KUBERNETES MULTI-CLUSTERS: HOW & WHY TO USE THEM

Containerized deployments offer more scalability and availability improvements over traditional deployments. Even with these improvements, complex applications can quickly overwhelm containerized environments without proper management. Kubernetes helps organizations to:

- Effectively orchestrate containerized environments
- Efficiently manage the underlying resources and user demands

However, there are situations where a single Kubernetes cluster is unable to handle the application load or properly distribute the application to end-users. In such instances, multi-cluster Kubernetes solutions are ideal for distributing the work across multiple clusters.

In this article, we'll take a look at Kubernetes multi-cluster implementations.

(This article is part of our Kubernetes Guide. Use the right-hand menu to navigate.)

What is a Kubernetes multi-cluster?

Kubernetes multi-cluster is an environment with multiple Kubernetes clusters. They can be configured in several ways:

- Within a single physical host
- With different multiple hosts in the same data center
- In different regions within a single cloud provider
- Utilizing multiple cloud providers and distributed across multiple regions to provide a truly global multi-cluster environment

A multi-cluster environment is not simply running multiple Kubernetes clusters.

Instead, it should consist of proper application deployment strategies to distribute and maintain the application across multiple clusters with appropriate tools and practices baked into the DevOps process.

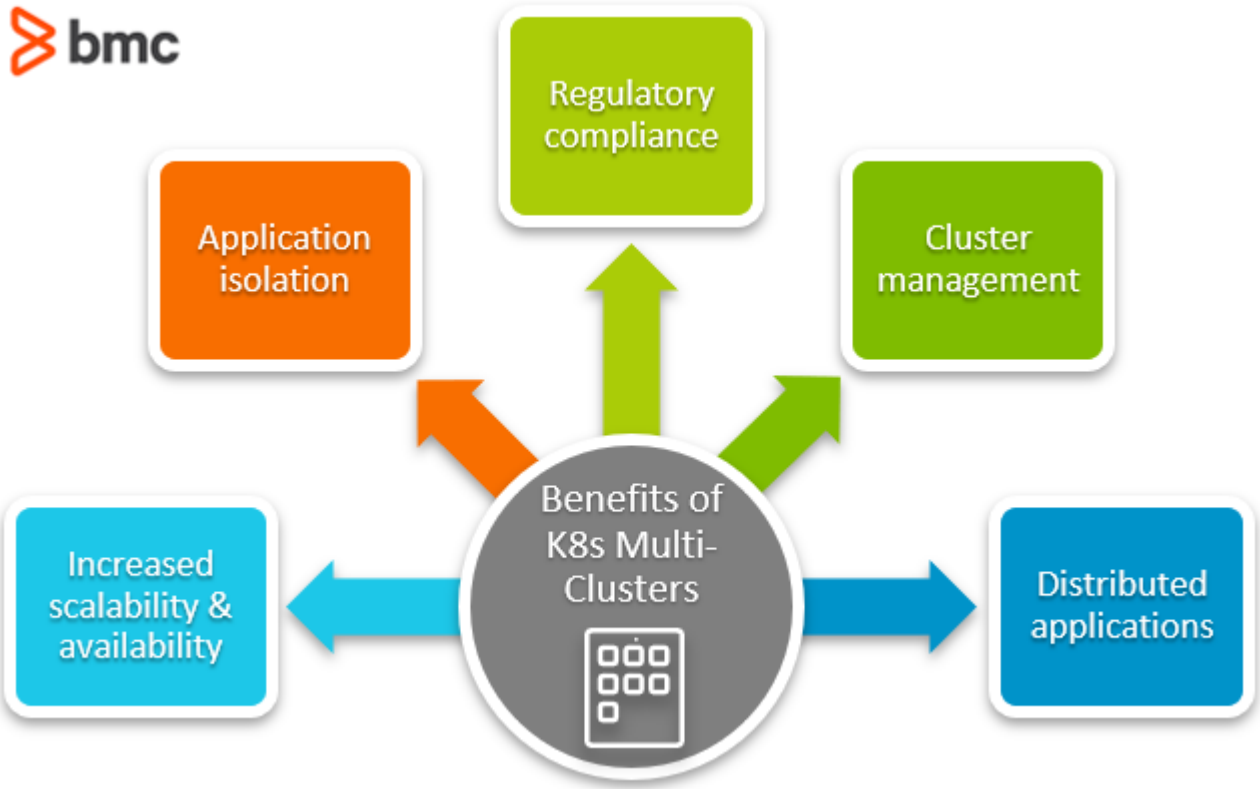
(See how containers & K8s work together.)

