ActualTech Media

PLATFORM9

Considerations for Distributed Kubernetes–From the Data Center to the Edge

CONTENTS

IN THIS PAPER

Kubernetes is moving beyond the data center to micro data centers, point-of-presence facilities, and even the edge. Managing Kubernetes is difficult when it's isolated to a data center, but multiple deployments in different environments compound your management challenges.

Platform9 understands these challenges, and has the tools and expertise to help you leverage Kubernetes without becoming drawn into additional operational overhead.

Kubernetes is widely recognized as the platform of choice for running efficient, distributed, containerized applications. It's also common to think of Kubernetes in terms of a single, large cluster or set of clusters running in a data center. This is certainly a common deployment approach, but it's not the only one.

VARIETY OF DEPLOYMENT MODELS

Kubernetes can be deployed in many kinds of environments. The platform is well-suited to run in micro data centers that are closer to the edge. A branch office may only need a small cluster to support the remote operations of an office. This kind of use case can typically run components that fit into a single rack. Kubernetes can also run at point-of-presence sites. For example, retailers may deploy Kubernetes clusters to physical stores and distribution centers to run applications, store data locally, and coordinate operations with centralized processes.

> It's clear there's a spectrum of cluster deployments. When you're considering and planning your Kubernetes strategy, it's important to understand where your deployment falls on that spectrum because there are requirements particular to each.

Kubernetes may also run at edge locations to support Internet of Things (IoT) systems. A manufacturer may deploy Kubernetes in multiple locations within a manufacturing facility to collect IoT data and perform preliminary processing and analysis. This kind of processing close to the environment can help compensate for unreliable networks and long latencies that can reduce the effectiveness of highly centralized processing.

It's clear there's a spectrum of cluster deployments. When you're considering and planning your Kubernetes strategy, it's important to understand where your deployment falls on that spectrum because there are requirements particular to each. A data center cluster, for example, may have ample resources to scale up the number of pods in a deployment, while a micro data center is more constrained.

In the case of Kubernetes deployed at the edge, you should consider how continuous integration/continuous deployment (CI/CD) will work with potentially unreliable networking. The number of sites can quickly become a factor you need to consider. Updating a single cluster in a data center is challenging enough—updating hundreds of point-of-presence sites is even more difficult.

NETWORK ISSUES AND MULTIPLE KUBERNETES SITES

When deploying Kubernetes clusters to multiple data centers and remote sites, the quality and capacity of network infrastructure can impact the overall performance of the platform.

Data centers typically have high-bandwidth connectivity. Clusters are composed of servers with high-speed network connections between them and run in an environment with multiple racks. The combination of high-bandwidth networking and the ability to distribute pods over multiple racks provides the optimal environment for performant and reliable Kubernetes clusters.

That level of network capacity extends beyond single data centers, too. Hybrid clouds composed of resources in a data center and in one or more public clouds can have high bandwidth dedicated direct connections between sites.

Micro data centers and point-of-presence deployments typically won't have the same network bandwidth available in data centers and within hybrid clouds. Edge processors and IoT devices are even more constrained in terms of bandwidth. This is one of the reasons it's advantageous to deploy Kubernetes to multiple locations—with remotely deployed clusters, the processing is brought close to where the data is being generated. Local processing reduces the amount of data that must be sent to the data center and gives local sites the ability to function autonomously in the event of a network outage.

This highlights another factor to consider when planning your Kubernetes strategy: There may be periods of extended outage. Short outages in well-architected deployments won't significantly adversely affect operations. Longer outages, however, will cause different clusters to get out of sync. Changes to data will accumulate in the clusters that are isolated by the network outage and when connectivity is restored, recovery can begin and data can be synced. Depending on the duration of the network outage, the recovery may be long enough to impact performance and service delivery.

LOCAL DATA PROCESSING

The ability to process data locally is a key advantage of having multiple Kubernetes deployments. This approach, however, does make it more difficult to deploy services to multiple clusters. Consider, for example, the various use cases for distributed Kubernetes.

A retailer may deploy Kubernetes to a centralized server in a store, as well as to point-of-sale systems. The centralized server could collect data from point-of-sale systems, generate real-time reports and dashboards for local managers, as well as coordinate services running in a corporate data center or cloud.

