

[How Kubernetes and Mesos are Solving the future of abstraction - Part 1 of 3](#)

Since the dawn of computing, we have seen some incredible advancements. There have been numerous inflection points along the way. The most significant shift in many years was led by a little startup led by Diane Greene. You may know them as VMware.

I can't tell you when the data center operating system tagline was first published, but you can see now that VMware is fighting towards trying to be the everything system which will hold the reins as a dominant force in enterprise computing.

Then the next shift happened.

Enter Containers

Containers seemed to solve an issue that the virtualization platforms and virtual machine architecture couldn't solve. Speed of delivery was now shortened to seconds instead of minutes/hours. Along with the agility and speed came the realization of a better abstraction which enabled a true every-infrastructure capability. It didn't come without complexity.

The container construct has solved some very specific problems that were challenges in a traditional VM architecture. We have effectively come close to solving the fundamentals needed to see the more pragmatic technology adopters start to lean into containerization.

The problem comes with scheduling, visualization, security, and networking. These issues are handled in a number of ways. Two products that are leading the charge are Kubernetes and Mesos.

The Container Scheduler Wars Begin

Wars are decided, not by who outright wins, but by who loses more slowly. It can be a glacial speed event that becomes a massively prolonged push to find the leader and followers in the product podium battle.

Kubernetes is inevitably being groomed as the gold medalist in this war. Mesos solves similar challenges, in a slightly different way. In my opinion, it's like arguing over whether a 4-pronged fork is better than a 5-pronged fork. They both solve the same fundamental issue in slightly different ways.

What is it that Kubernetes and Mesos solve? The final realization of a true data center operating system. A common model by which you can abstract infrastructure which spans on-premises and public infrastructure. Is it too good to be true? It's still early to tell what challenges are being introduced that will be most easily surpassed to get to the more simple to consume data center OS.

The Players in the DCOS Game

There are many projects, products, and companies who are leading the charge towards the hope of becoming the de facto standard in the ultimate data center operating system. The two mentioned in the title are clearly leading the news and the number of active downloads.

There is no doubt that Kubernetes holds the title of leader at the moment. It's the core of many other

products that are aiming to become the platforms of today and tomorrow. Don't get confused (as I sometimes do) when we talk about Mesos and DC/OS. Having had a great [chat recently on the DC/OS](#), I am careful to differentiate between DC/OS which uses Mesos, which is different from Mesos as a standalone core, and also not the same as Mesosphere, who happens to be the largest contributor to DC/OS.

The notable absence of Docker Swarm in this article was also on purpose. Docker Swarm is much more of a purpose-built product that is aligned with Docker the product, but more importantly, Docker the company. Many folks in the industry have chosen Docker as the container ecosystem to bet on. Kubernetes and Mesos are positioned to leave flexibility for all, which is there real reason that I'm aligning my own work to these two for the time being.

The OpenShift product suite is very interesting, and also uses Kubernetes under the covers. VMware has taken a different direction with the creation of their Photon products. Photon is also being open sourced for the most part, with some proprietary bits being held close to the chest when it comes to the integration with the vSphere products.

Thoughts on What's Next

Kubernetes will be the larger player. Despite being second to market behind Mesos, it was created as a result of learnings from the Borg and Omega infrastructure running inside the Google data centers for 10 years before it was spawned and thrust out into the open. I believe that Mesos is powerful, but will lag behind in the same way that LXC and Rocket will trail Docker for now.

If you were to spend your effort investigating a platform, Kubernetes is my pick for your first, and then we can spin up Mesos next to see how they compare architecturally, As you can see from the title of the post, that is exactly where we are going with parts two and three!