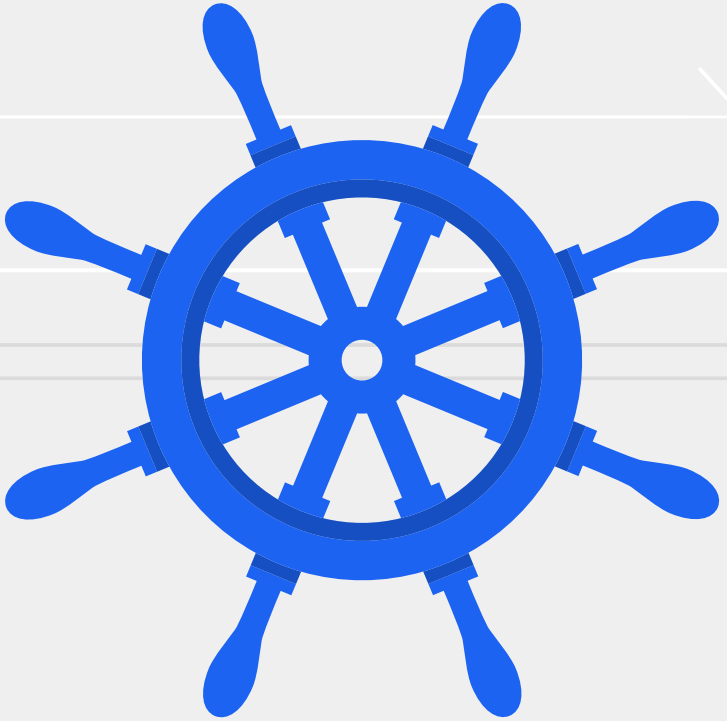
Kubernetes Governance: Take Control of Your Multi-Cluster Operations

D2IQ

Introduction

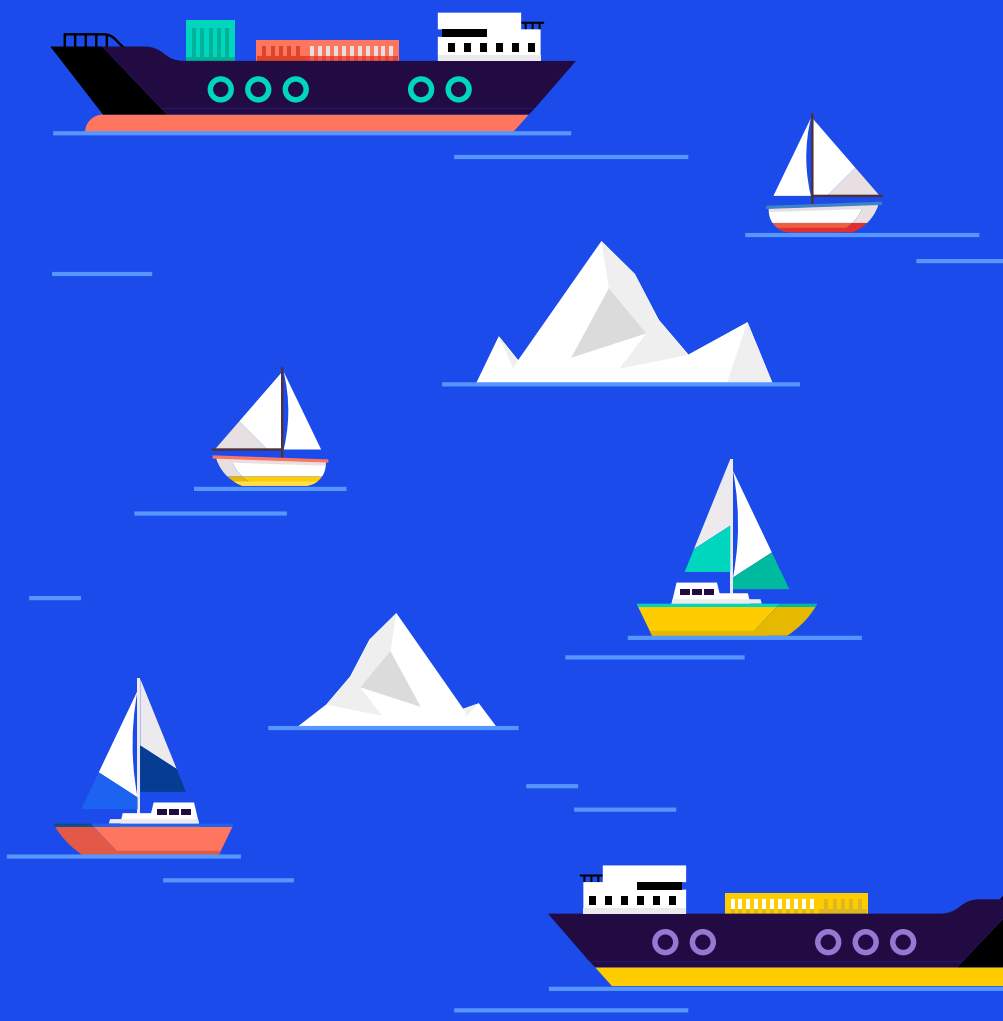
The adoption of Kubernetes is continuously growing in enterprises. Kubernetes gives organizations the freedom to select the applications they need and run them on the infrastructure of their choosing. What enterprises haven't fully realized is what it means to govern those activities or use data in the cloud in a compliant and reliable way. How do you balance the needs of your team as new tools and approaches emerge? And how do you structure your team to be agile, yet responsible?

