Kubernetes: Your Hybrid Cloud Strategy

Simplifying the deployment and management of applications in a hybrid cloud

By Rick Ross, Google Cloud Platform Customer Engineer

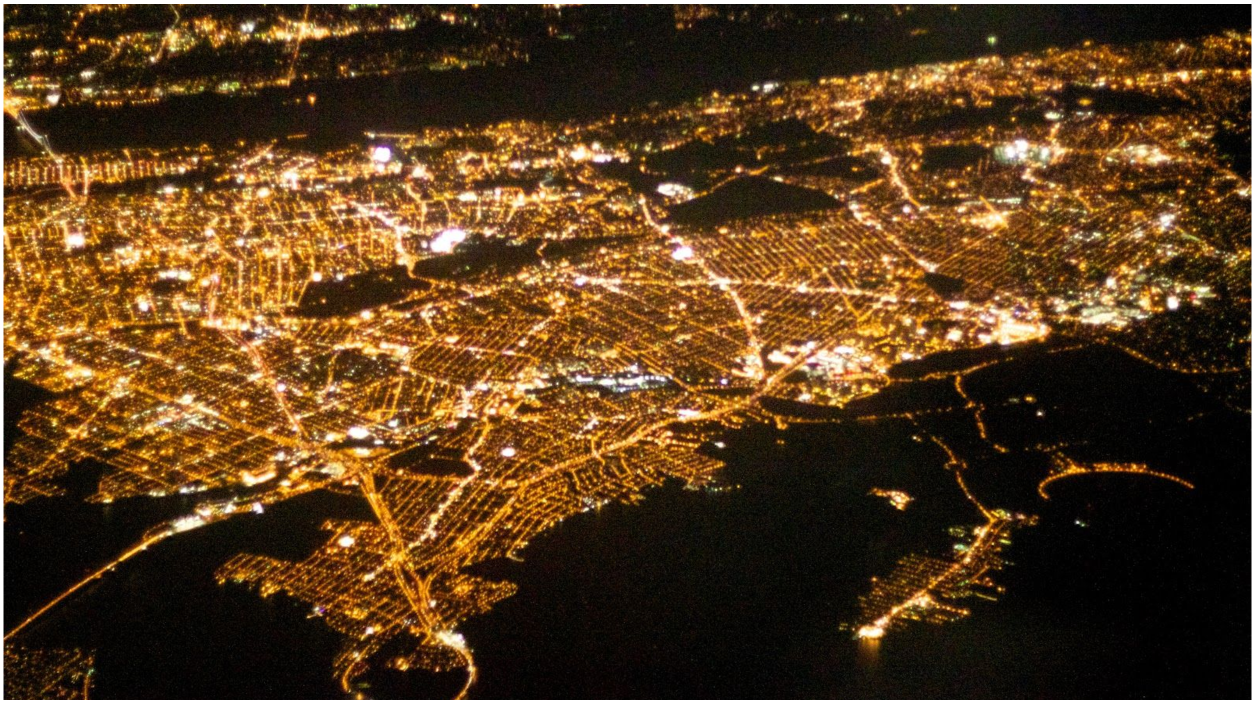


Image credit: Alan Levine

Introduction

As more and more enterprises look at leveraging the capabilities of public clouds, they face an array of important decisions. For example, they must decide which cloud(s) and what technologies they should use, how they operate and manage resources, and how they deploy applications.

A lot of technologies can help, but not all of them are equal. If you have heavily invested time, money, and energy in creating software, shouldn't you have the ability to deploy and manage that software seamlessly across your hybrid environments and avoid the costs of rewriting it? How can you scale your software to meet customer demand? Do you want to deploy your software where it makes sense, whether on-premises or to a specific public cloud, based on business value?

This white paper discusses how Kubernetes is the answer to your hybrid cloud strategy and how it provides a holistic solution that simplifies your deployment, management, and operational concerns. The paper also provides links to additional resources that can help you refine your hybrid cloud strategy.

Kubernetes: What is it?

So what exactly is Kubernetes? According to Wikipedia, “Kubernetes is an open-source container orchestration system for automating the deployment, scaling and management of containerized applications.” For some people, that definition might not be enough.

