

# Spark on Kubernetes

## An overview



+



Saurav Agarwal

# What is Spark on Kubernetes?

- Kubernetes (also known as Kube or k8s) is an open-source container orchestration system initially developed at Google, open-sourced in 2014 and maintained by the Cloud Native Computing Foundation. Kubernetes is used to automate deployment, scaling and management of containerized apps - most commonly Docker containers.
- Spark on Kubernetes added the advantage of using the above features of Kubernetes and replacing Yarn, Mesos etc as a de facto resource. management and scheduling mechanism.



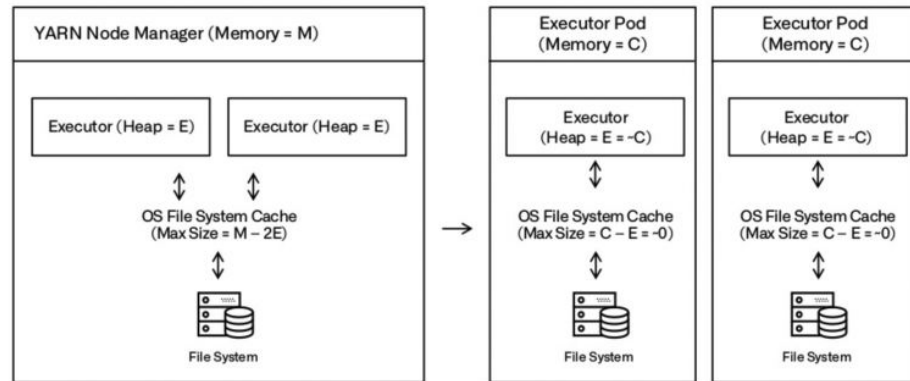
# Motivations behind Spark on Kubernetes

- Containerized Spark to provide shared resources across all the Data engineering and Machine Learning jobs
- Out of the box support for multiple Spark versions, Python versions, and version-controlled containers on the shared K8s clusters
- A single, unified infrastructure for both majority of batch workloads and microservices
- Fine-grained access controls on shared clusters



# Challenges in Spark on Yarn solved by Kubernetes

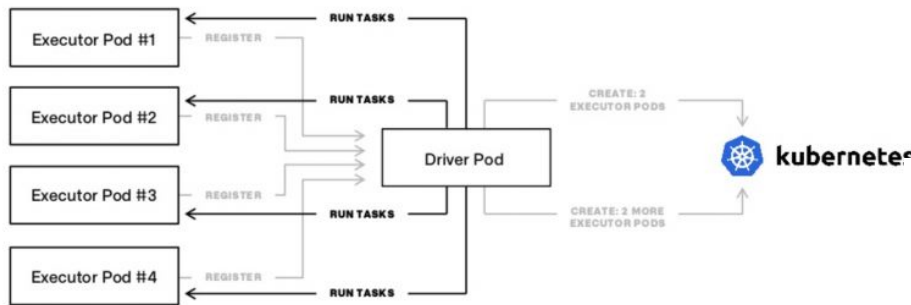
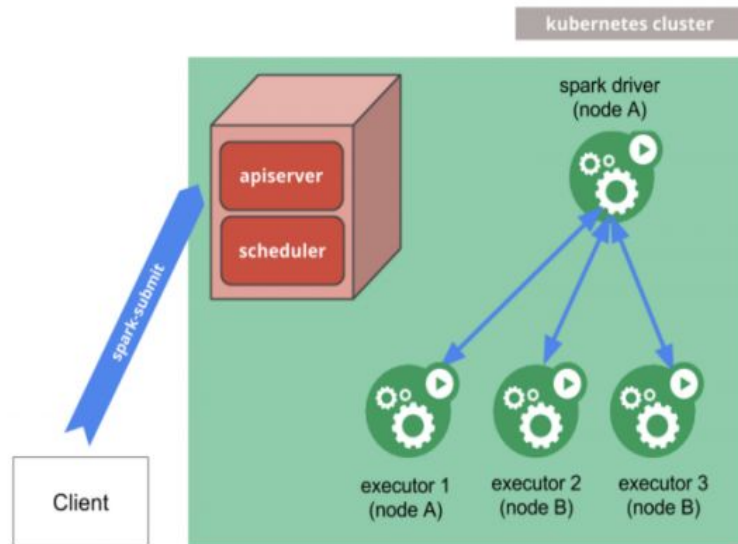
- Based on the diagram we can see below the advantages of Kubernetes over Yarn:
  - Providing consistency from run to run
  - Isolating performance between different tenants
  - OS file cache penalty due to different executors running in a common node manager



# Spark on Kubernetes Architecture

The architecture flow can be broken down to below steps:

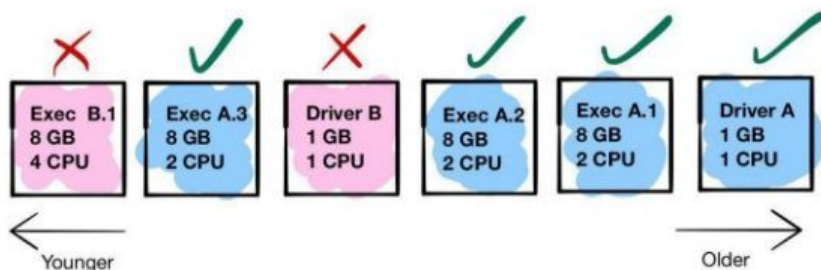
- Client submits Spark submit with arguments
- Arguments get converted to pod-spec for the driver
- Scheduler backend on Kubernetes requests executors pods in batches



# Spark on Kubernetes Architecture

However, the problem in this approach is the pods are launched one by one hence there can be a situation where an application gets stuck due to unavailability of executor pods. To resolve this various techniques are used as bellow:

- Gang scheduling – Run pods app by app instead of executor/driver in a FIFO manner
- Yunikorn- Maintains job ordering, resource capacity management and fairness



# Spark 3 Support for Kubernetes

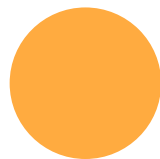
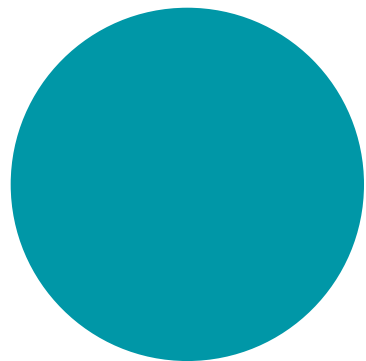
Kubernetes support began in v 2.3.x, However new features for Spark 3 are as below:

- Pod Templates
- GPU-aware scheduling (Isolation at GPU Pod Level)
- Improve behaviour with dynamic allocation
- The Kerberos authentication protocol is now supported in Kubernetes resource manager.

To see Step by Step commands to run Spark on Kubernetes, See Apache documentation [here](#).

- <https://spark.apache.org/docs/latest/running-on-kubernetes.html>





Thank you!



Saurav Agarwal