In this ebook, we'll go over the challenges that organizations face today as they adopt and deploy new technologies. We'll also provide a blueprint on how to evolve your existing governance models in a way that empowers everyone.



CHAPTER 1

Technical Freedom: How Much is Too Much?



Businesses of all shapes and sizes are taking advantage of the hundreds of technology services that the cloud native landscape offers. Every few months, new open source projects, databases, and developer tools are coming out, and it's empowering innovation like never before.

As various teams are discovering creative ways to leverage Kubernetes, they're building an expanse of new clusters to support their efforts.

Unfortunately, this is where many of the challenges begin. While managing one Kubernetes cluster is not trivial, trying to manage multiple Kubernetes clusters on multiple clouds becomes exponentially more difficult. And, if you're operating in an enterprise environment with a growing number of clusters that are being managed independently with very little uniformity, the complexities can be a huge barrier to success.

Lack of Visibility and Management

As various parts of the organization require new Kubernetes clusters, it becomes increasingly difficult to know where they exist, how they are performing, and to govern the usage and versions of cloud native software to support application efforts. For example, teams will be building one stack on cloud provider “A,” a different stack on cloud provider “B,” and a third stack at the edge or in a data center. As the number of clusters and workloads grows and multiplies across different environments, operators struggle to create standardization around how and where new clusters are configured and used.

“Developers are introducing a lot of these new stacks often in a bottoms up, organic way,” explains Tobi Knaup, Co-CEO of D2iQ. “So very quickly an enterprise finds themselves with 10 or 15 different ways to provision infrastructure and the teams that are in charge of governance aren’t even aware of these things.”

Operators need consistent ways of administering, standardized user interfaces, as well as managing and obtaining insights about their infrastructure. When there’s a lack of centralized governance and visibility over how and where these resources are provisioned, it negatively impacts the bottom line.