Stated another way, Kubernetes is a platform that provides a set of standards that helps you run applications and ensure that they can handle varying traffic. Kubernetes also helps you with day-two operations such as upgrading, monitoring, and logging.

Problems of the past

Even though enterprises try to maintain standards for applications deployed in data centers, as new technologies are introduced, more and more unique configurations are required. Over time, managing the differences between all of these unique configurations becomes challenging and requires extensive documentation to understand how to manage and keep the data center operational.

In the past, the standard practice was to create “snowflakes” in data centers. That is, each project was like a snowflake, with its unique requirements and configurations. Over time, the projects became unmanageable. Anticipating the number of users and predicting how much performance you could get out of your infrastructure was a black art, and usually ended up with servers sitting idly most of the time. Installing operating system updates and deploying new versions of applications generally required downtime and was usually done at odd hours.

What a hybrid strategy offers

Kubernetes is an ideal foundation for a hybrid cloud strategy because it provides consistency, no matter if it’s been deployed on-premises or to one or more public clouds. Because Kubernetes is an open source project, you have several options to choose from:

- Roll your own from open source.
- License Kubernetes from a vendor.
- Use public cloud offerings.

Using Kubernetes provides a consistent platform everywhere. Now you can choose where to deploy your applications based on business decisions. Imagine a future where the business says, “We need to run this application on-premises today, and in China six months from now.” With Kubernetes, you can easily do that.

After you deploy an application, you must keep it running and ensure that it can scale based on varying user demand. With Kubernetes, the platform provides the ability to automatically scale applications up and down, resulting in better utilization of the underlying infrastructure.

But deploying applications is only the first step. After the applications are running, day-two operations are vital to keep them running and operational. For example, as new software versions are released, you can automatically upgrade the software and avoid downtime. This approach provides a much better customer experience.

Kubernetes and Google

Behind the scenes at Google is a technology called Borg, which is used to manage Google's massive infrastructure. As Borg evolved and matured, it became one of Google's go-to technologies, but it wasn't publicly available. Open sourcing Borg wasn't feasible, however, so a group of Google developers decided to create an open source project called Kubernetes. Their goal was to take the knowledge that Google had learned about managing billions of containers and bring that to the world.

The open source world enthusiastically joined in, and Kubernetes has grown tremendously, with over 1700 contributors. Companies such as Red Hat, Microsoft, IBM, Intel, Pivotal, and VMware are actively participating to continue to evolve Kubernetes.

Google offers a managed version of Kubernetes on Google Cloud Platform (GCP), called Google Kubernetes Engine (GKE). GKE offers enterprise-grade compliance and auditability, a world-class security-hardened environment, and zero-downtime upgrades of the operating system and Kubernetes.

Another offering on GCP is Stackdriver, which provides monitoring and management for your containers, applications, and infrastructure. Stackdriver helps you understand how your application and Kubernetes are running. Stackdriver also provides you with great insights, and even allows you to debug a production application without stopping the code from running! Best of all, Stackdriver works with multiple clouds and on-premises infrastructure, giving you insight into your running applications everywhere.