Why use Kubernetes multi-cluster?

Configuring and managing multi-cluster Kubernetes deployments is not an easy task. So, why would an organization allocate valuable resources for this endeavor?

The reason: user experience.

When it comes to maintaining large-scale applications with thousands of users, multi-clusters are the core component to providing an excellent user experience. Let's see all the benefits a multi-cluster environment offers.



Increased scalability & availability

With multi-clusters, users can distribute the application across different regions, exponentially increasing the availability. With that, end users will have a much better experience interacting with the applications with:

- Less latency
- Faster performance

The same can be applied for scaling instead of a single cluster trying to handle all the scaling requirements. Multi-cluster deployments are distributed across multiple clusters and scales according to a load of a specific cluster. It also drastically reduces resource consumption of a single

cluster and leads to less load on backend services like databases.

Application isolation

Multi-cluster deployments allow organizations to achieve true application isolation with separate Kubernetes clusters. They can be either:

- Different clusters for staging and production (simple separation)
- Different clusters facilitating distinct applications

With different clusters, any cluster failure or configuration change affects only that specific cluster. This isolation helps users to:

- Easily diagnose issues
- Experiment with new features
- Carry out optimizations without causing disruptions to other clusters

Regulatory compliance

With applications serving users across the globe, one major considerations is adhering to regulations and policies of different regions and countries—policies like GDPR and the California Consumer Privacy Act (CCPA). This is particularly prevalent when user data is involved, as most regions require the user data to reside within geographical limits.

Importantly, applications hosted in a single cluster would not be able to comply with all these compliance requirements. Multi-cluster deployments, on the other hand, enable organizations to meet these requirements by locating the Kubernetes cluster in specific geographical regions.

Applications that cater to specific geographic regions with isolated Kubernetes clusters can:

- Limit the scope of regulatory requirements
- Enables organizations to target specific clusters to meet different requirements and gain regulatory compliance relatively easily

Cluster management

Setting up and managing multi-cluster deployments can be a complex and time-consuming task. Fortunately, a properly configured multi-cluster environment can reduce the overall management complexity.

With proper cluster management, you can have better visibility with standardized processes and proper DevOps practices while managing all the infrastructure and applications from a centralized location. This can be a significant factor in managing operations expenditure—users can effectively allocate resources.

Additionally, in a multi-cluster environment, the production issues are isolated to that specific cluster. That means developers only need to investigate a limited scope to identify and fix issues without affecting other clusters. Crucially, a complete cluster failure would not cause any downtime in a multi-cluster environment. The other clusters step in to ensure availability.

Users are not vendor-locked into a true multi-cloud deployment. When new features or cost savings are available with new cloud providers, users can seamlessly transfer applications, with minimal

modifications, between different providers or Kubernetes clusters.

Distributed applications

With increasing edge computing requirements and the popularity of Internet of Things (IoT) services, distributed computing is becoming the preferred solution.

A multi-cluster Kubernetes environment is the ideal platform for that.

Developers can simply create containerized applications that can be deployed in Kubernetes while providing a powerful orchestration engine distributed across multiple regions. This eliminates the need to cater applications to different environments and infrastructure since these containers can act as the edge nodes to facilitate a global IoT network.