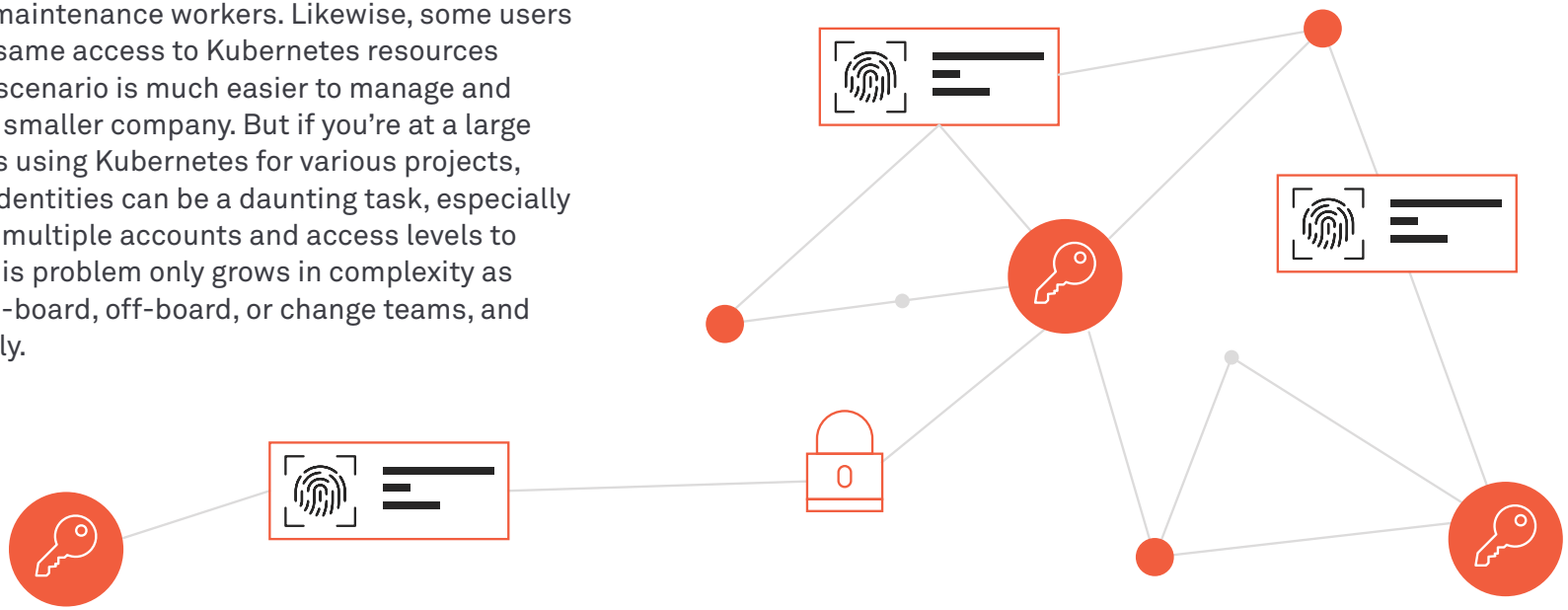
If a cluster goes down, you can’t troubleshoot problems without losing valuable time. You also can’t easily obtain insights on cluster performance to deliver better resource utilization. And if there are dozens of potential software versions in use, managing all of them across the organization is nearly impossible. All of these issues will eventually lead to inconsistent performance and reliability issues and an increase in security risks and development and maintenance costs.

Operational Complexity and Overhead

Identity and access management are critical components of many applications. However, when various teams begin the effort of deploying broad sets of clusters, it creates several challenges when it comes to managing authentication credentials, resource sharing, and security.

You can think of Kubernetes multi-tenancy in terms of a hotel. In a hotel, the concierge needs to be able to map each guest and their activities to their assigned room. Similarly, admins need to be able to map each user and their activities to their cluster so they know who is doing what and when. When it comes to resource sharing, not everyone needs the same access to resources as others. In a hotel, guests don't need the same access as the concierge and maintenance workers. Likewise, some users don't need the same access to Kubernetes resources as others. This scenario is much easier to manage and implement at a smaller company. But if you're at a large company that is using Kubernetes for various projects, administering identities can be a daunting task, especially when there are multiple accounts and access levels to manage. And this problem only grows in complexity as more people on-board, off-board, or change teams, and projects multiply.

As teams expand their usage of Kubernetes, clusters will exist in different pockets each with differing policies, roles, and configurations in their usage. This variety makes it incredibly challenging to create consistency across clusters. Operators lose the flexibility to define user roles, responsibilities, and privileges to ensure the right people are performing the right tasks within the environment. In addition, when there's a lack of governance and access control, operators can't identify role violations, assess governance risk, and perform compliance checks. When more time is spent putting out fires, there is less time for efficient operations.



Empowering Both Developers and Operators

When adopting Kubernetes and other cloud native services, it's important to think about how they'll impact the engineering culture. Kubernetes is popular amongst developers because it enables them to spin up their own environments with ease and agility. While this isolated autonomy gives them greater flexibility, it makes it tough for operators to create standardization across organizational clusters.

The challenge becomes how much freedom should developers have when it comes to selecting and developing new technologies that accelerate innovation? And how much structure should there be in place to deploy policy and secure governance requirements without sacrificing innovation?

"I understand that developer autonomy is very important, but every time a development team chooses a new technology or a new way to configure an existing technology, that's an expansion of an attack surface that I'm very concerned about," says Charles Betz, Analyst at Forrester Research. "We have to keep our environments secure, well patched up, and up-to-date, and if you only have one or two ways that things are configured that means your staff is more likely to do the right thing as oppose to the infinite levels of variation on a hundred different ways of configuring Kubernetes."

"Businesses want to create software that's more integrated, but they lose the opportunity to integrate those things if they're increasing the transaction costs by introducing a plethora of discord and governance models."

