around 35 seconds. The experiment performed 25 analysis in two different environment: the first one with 20 pods and the second one with 40 pods. The expected result was that in the first environment some analysis would be queued whereas in the second environment all the analysis would be done in parallel because the number of pods was bigger than the number

of analysis. The result was that in the first environment 6 analysis were queued whereas in the second environment 3. The conclusion is that Kubernetes does not balance the load between the available pods because it adds more analysis to pods which are already busy when there are free pods.

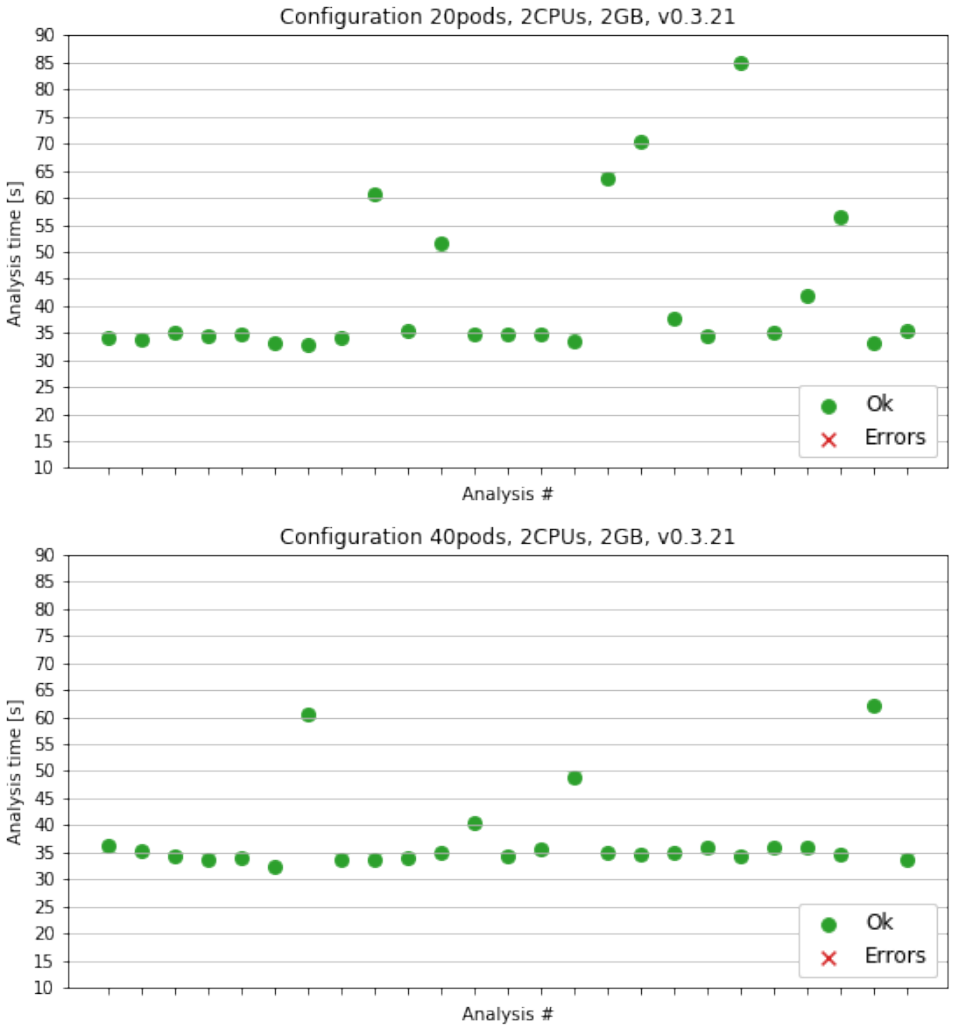



Figure 1: Time per analysis between the request and the response from the client side

The last experiment was for finding the optimal configuration of the pods. Eight different combinations where tried with 2/4GB of memory and 2/3/4/5 CPUs. The result was that a configuration with 3 CPUs and 2GB of memory offers an acceptable throughput. Despite that 4 CPUs with 4GB offers the best throughput, the increment in resources for each pod does not justify it's increment in performance.

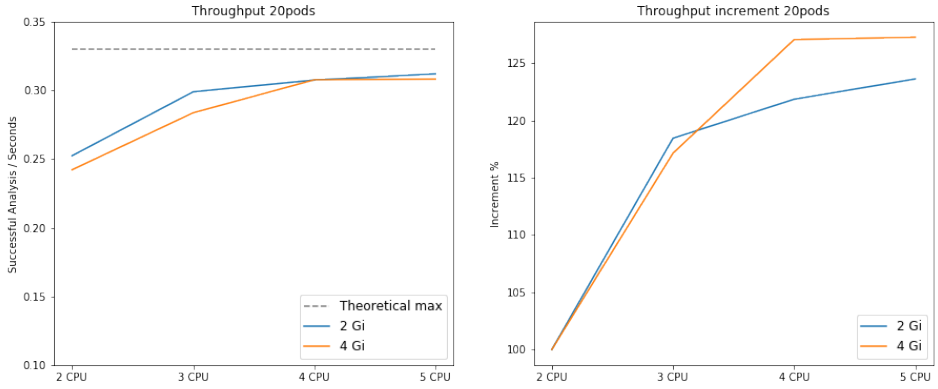



Figure 2: LHC Coupling Analysis Service throughput with 20 pods and different CPU and memory configurations

# References

[1] Docker. https://www.docker.com/.

[2] Dockerfile. https://docs.docker.com/engine/reference/builder/.

[3] Kubernetes. https://kubernetes.io.