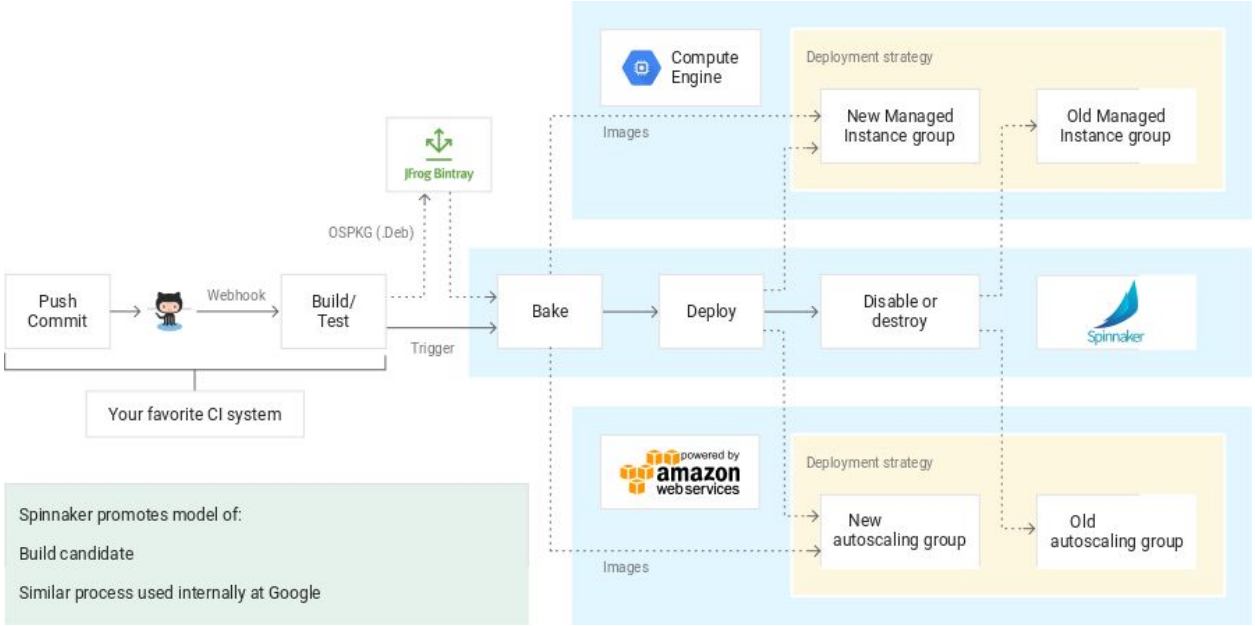
At Next 2018, Google announced GKE On-Prem, a service that brings the GKE experience inside your data center. Now, with a single pane of glass, you can manage all of your Kubernetes clusters from one location. By combining GKE On-Prem with Stackdriver, you get the metrics, logging, and metadata from all of your Kubernetes environments in a single comprehensive view, and you gain the ability to quickly understand and act on behavior.

Bringing it all together

Now that you see how Kubernetes lays the foundation for a hybrid cloud strategy, what else do you need to bring everything together? The answer is automation, one of the foundational principles of Site Reliability Engineering (SRE), a practice that Google created to run their production systems.

To help automate the building, testing, and deployment of their applications, many enterprises are adopting continuous integration/continuous delivery (CI/CD) pipelines. Similarly, some enterprises are using pipelines to deploy their infrastructure. As you expand beyond your data center into public clouds, it's even more critical to use automation so that you establish a repeatable process that significantly reduces the chance of mistakes resulting from manual processes.

For example, the following diagram shows how a CI/CD pipeline that uses Spinnaker can automatically deploy applications in a hybrid environment to both AWS and Google Cloud Platform (GCP). When a developer checks in code, a CI/CD pipeline such as Jenkins or Concourse kicks off, builds the application, and runs tests. The system then triggers Spinnaker to take the artifacts and deploy them to AWS and GCP.



Conclusion

Coming up with a hybrid cloud strategy is challenging, and until recently, there were few options that provided a consistent experience everywhere. With recent innovations and advancements on GCP, you can securely deploy applications at scale using an open source, hybrid cloud model. Google Kubernetes Engine (GKE), GKE On-Prem, Stackdriver, and many other application development and analytics services allow you to modernize your application landscape both on-premises and across public clouds. Google Cloud Platform allows you to focus on writing applications, making it easier than ever to deploy, manage, and operate them. While Google takes care of the underlying infrastructure and operational concerns, you can focus on adding business value.

Further reading

For a more in-depth look at the strategies discussed in this paper, see the following resources.

- Learn how to create a CI/CD pipeline using Azure Pipelines (formerly VSTS) and Google Kubernetes Engine.
- Read this multi-part series about hybrid and multi-cloud deployments, architecture patterns, and network topologies:
  - Hybrid and Multi-Cloud Patterns and Practices
  - Hybrid and Multi-Cloud Architecture Patterns
  - Hybrid and Multi-Cloud Network Topologies
- See these related topics:
  - GKE networking overview
  - Hybrid cloud articles
  - Spinnaker articles
  - Stackdriver articles