Developers want their own community clusters and to do it in their sandbox. They want to install their own applications and don't want to talk to a different team when they do it. So how can you give them that while at the time enforcing the standards that you need to? How do you make sure those clusters follow a certain blueprint that has the right access control rules? How do you ensure that sensitive information like credentials are distributed in the right way? And how do you ensure the right versions of software or workloads are available?

"I think it's about finding the right balance between that flexibility and governance," says Knaup. "Giving developers the flexibility they want to leverage the benefits of cloud native, while informing operations about the different stacks that are provisioned so they can enforce governance."

CHAPTER 2

Rethinking Your Governance Model

The problem with a majority of governance models is that they aren't continuous. As development teams adopt cloud native technologies and evolve to more agile methods, such as continuous flow and continuous iteration, they are up against decades of policy that assume an older model and don't fit into a month-long sprint. While governance models need to be restructured, if they're too restrictive, it can discourage developers and prevent innovation. What organizations need is to be able to foster this kind of agile and speedy acceleration that leads to a devops orientation with good, solid governance practices folded into the mix.

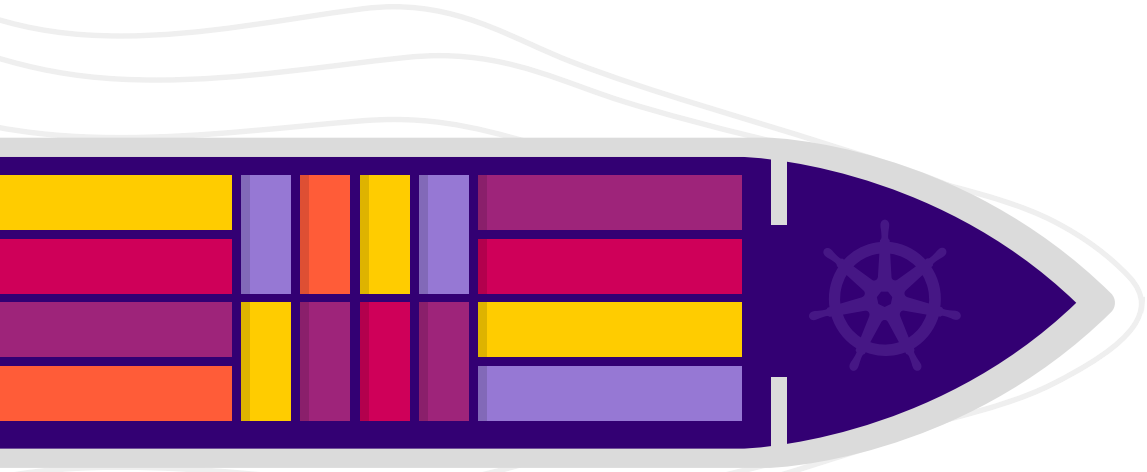
In this section, we provide a framework and blueprint to continually balance the needs of everyone on your team.

Multi-cluster Visibility and Management

As the number of clusters grows, operators are forced to spend increasing amounts of time managing clusters and less time doing actual work. They need to be able to centrally view, manage, and consolidate disparate clusters as they are discovered so that they can better optimize resources in a cost-effective manner and troubleshoot issues without losing valuable time.

Configuration Management

In order to reduce the potential vulnerable surface area of software in use, operators need to maintain granular control over how and where clusters are provisioned, as well as which versions of software can be used within project efforts. This level of control can help organizations meet risk and compliance demands and simplify the provisioning of services.



Authentication and Access Management

Organizations can have differing governance and access control requirements depending on the type of business they are in. The access requirements for different roles may also evolve as employees change job roles and leave the organization. Operators need a simplified way to manage the individual logins and permissions and service the needs of a wide range of clusters with centralized policy-driven capabilities.

