

1130 - Introduction to Azure Kubernetes Service and SAS Viya 2020.

Vasilij Nevlev, Analytium Group

INTRODUCTION

With the increasing acceptance of the cloud, there has been a massive shift in how software is designed, built, and deployed. Developers have turned to containerisation to make it easier and faster to create new products and release them. This paper has been written for people unfamiliar with terms such as Kubernetes and Docker containers. It discusses how SAS Viya 2020.1 uses these through Azure Kubernetes Service. By the end of this paper, you will be equipped with enough basic knowledge to understand technical deployment documentations.

WHAT ARE CONTAINERS?

A community definition states that:

“A container is a standardised unit of software that packages up code and all its dependencies, so the application runs quickly and reliably from one computing environment to another.”

Containers have been developed to minimise issues with software as they are transferred for use in different environments. For example, a developer’s laptop will have different configurations from a staging, testing, and production environment. When there are disparities in operating systems, software tools, security policies, network topologies, among many other factors, applications can behave unpredictably. In the context of the cloud, it may be challenging to support your product with the infinite combinations of components out there.

In other words, a container consists of a complete runtime environment: an application, all its dependencies, including libraries, binaries and configuration files needed to run it. This, in effect, abstracts the OS and any underlying infrastructure.

HOW ARE CONTAINERS DIFFERENT FROM VIRTUAL MACHINES?

Before containers, IT departments would rely on virtual machines to address compatibility issues. Operating systems (OS) kernel would be packaged together with the application for every deployment. This meant that the VM image would be large and compute-intensive. Virtual machines would have a hypervisor to oversee the multiple OS instances running on the server.

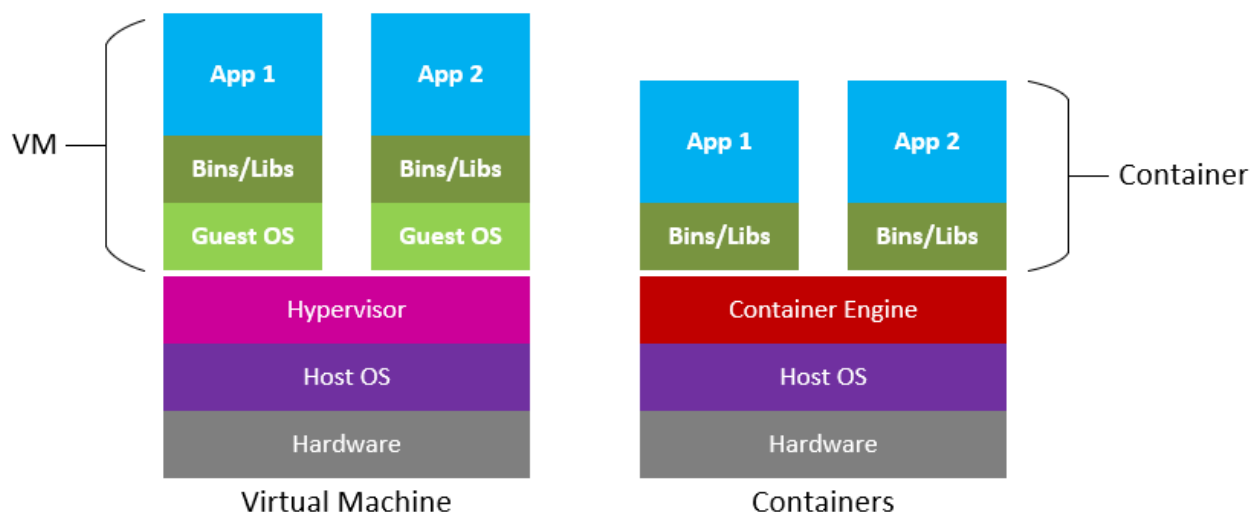


Figure 1 Difference between virtual machines and containers
(<https://cloudblogs.microsoft.com/opensource/2019/07/15/how-to-get-started-containers-docker-kubernetes/>)

In contrast, for containers, only the application, binaries and libraries needed for the software to run are included in the package. Each container shares the operating system kernel with the other containers.

WHAT ARE THE BENEFITS OF USING CONTAINERS?

Lightweight

As opposed to larger virtual machine images, a container image is typically much smaller. Having smaller container images has many implications, including:

- Highly portable and highly modular deployment method
- Ease in transferring files (uploading and downloading less prone to interruptions)
- Less storage is required
- Faster runtime, with no booting required
- Resources freed up as applications run just-in-time

Secure

Docker containers provide complete isolation of resources and access between containers.

Containers only share the kernel of an operating system. This read-only access means that users running multiple applications on multiple containers will not be able to interact with one another.