Kubernetes multi-cluster architecture

When it comes to multi-cluster architecture, there are two considerations:

- The Kubernetes multi-cluster configuration
- The application architecture

Kubernetes multi-cluster architecture

There are primarily two methods to handle a multi-cluster Kubernetes environment:

- The Kubernetes-centric federated method
- The network-centric method

Different vendors offer variations and improvements upon these methods.

A **Kubernetes-centric** method uses tools like kubefed to manage multiple Kubernetes clusters from a centralized location (federated clusters). Other tools like Shipper and Admiralty can facilitate:

- Rollout strategies
- Multi-cluster orchestration
- Workload distribution

The **network-centric** method aims to provide a robust network to facilitate communication between different clusters. The clusters act as separate entities, yet applications can communicate with each other using a network. The network approach is based on mesh networking principles and has adapted service mesh concepts to the infrastructure. An example of this approach is:

- Linkerd service mirroring

Most organizations prefer the network-centric method due to the lack of maturity of Kubernetes-centric multi-cluster configurations. Even with the increased complexity of managing a mesh network, this approach effectively handles multi-cluster environments with automated pipelines and standardized configurations with GitOps, reducing management complexity.

Multi-cluster application architecture

There are two application architectures to consider when developing multi-cluster applications—replicated apps and service-based apps.

Replicated application architecture

This approach replicates the complete application across multiple Kubernetes clusters, and it is the simplest way to deploy a multi-cluster environment. With the complete application distributed across multiple clusters in different regions, end users can be directed to the nearest cluster, helping you to both:

- Manage the load effectively
- Provide the best user experience

Coupling this replicated application architecture with proper networking infrastructure and load balancers can handle cluster failure with ease. When a cluster faces an issue, users can be redirected to a different cluster without any effect on the end-user or application functionality.

Service-based application architecture

Another development architecture is to divide the application into multiple components or services and distribute them across clusters. It provides the best isolation and is the best option for obtaining regulatory compliance. However, all this comes with increased application complexity.

With different components running on different clusters, the communication layer between each component should be designed to handle user and application traffic. When configuring a fault-tolerant architecture, organizations will need to:

1. Treat a set of clusters with different components of the application as a single application
2. Replicate it across a different region

This increases the number of overall clusters and operational costs.

Even with the increased complexity, however, this approach provides the best security and performance for the application. With its component-based approach, the regulatory scope can be confined to a single component in a defined region without having to modify the complete application.

This approach also enables target troubleshooting and optimizations for specific components.

Platforms for Kubernetes multi-clusters

There are a lot of tools that support configuring or managing Kubernetes multi-cluster environments. Following is a list of such platforms.

- Rancher is an enterprise-grade container management platform with built-in support for Kubernetes multi-cluster management on various platforms from on-premise deployment to multi-cloud.
- Fleet is a GitOps-at-scale project designed to facilitate and manage a multi-cluster environment. It started as an extension of the Rancher but has since developed into its own project.
- Google Anthos is designed to extend the Google Kubernetes engine across hybrid and multi-

cluster environments. This enables users to provide a consistent Kubernetes experience while having a centralized multi-cluster management environment.
- Microsoft Azure Arch is designed to extend Azure management and services across hybrid and multi-cloud environments. It is not limited to Kubernetes but can be used to extend Kubernetes clusters and Azure Arch providing a centralized location to manage, secure, and govern multi-cluster Kubernetes deployment.
- VMWare Tanzu is platform offered by VMWare to manage and secure Kubernetes infrastructure across multi-cloud environments.

Multi-clusters offer multi benefits

Kubernetes multi-clusters is an extensive field composed of multiple implementations and integration methods dictating multi-cluster deployments.

With all the advantages it offers for large-scale applications, it is paramount that you select the appropriate architecture for your multi-cluster environment and tool stack to manage the Kubernetes multi-cluster centrally.

Related reading

- BMC DevOps Blog
- Kubernetes Deployments Fully Explained
- How To Set Up a Continuous Integration & Delivery (CI/CD) Pipeline
- Continuous Delivery Metrics
- How to Setup a MongoDB Cluster
- What Is DevSecOps? Combining Development, Security & Operations