Building and Maintaining Line of Business Relationships

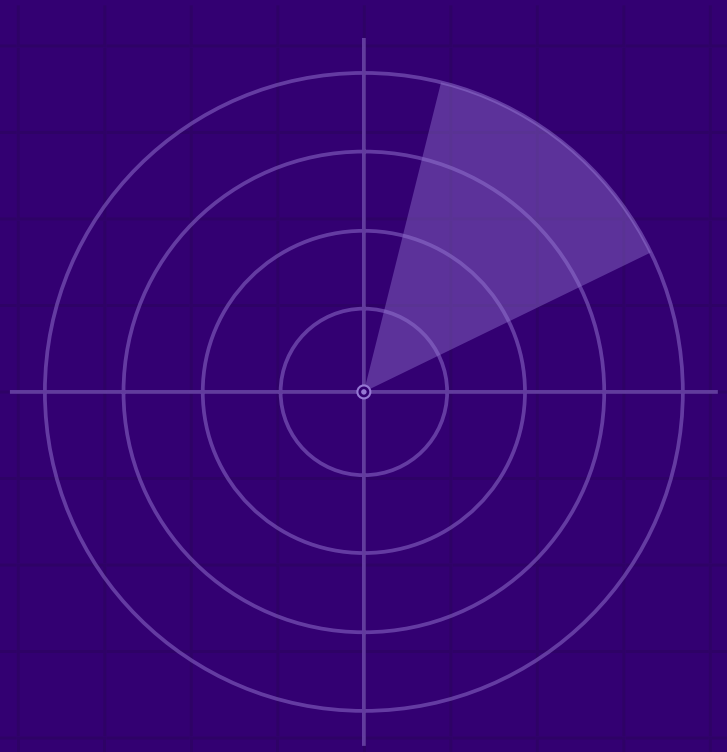
Finally, a key goal is to avoid conflict between IT's efforts to monitor and to support the needs of the business and its strategy in innovation and revenue acceleration. Operations should not restrict technology, instead it should look to simplify its management for development teams. Although developers like the self-service model of Kubernetes, it's become clear that enterprises want some control and opinions regarding which infrastructure, provider, and application services are best for the organization.

CHAPTER 3

How D2iQ's Kommander Delivers Governance and Manageability Across the Kubernetes Landscape

Balancing the needs across the organization can be a challenging task, so you need a tested framework and a resilient solution that brings teams together and benefits everyone. That's where D2iQ's Kommander comes in.

Developed to address the broad issues caused by cluster sprawl, D2iQ's Kommander gives teams the ability to deliver scale, consistency, governance, and operational efficiency for disparate clusters across an organization's on-premise and cloud footprint. With Kommander, multiple teams can create and maintain domain-specific clusters, while providing Operations with cluster visibility. This capability allows for greater standardization across clusters, without interfering with the day-to-day business functions and requirements that different clusters support.



