StormForge

Closing the Day 2 Kubernetes Operations Gap

How AI and automation can optimize Kubernetes for cost, performance, and productivity

The Kubernetes Day 2 Gap is Real

The rapid adoption of cloud native architectures is key for organizations that want to remain competitive in today's world. The promise of speed, agility, scalability, and a better user experience are persuading more and more organizations to adopt Kubernetes and containers to underpin cloud native architectures.

Unfortunately, when the reality of Kubernetes sets in, organizations can find themselves challenged to realize the potential benefits. Kubernetes is complex and skills are in short supply, exacerbated by the COVID pandemic and increasing demand for digital transformation. And, even when you get the right people and skills in place, the job is not yet done – getting started with Kubernetes is just the beginning. Scaling up from Day 1 (deployment) to Day 2 (management, monitoring, and maintenance) can trip up even the best team and impact the business value you derive from your Kubernetes investments.

Why? Because while moving out of Day 1 operations can be exciting, Day 2 operations will determine the ultimate success or failure of your initiative.

Typically, when you adopt Kubernetes, you begin by experimenting in pre-production and then scale up in production – Day 2. At that point, it's showtime! You're ready to install the environment and scale with real users. But it doesn't end there, because Day 2 brings a whole new set of challenges: How will you monitor the environment? How will you onboard new internal customers? How will you ensure security? How will you control costs while still ensuring application performance?

What can go wrong in Day 2:

- Costs to run applications exceed expectations.
- The applications that were supposed to be deployed don't come near the number anticipated.
- You build it, but they don't come.
- SLAs can't be met.

EXAMPLES:

- As Airbnb created new containerized services and also migrated existing services to Kubernetes, they encountered an increasing number of complaints about latency.
- As online classified ad provider Adevinta started routing production traffic to their Kubernetes environment, they found request latencies up to 10x higher than expected.
- Monitoring provider Blue Matador saw issues in their production Kubernetes cluster with nodes running out of memory, causing their team to work after hours troubleshooting and debugging the problem.
- Retailer Target experienced a cascading failure of distributed systems when they upgraded their network subsystem, causing connectivity to be disrupted for several hours.
- Online fashion store Zalando experienced a high number of errors in production that affected customer experience, caused by an outage of the Kubernetes cluster DNS infrastructure.

Read about these examples of day 2 challenges and many more at **k8s.af**.

Ramping Up to Day 2 Operations

As the realities of Day 2 operations set in, many organizations find they are forced to make choices that dramatically affect applications, users, and your business:

- Over-provision cloud resources by 50% or more, driving costs through the roof.
- Risk business-impacting application performance and availability issues.
- Slow down time-to-market to spend time manually tuning your complex Kubernetes environment.

For modern organizations, learning as fast as you can about how your business applications operate and behave is vital to accurately forecasting risks that affect business. You can acquire knowledge in one of two ways: **Learning through experimentation or observation**. In Kubernetes environments, you experiment in non-production by simulating a wide range of scenarios to determine how you can optimize, and you observe in production to make recommendations based upon what you see.

While in pre-production you decide what scenarios you're planning for, in production there are things you can't plan for because you don't know exactly what's going to happen. Too often, discovering what went wrong is an exercise in itself, involving cross-functional teams and taking significant time until you realize that your best people are spending more time troubleshooting than they are developing and innovating to move the business forward.

This is where automation and AI can make things easier by helping you manage Day 2 operations.

From People to Process, Automation and AI are Key

Although you need to acquire knowledge and information fast, you don't know how applications are going to operate in production. This results in teams typically beginning by deploying the first few apps and learning from there. But each application is different, and you need to iterate as many times as possible to optimize – doing this manually at scale is not viable or operationally efficient.

By leveraging machine learning, organizations can improve operational efficiency, performance, and cost. While observability tools collect massive amounts of data, we need to be able to move from *observability* to *actionability*. That means you need to be able to interpret the data and determine what it means and how it informs what your team will do as a result. AI enables you to automate with machine learning that can look at your environment, the variables, and data – at scale – and then make recommendations or provide insights that allow you to make intelligent business decisions.

Automation isn't about eliminating jobs or roles, but rather about relieving resources from time consuming, task-oriented activities and enabling them to work on business-driven initiatives that are more strategic (and, ultimately, rewarding).

94% **of organizations adopting Kubernetes say it's a source of pain for their organizations.**

SOURCE: D2IQ, KUBERNETES IN THE ENTERPRISE: UNCOVERING CHALLENGES & OPPORTUNITIES, 2020

A Better Approach for Kubernetes Day 2 Operations

You can leverage ML and automation in both pre-production and production to ensure that applications run well and meet, or exceed, performance SLOs and SLAs with minimal effort and cost. You can do this with an approach that allows you to experiment in pre-production to understand app behavior before deployment *and* make adjustments in production based upon data that provides insights for action.

Pre-Production

Pre-production, where learning is experimental, is ideal for performing deep analysis and scenario planning. The trade-off to pre-production experimentation is that it requires teams to spend time creating load tests with various scenarios. The benefit from this up-front investment in load test creation is that Automation and ML can do the heavy lifting when applied to pre-production environments, giving you the ability to benefit from insights, build confidence in how your application will behave in different scenarios, and eliminate the time and resources that would be dedicated to pre-production optimization.

Production

Production, where observation is used to isolate variables that can impact optimization, is an opportunity to leverage the large amount of observability data that is already being generated by existing tools to show how applications are running. With ML, and the intelligence it introduces to your environment, that data can be efficiently analyzed over time to make informed decisions about how to change the configuration of apps in real time. By improving how apps run, and optimizing trade-offs, ML helps to improve performance and cost while ensuring that you are not overpaying for the value you are getting from the application. Automation can also make app optimization in production easier and faster by finding insights with little to no effort and taking action based upon those insights. You have the option to fully automate the process or include approval steps based upon the criticality of the application and the level of confidence in the machine learning.

When we talk about optimization – whether in pre-production or production – it isn't about one approach being better than another. Where in the process you optimize depends upon your Kubernetes journey. That's why StormForge addresses Kubernetes application testing and optimization both in the development cycle and during production. StormForge informs, optimizes and operates throughout the entire cloud-native development cycle for developers and operations managers who require an intelligent and comprehensive platform that maximizes their returns on Kubernetes investments to realize the promise of Kubernetes and cloud native.

About StormForge

StormForge brings together world-class data scientists and software engineers to enable businesses to drive breakthrough IT and operations efficiency. The StormForge Platform is set apart by its unwavering focus on building AI-powered software products that are designed to help people, not replace them. The StormForge Platform uses enterprise grade performance testing coupled with machine learning to drive major application performance gains and cost reductions in complex environments. In February 2020, StormForge announced funding from Insight Partners to accelerate the growth of its Platform. StormForge was founded in 2015 and is based in Boston, Washington, DC, and Cologne, Germany. Learn more at **www.stormforge.io**.

Discover the Advantages of StormForge

Let our experts assess how StormForge can deliver the foundation for your Kubernetes success.

Schedule your demo today.

SCHEDULE DEMO →

StormForge

+1 (857) 233-9831 | info@stormforge.io | stormforge.io