There are countless articles, discussions, and legions of experts weighing in on Docker, Kubernetes, and Mesos. Unfortunately, many of these articles suffer from a fundamental flaw: the myth that these three open source projects are in a fight-to-the-death for container supremacy!

While all of these three technologies make it possible to use containers to deploy, manage, and scale applications, they each solve for different things and are rooted in very different contexts. Read on to find out more.

The Myth

There are countless articles, discussions, and legions of experts weighing in on Docker, Kubernetes, and Mesos. Unfortunately, many of these articles suffer from a fundamental flaw: the myth that these three open source projects are in a fight-to-the-death for container supremacy!

While all of these three technologies make it possible to use containers to deploy, manage, and scale applications, they each solve for different things and are rooted in very different contexts. Read on to find out more.

Let's Start With Docker

Docker is a container-orchestration platform that enables your applications to work seamlessly in any environment by packaging your app and all its components into one container.

Using containers means:
- Consistency across environments
- More streamlined workflows
- Easier to maintain and distribute services as needed
- Easier adoption of new technologies

Organizations started using so many containers that they needed a way to coordinate them across multiple machines.

Enter Container Orchestration & Kubernetes

Container orchestration is the process of automating the coordination of large numbers of containers by provisioning hosts, scheduling and restarting containers, linking them together, and more. Usually if you are running multiple containers you need a way to orchestrate them.

Kubernetes helps all of your containers move together by:
- Automation container deployment
- Scheduling and scaling containerized apps
- Running on multiple environments
- Being highly available

But... if you want to run other workloads like analytics and stateful data services? I mean, containers aren't the only thing, right?

All the Workloads All the Time With Apache Mesos

Apache Mesos was started as a UC Berkeley project to create a next-gen cluster manager. Mesos' modular architecture enables companies to leverage a single unified platform to run any combination of workloads together on any combination of clusters.

What workloads does Mesos support?
- Containers and microservices
- Data services
- Analytics
- Legacy applications

Container orchestration, like Kubernetes, are the top workloads that run on Mesos. But containers aren't the only workloads you want to run on Mesos. Because you can use Mesos to run your datacenter and cloud as a single computing resource.

Why Does This Matter?

Let’s say you want to scale out a single container. Kubernetes, and Mesos (all open-source projects) make this possible. But Mesos gives you the flexibility to scale other workloads, like stateful data services or analytics, on the same compute platform.

Do You Know What to Use Where?

This infographic shows the different use cases of Docker, Kubernetes, and Mesos (all open-source projects). This visual tool will help you decide what tool to use where. This is a great tool for all teams who work with Kubernetes, Docker, or Mesos.