MESOSPHERE

A Short History of **Container Orchestration**

A brief primer of Mesos and the container orchestrators, like Kubernetes, that run on top of it

Introduction

Apache Mesos and Kubernetes are often compared directly to each other as container orchestrators, but they were built with very different goals in mind.

The main goal of Mesos is to simplify building and operating fault-tolerant and elastic distributed systems (like Kubernetes). In contrast, the main goal of Kubernetes is to manage and orchestrate containers.

So what is Mesos? The original Mesos paper (it was called Nexus at the time) argues that there is no one-size-fits-all cluster computing framework. The consequence of this is that organizations will run multiple frameworks to support different workloads. Building and operating distributed systems is hard, so the design of Mesos focuses on the low level heavy lifting that workloads need, like resource sharing and isolation.

Mesos leverages this abstraction to take care of the hard parts of operating a distributed system, while giving systems engineers the flexibility to design their own orchestrators to run on top of the Mesos framework.

Why does this matter for container orchestration? Ultimately, most developers want the ease and feature set of a PaaS to deploy their applications, but both developers and PaaSes tend to be opinionated about their technologies and workloads, so one size fits all PaaSes rarely succeed broadly.

The Mesos team recognized this early and designed Mesos so users could build opinionated workflows on top of it without being opinionated itself.

We wanted to put together a short history of mesos and container orchestrators by focusing on different container orchestrators and the companies who use them.

Container Orchestration on Mesos: A Timeline

Companies in the Mesos community have taken advantage of Mesos' unique architecture. Over the years, multiple container orchestrators and PaaSes have emerged on top of Mesos.

Released
June 2013

Marathon

(Mesosphere)

Project page ›

Marathon is the first open source project released by Mesosphere. It is designed as an un-opinionated building block for SRE teams to build a PaaS on top.

Mesos handles rolling deployments, application groups, health checks, and scaling, but doesn't prescribe a workflow. It has a JSON+REST API so that it's easy to integrate with other systems and tooling. Partially inspired by internal toolsat Airbnb, JSON was the best fit for machine-machine communication.

Aurora

(Twitter)

Project page ›

Aurora on Mesos has been running in production at a very large scale across a shared pool of machines at Twitter. When these machines experience failure, Aurora updates those machines until they are healthy.

There are a lot of “Twitter-isms” which users outside of Twitter may find strange. Thrift is their internal standard protocol. Google’s is Protobuf / GRPC which is used by Kubernetes. Not a common protocol for SREs at most companies.

Thermos DSL inspired by Borg (engineers came from Google). There’s a learning curve to any DSL and more common languages like JSON, YML etc. have better tooling such as syntax validation, etc.

Released
October 2013

Singularity
(Hubspot)

Project page ›

Singularity on Mesos, and its components, provide an entire Platform as a Service (PaaS) to end-users. Singularity has many features built to reduce developer friction and ensure proper operation and reliable deployment of tasks.

Singularity Integrates with their custom load balancing system (Baragon), which is a system for automating load-balancer configuration updates.

Released
June 2014

Kubernetes
(Google)

Project page ›

Kubernetes launched with a great API and CLI that developers love. At Mesosphere, we saw its potential early and invested in bringing it to Mesos.

Kubernetes was built by Google and it runs on top of a cloud. Mesos serves as a cloud provider for Kubernetes, enabling operators to run it on any infrastructure, cloud, datacenter or a hybrid combination, with a similar degree of automation to what public clouds offer.

Kubernetes is currently the fastest growing workload on Mesosphere DC/OS.

Released
November 2015

PaaSTA

(Yelp)

Project page ›

PaaSTA is an opinionated PaaS that integrates with existing Yelp infrastructure. PaaSTA is a bit different from the other examples in this ebook in that it's not a new Mesos framework, but rather a PaaS interface built on two Mesos frameworks: Marathon and Jenkins.

From the PaaSTA Principles: "PaaSTA was built because we believed we could provide business value by building our own PaaS to seamlessly meld with Yelp's existing infrastructure. At the same time PaaSTA strives to avoid the downsides of building your own PaaS by heavily reusing existing open source components."

Released
April 2018

Titus

(Netflix)

Titus was built as a container management platform to be used in production to power Netflix's streaming and content systems. Titus' tight integration with VM-centric-Netflix cloud tools to enable a smooth transition between native Netflix OSS projects and containers.

Titus is integrated with AWS because it's broadly used, enabling ENI and security group management support, autoscaling, and load balancing. Another reason for their integration with AWS is the ability to quickly add features when business needs arise.

According to Netflix, "when it comes to container execution, we have a unique approach to container composition, Amazon VPC networking support, isolated support for log management, a unique approach to vacating decommissioned nodes, and an advanced operational health check subsystem."

Conclusion

Container orchestrators are now a main part of the software delivery pipeline. They need to integrate with other (often internal/proprietary) system, and support the unique workflows many companies have.

Mesosphere provides support for Marathon and Kubernetes, and our community members have built their own orchestrators to fit their needs.

If you're developing your own container orchestrator or PaaS on top of Mesos or DC/OS, let us know!

Learn More ›

Mesosphere is leading enterprise transformation toward distributed computing and hybrid cloud. We combine the rich capability you get from public cloud providers with the freedom and control of choosing your own infrastructure. Mesosphere is the premier platform for building, deploying, and elastically scaling modern applications and big data services.

mesosphere.com