Standard

In 2015, an initiative called the Open Container Project (now known as the Open Container Initiative) was announced. Under the guidance of the Linux Foundation, it sought to standardise container formats and runtime software for all platforms. To jumpstart the project, Docker donated 5% of its codebase. Support from many of the industry's most significant players, including IBM, AWS, Microsoft, Google, HP, VMware, Red Hat, Oracle, Twitter, HP, and CoreOS, soon followed.

WHAT IS DOCKER?

Unix first introduced the concept of container technology in the 1970s. Because of its success in popularising the use of containers, Docker is now the household name. Docker is a tool used to create, deploy, and run applications using container technology. For many, it is the container platform of choice.

WHAT IS KUBERNETES?

Kubernetes is an open-source container orchestration software originally developed by Google but maintained by a cloud-native computing foundation.

The name Kubernetes is of Greek origin, which means helmsmen or pilot. Rightfully so, the software is used to arrange, coordinate, and monitor Docker containers.

Here are a few keywords that are essential to understanding Kubernetes:

Node is a machine (computer) running containerised applications. Nodes can be physical or virtual instances of a server. Each node can vary in size. Nodes host applications running in containers. Nodes can be independently started and stopped. Nodes can be spread across different data centres.

Kubernetes Cluster is a set of nodes.

Node Pools is a set of nodes grouped together to by workload or by resource requirements.

WHAT ARE THE BENEFITS OF USING KUBERNETES?

Velocity

Organisations that use Kubernetes can develop, deploy, and maintain their software components at very high speeds. They can introduce upgrades with continuous development.

Scaling of Systems and People

With Kubernetes, companies can make full use of the cloud's elasticity. Organisations can choose to scale vertically by augmenting resources (CPU, memory, I/O, storage, etc.) to your existing server. Another option is to scale

horizontally by increasing (or decreasing) the number of your servers to support your workload. Regardless of how you want to scale your deployment, Kubernetes helps ensure that your applications are unaffected by such changes.

Portability

The use of Docker containers and Kubernetes entirely abstracts the infrastructure, including hardware, OS, and necessary software. As a concrete example, when a server lease is finished, there is no need to reinstall SAS components.

Efficiency

With the security features of Kubernetes in place, developers no longer need to be constrained by the impact of their applications on others. Tasks from multiple users can be packed tightly on fewer machines, thereby maximising your hardware. Efficiency can be measured by the ratio of users over work performed by the machine. By co-locating containers on fewer machines, efficiency is maximised.

WHAT IS AZURE KUBERNETES SERVICE (AKS)?

Microsoft provides a complete ecosystem for containerised software deployment through Azure Container Services. For those that are comfortable with Kubernetes for container orchestration, Microsoft offers Azure Kubernetes Services.

Within a single service, organisations can:

- Order infrastructure
- Utilise tools for building and deploying containers
- Fully monitor your applications

Microsoft offers all these functions through a flexible pricing model, where you only pay for what you use.

KUBERNETES + SAS

Like other vendors, SAS has phased out software delivery through CDs and DVDs. Instead, SAS now distributes its software electronically over the network via web-based services that are updated frequently. Receiving updated software is now as easy as replacing one container image with another and then requesting the Kubernetes Cluster to make the change effective.

CONTAINERS IN SAS VIYA 4

Containerised delivery has allowed the SAS institute to move into Continuous Delivery for SAS Viya, with new features released frequently and seamlessly. Customers can make use of the updated versions of the software with almost no downtime during updates.

The diagram below shows how SAS Viya has been organised for containerisation.