5G is changing how businesses deliver services and collect data. With significantly more bandwidth than previous generation networks, 5G enables more data-intensive applications. To achieve the higher bandwidths, 5G networks use higher frequency signals. The disadvantage of this is that 5G networks need more cell towers because the signal degrades over long distances. Those cell towers all have to run networking services, so managing the deployment of software is a significant challenge for carriers. Distributed Kubernetes can help here, as well. Networking services can be deployed in containers and updated as needed from a central location (see **Figure 1**).

> 5G is changing how businesses deliver services and collect data. With significantly more bandwidth than previous generation networks, 5G enables more data-intensive applications.

IoT devices can also require local data processing. Machine learning models for analyzing images or controlling autonomous vehicles are best run locally to avoid

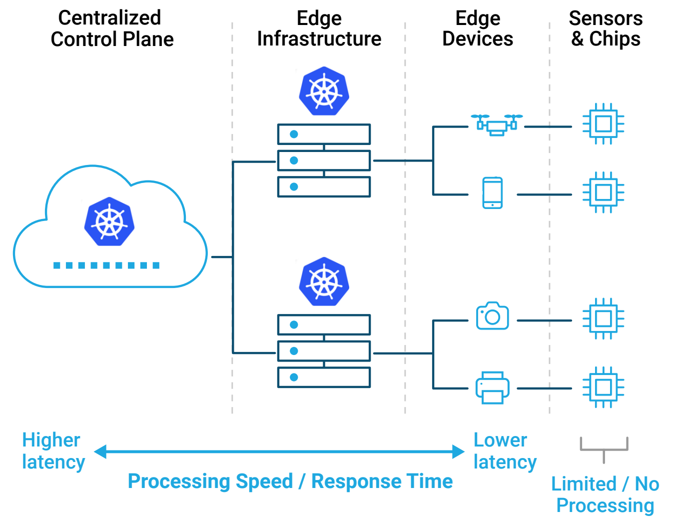

Figure 1: The central management of data centers and edge locations

unnecessary latency introduced by centralized processing. Distributed Kubernetes can again mitigate the challenges of managing software deployed on thousands of geographically distributed devices.

These three examples share some common requirements. They all need consistent and reliable methods to update software. In addition, these update methods have to be essentially zero-touch and automated in order to scale.

Stateful services, such as databases, bring another set of challenges to managing multiple Kubernetes clusters. These services need persistent storage, so you'll need to understand how to architect the cluster to deliver the needed read and write performance. To enable some level of autonomy within the cluster, plan for graceful degradation of services when the network is down. For example, data stored on a remote cluster could be cached locally so that data is available to local processes. When the network is available, the databases can sync and caches can refresh.

SECURITY CONSIDERATIONS

Security operations need to be coordinated across all environments—especially encryption for data at rest and key management.

Encryption at rest is required to comply with a wide variety of regulations, especially when personally identifying information (PII) or other sensitive data is stored. There may be multiple levels of encryption, starting with the storage device.

Middleware, such as databases, may also provide for encryption. For example, some relational database management systems allow data modelers to specify that particular columns of data should be encrypted. Applications can also provide for their own encryption policies and methods. Regardless of the combination of encryption options you may employ, they need to be coordinated across all Kubernetes environments.

Key management is another security process that will need to be managed across environments. Key management services can provide all the required functionality, but you'll still need to define policies and monitor operations. For example, you'll want to define policies for key rotation and be able to verify the operation occurs.

CENTRALIZED MANAGEMENT OF MULTIPLE ENVIRONMENTS

Multiple environments can be a challenge to manage, especially as the number of sites grows. Some clusters will be in the data center and can be managed to some degree with existing tools. Clusters at point-of-presence sites and on the edge need to be monitored and managed to maintain the necessary quality of service.

> Clusters at point-of-presence sites and on the edge need to be monitored and managed to maintain the necessary quality of service.

Fortunately, Kubernetes has auto-healing capabilities that reduce the need for human intervention. Unhealthy pods are replaced automatically without requiring a DevOps engineer to log into a cluster, identify the failing pods, and replace them. Auto-healing also promotes autonomy—if the network is down, the cluster can continue to function and correct for some failure within the system.

FOCUS ON YOUR CORE BUSINESS OBJECTIVES

Kubernetes is moving beyond the data center to micro data centers, point-of-presence facilities, and even the edge. Managing Kubernetes is difficult when it's isolated to a data center, but multiple deployments in different environments compound your management challenges.

Platform9 understands these challenges, and has the tools and expertise to help you leverage Kubernetes without becoming drawn into additional operational overhead. Partnering with Platform9 gives you the performance and efficiencies of Kubernetes, while still allowing you to focus on your core business objectives.