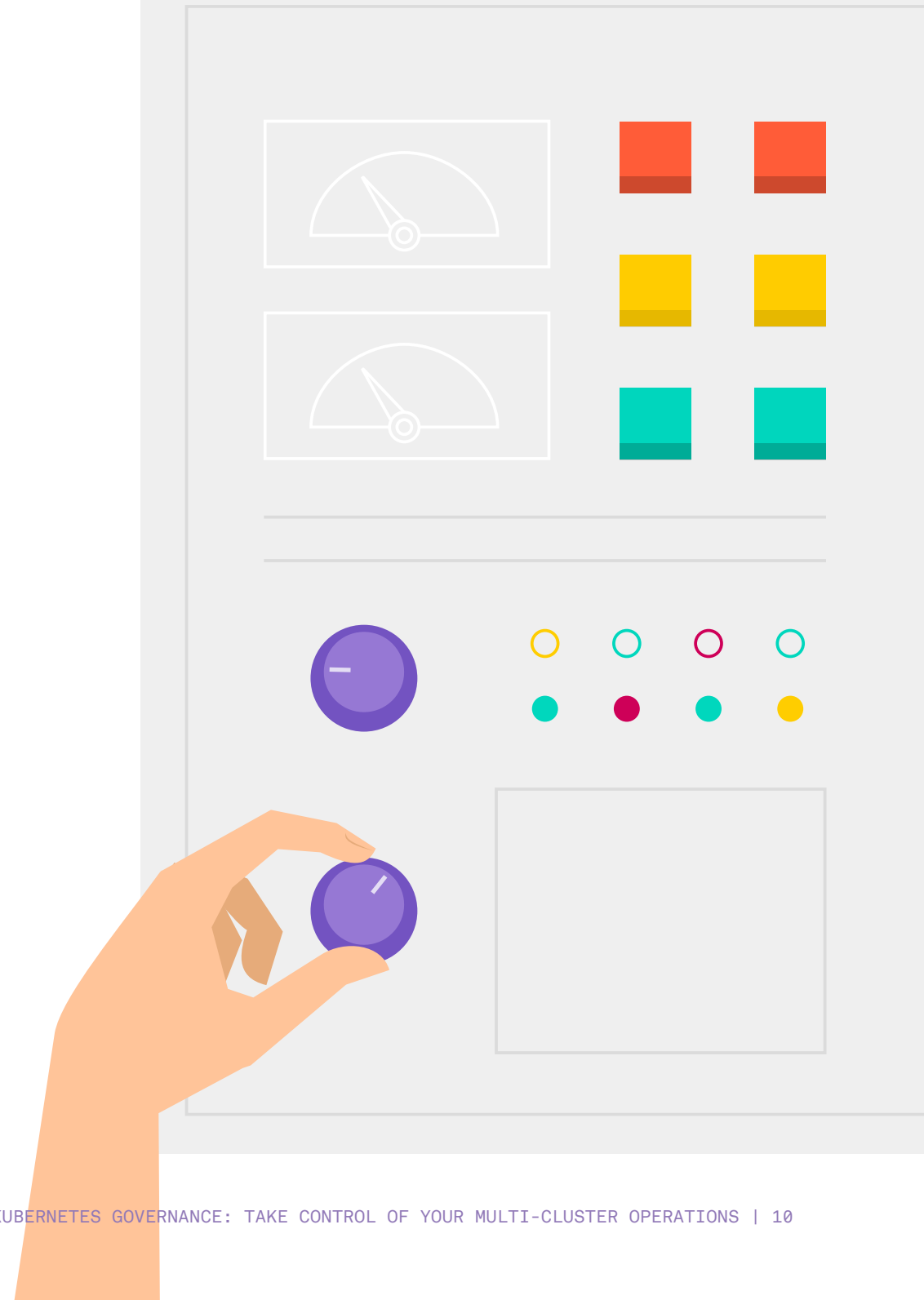
Key benefits of Kommander

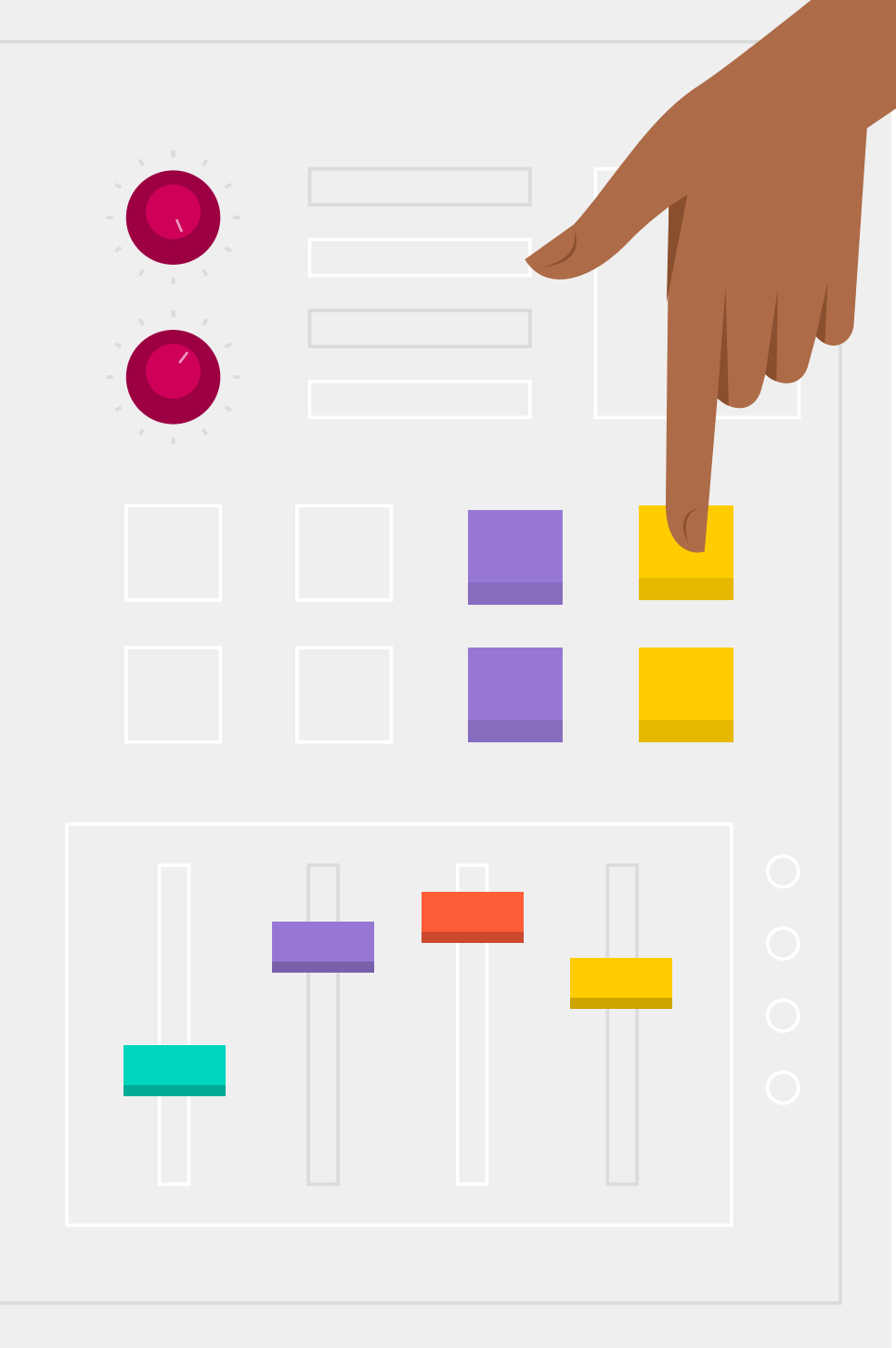
Multi-Cluster Visibility and Management

As teams deliver a variety of Kubernetes implementations across different infrastructures, it becomes increasingly harder to track and manage all of them. Kommander's single management plane provides instant visibility across any cluster you are monitoring, regardless of the infrastructure or distribution being used. As issues are detected, they can be resolved before they escalate, saving valuable time. And when insights are obtained on cluster performance, workload metrics, and cluster activities, it can help your organization understand resources and utilization.

Configuration Management

Maintaining control over how and where clusters are provisioned, as well as which versions of software can be used within project efforts, is critical in many environments with global regulatory burdens. Without this capability, managing an unpredictable set of services and versions across the organization is next to impossible. Kommander comes with an out-of-the-box configuration manager that simplifies and delivers consistent configuration for services and cross-cluster operations. In addition, Kommander's catalog of prevailing open source technologies helps you quickly deploy services to multiple clusters, while governing which versions of software can be used within project efforts. This ability significantly reduces security exposure and simplifies the supportability of services.





Authentication and Access Management

Kommander provides single sign-on and federated role-based policy across your organization's clusters. With Active Directory integration and security based on Open Policy Access (OPA), credentials (secrets), and permissions, users can leverage their existing authentication mechanism already in place and the health of the environment can be effectively monitored. With Role-Based Access Control (RBAC), admins can define user roles, responsibilities, and account privileges, and flexibly configure access as a user's role within the organization changes. RBAC security for multiple clusters means high-availability clusters and reduced time for efficient operations.