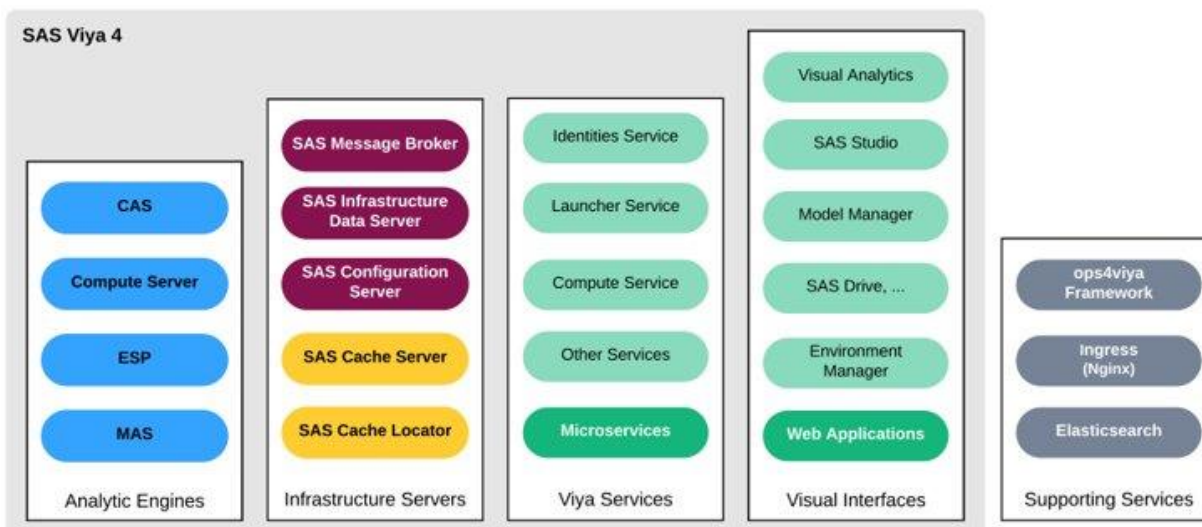


Figure 2 Containers in SAS Viya 4

Important note: Node pools should be comprised of containers (and thus applications) that use similar resources to maximise hardware configurations.

The different applications have been grouped by node pools.

- *Analytics Engines*
- *Infrastructure Servers*
- *Viya Services*
- *Visual Interfaces*
- *Supporting Services*

SAS VIYA 2020.1 WITH KUBERNETES

Below is a diagram of a typical Kubernetes Cluster deployment and the Docker containers within it for SAS Viya 2020.1.

Users can connect with SAS Viya in 3 ways. The most common way to access Viya is through SAS Microservices, using a mobile client, web browser, or an equivalent. An example of this is the user experience of logging on to Visual Analytics. Other users may prefer other SAS clients, such as Enterprise Guide, which accesses SAS Viya through the SAS Compute Server. Lastly, CAS users can directly access the CAS through TCP/IP.

Each of the components is deployed in its own container, making the components portable and easy to upgrade. For instance, upgrading a single microservice C2 is possible by simply swapping out the container image.

Alongside the Microservices, SAS, and CAS clusters, you can also find several support services that log metrics and monitor the health of the containers in the Kubernetes Cluster.

Lastly, RabbitMQ, Postgres, and Consul are also deployed in the same Kubernetes Cluster. All of which can interact with the Microservices. These external applications can be updated just like the other containers in the Kubernetes Cluster.

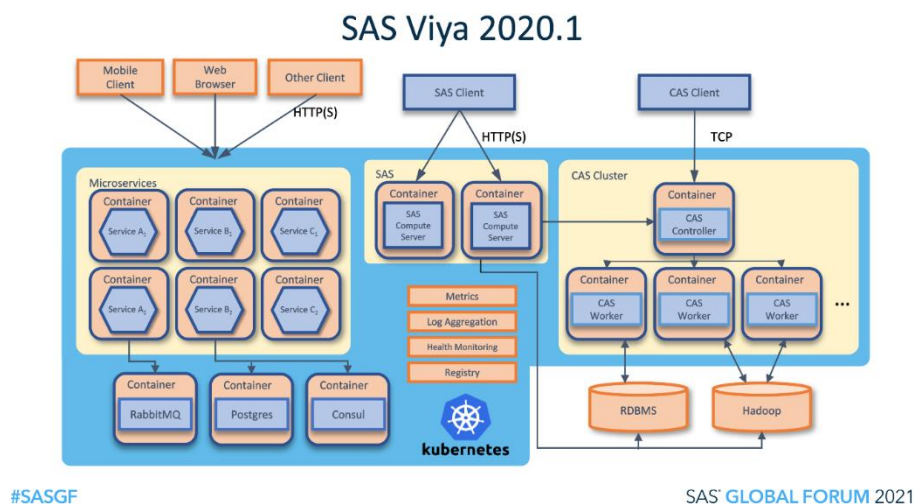


Figure 3 Kubernetes Cluster Architecture for SAS Viya 2020.1

SAS VIYA 2020.1 WITH KUBERNETES ON AZURE AKS

If the scheme presented above were to be deployed on Azure, the architecture could look like this:

