Running Apache Beam on Kubernetes

Sascha Kerbler
Sascha Kerbler
Cloud Space Architect
What are containers?

Containers are a **method of packaging** an application executable and its dependencies (runtime, system tools, system libraries, configuration), and **running the package as a set of resource-isolated processes**.

Buzzwords associated with containers:

- Lightweight
- Portable/Standard
- Efficient
- Secure
Kubernetes in simple terms

Think of Kubernetes as the OS for your compute fleet

It provides features such as:

- Scheduling workload
- Finding the right host to fit your workload
- Monitoring health of the workload
- Scaling it up and down as needed
- Moving it around as needed
Why Containers & Kubernetes?

- Containers and Kubernetes artifacts are versioned
- Containers can be centrally deployed
- Infrastructure portability
- Consistency between teams
- Simplified debugging and test
Portability in Beam

Languages
- Java
- Python
- GO

Runners
- Dataflow
- Nemo
- Flink
- Spark
- Samza
- Twister2
Architecture

Java
Go
Python

Dataflow
Flink
Direct
Spark

Kubernetes

Infrastructure
Options on Kubernetes

Kubernetes native

Data Analytics Native
Demo

- Goal: Run a beam wordcount on a kubernetes cluster leveraging the Flink Operator

Requirements:

- JAR File
- Docker Container
- Kubernetes Deployment
- Flink Cluster
Conclusion

*Be careful and think twice...*
Questions?

kerbler@google.com