Building and Maintaining Line of Business Relationships

As organizations create new clusters within their organization, it can be critical to create lines of separation across clusters. Kommander's centralized authentication and authorization can empower division of labor across a wide variety of roles to ensure the greatest management flexibility possible. With Kommander, custom needs can be met and critical services can be deployed as needed by individual teams. At the same time, operational teams can create standardization over which infrastructure, provider, and application services are best for the organization. When your organization is able to deliver balance between developer flexibility and operational control, team morale and productivity goes up and the number of redundant efforts and wasted resources goes down.

Conclusion

As you move swiftly to adopt new Kubernetes-based applications and cloud native services, needs will arise for simplifying ongoing operations and ensuring organization control over an expanding Kubernetes footprint.

Kommander is designed from the ground up to help your organization rein-in wasted resources, deliver organization-wide governance over cluster use, and empower greater division of labor within the organization for both tremendous control and flexibility.

To learn more about D2iQ's Kommander, click here to sign up for a free demo.

Free demo

D2iQ, formerly Mesosphere, is the leading provider of enterprise-grade cloud platforms that enable organizations to embrace open source and cloud native innovations while delivering smarter Day 2 operations. With unmatched experience driving some of the world's largest cloud deployments, D2iQ empowers organizations to better navigate and accelerate cloud native journeys with enterprise-grade technologies, training, professional services and support. Whether you are deploying your first Kubernetes workload, optimizing your business analytics with Spark or Jupyter, or looking to educate your developers on the benefits of cloud native, D2iQ has the expertise, services, and technology to enable you on the journey.

d2iq.com