SAS Viya 2020.1 on Azure AKS

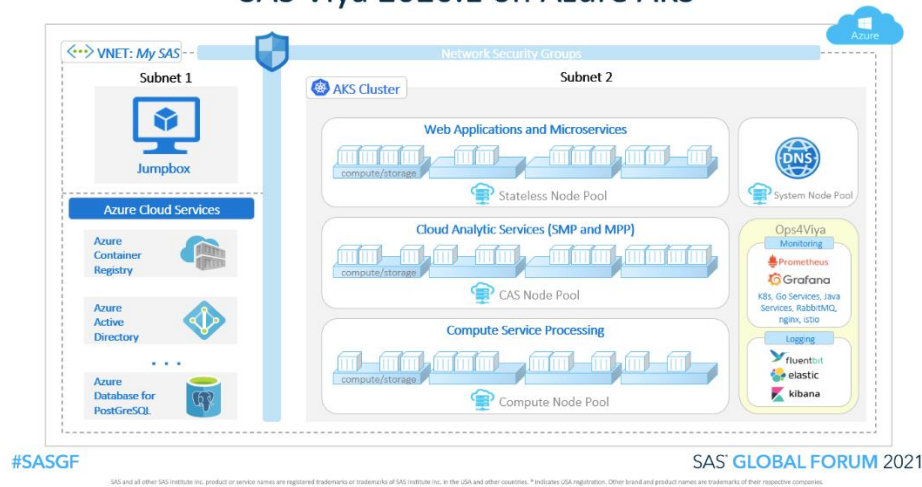


Figure 4 Architecture of SAS Viya 2020.1 on Azure AKS

The Jumpbox or Management server that manages resources and clusters in the cloud can be placed in Subnet 1, along with Azure Cloud Services. Viya can also be configured to incorporate Azure Container Registry, Azure Active Directory for user authentication, and Azure Database for PostgreSQL. Unlike the architecture shown in Figure 3, PostgreSQL is implemented externally as a service. Azure Cloud Services do not require containers because they are available as SaaS. As such, they do not require any deployment. Do note that additional configurations to your subnet may be necessary for your AKS Cluster to interact with your instance of Azure Database for PostgreSQL.

Subnet 2 contains the Kubernetes Cluster with five node pools: Stateless Node Pool, CAS Node Pool, Compute Node Pool, System Node Pool, and Ops4Viya.

Web Applications and Microservices requiring low RAM and CPU access are grouped in the Stateless Node Pool.

Whether you use a Symmetric Multiprocessing (SMP) for single-machine architecture or opt for Massively Parallel Processing (MPP) for a distributed CAS Server, Cloud Analytics Services are grouped in the CAS Node Pool, which requires very high RAM/CPU ratios.

Compute Service Processing lumps together applications that process workloads such as batch processes in the Compute Node Pool that typically require a high-performance CPU and lower CPU to RAM ratio.

System Node Pool includes containers needed to enable Kubernetes to function on Azure.

Finally, Ops4Viya, which consists of logging and monitoring applications. Examples of which are Prometheus, Grafana, elastic, and kibana.

CONCLUSION

Containers are an excellent way to create and deploy software without worrying too much about the underlying infrastructure. It enables organisations to utilise the cloud fully. By using containers, your team can move into a DevOps or Continuous Development model, which rapidly fuels innovation.

Kubernetes is an open-source tool used to orchestrate and monitor containers. It is available on Azure through Azure Kubernetes Service (AKS) and currently the only way to implement containers on SAS Viya 2020.1.

SPECIAL NOTES

Please note this article and the video accompanying it was written for SAS Viya 2020.1. As SAS releases new versions of the software as often as monthly, please be aware that the content could be out of date by the time you read this.

Enterprise Guide can connect to Compute in SAS Viya 3.5, but this is not available for Viya 4.

Ops4Viya has been rebranded as SAS Viya 4 Monitoring for Kubernetes since the release of the video and article.

In the article, I mention five node pools, but in reality, your cluster could have many more for your deployment of Viya and to support other software in the cluster. Future versions of SAS Viya 4 will have Stateful, Stateless, Compute, CAS, MAS and Connect node pulls to better utilise available resources.

SAS Viya 2020.1 uses Docker runtime at the moment of the recording. SAS Viya 2020.1.3 has now released support for other types of container runtimes.

QUESTIONS AND FOLLOW UP

If you have any questions, suggestions or would like to seek out advice, please reach out to me directly using the contact information below. Analytium Group is a SAS and Microsoft certified partner/reseller. We would be pleased to assist you with any queries regarding SAS, Azure, Kubernetes Services and data analytics in general.

REFERENCES

1. Pendergrass, J. (2017). "The Architecture of the SAS® Cloud Analytic Services in SAS® Viya™". Cary, NC: SAS Institute Inc. Available at <https://support.sas.com/resources/papers/proceedings17/SAS0309-2017.pdf>
2. SAS Institute Inc. (2017). "SAS® Micro Analytic Service 5.1: Programming and Administration Guide". Cary, NC: SAS Institute Inc. Available at <https://documentation.sas.com/api/docsets/masag/5.1/content/masag.pdf?locale=en#nameddest=p1dhh0p6zuvf9cn1bxhcstz51ddq>
3. SAS Institute Inc. (2021). "SAS Viya 4: Architecture". Cary, NC: SAS Institute Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. The author can be contacted through the following details:

Name	: Vasilij Nevlev
Enterprise	: Analytium Group
Email	: vasilij.nevlev@analytium.co.uk
Web	: www.analytium.co.uk
Address	: 4 th Floor, 86-90 Paul Street, London, EC2A 4NE, UK
Work Phone	: +44 20 399 49841
Mobile	: +44 77 3761 9734