

## SAS® and Containers: The State of the Union 2019

Paul Kent, SAS Institute Inc., Cary, NC

### ABSTRACT

Containers are a popular new idea for building and deploying applications. At SAS® Global Forum 2018, I introduced a recipe for building a Docker container from a SAS® Viya® order. The container enables you to stand up an editing environment that allows you to perform SAS® programming and experiment with SAS Viya and related SAS® Cloud Analytic Services (CAS) actions from SAS code or from a Jupyter (Python) notebook. This talk presents a progress report—pre-built containers from SAS, examples for different SAS/ACCESS® engines, and more complicated deployment patterns onto Kubernetes.

### INTRODUCTION

Containers are simultaneously a new cool idea, and just another spin on something older. In days gone by, we might have been talking about enabling individual developer (data scientist) productivity with a sandbox environment catering to their needs and preferences. Back then we might have meant a hot-rod, high-spec PC as their desktop. These days we might mean elastic cloud resources, with a wide variety of curated tools available from a menu--no need for each developer to install them (and worry about keeping the versions current) on "their" PC.

At the same time, enterprises have been pushing the boundaries of enterprise scale adoption: one larger system, that many people use/share/collaborate with. Data is prepared once (the rise of the data lake) and shared across projects that find it useful. New data sources are introduced to the data lake for the initial consumer/requester, but more often than not the data that was newly useful for one team/model/domain is also useful to mix in to other projects underway.

Containers help at both ends of this scale. In this talk I discuss a three-step model:

1. Containers to facilitate individual productivity for programming
2. Containers to facilitate individual productivity for programming [on larger problems; on shared data sets]
3. Containers to facilitate a full stack installation of SAS® Viya®

After reading/attending this talk, I hope you'll be able to find resources on github.com that allow you to build and deploy SAS software into containers in your infrastructure.

### WHAT ARE CONTAINERS (AND WHAT IS DOCKER)?

You can google for "what is a container," and you'll find a litany of resources giving their take on this. Wikipedia has a nice succinct definition:

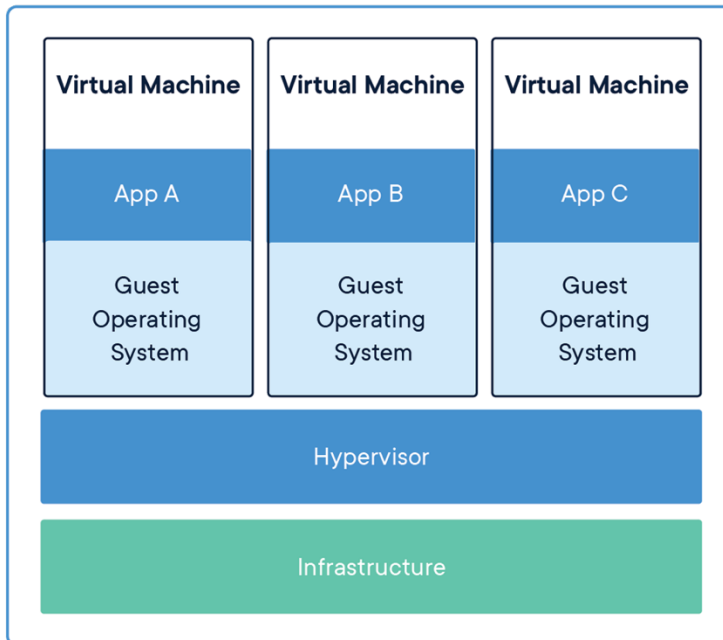
In computing, a container is an isolated user space in which computer programs run directly on the host operating system's kernel but have access to a restricted subset of its resources. A computer program running on an ordinary operating system can see all resources (connected devices, files and folders, network shares, CPU power,

quantifiable hardware capabilities) of that computer. However, programs running inside a container can only see the container's portion of the file system and the devices assigned to it. The mechanism by which a host operating system runs programs in isolated user-space environments is called containerization or operating-system-level virtualization.

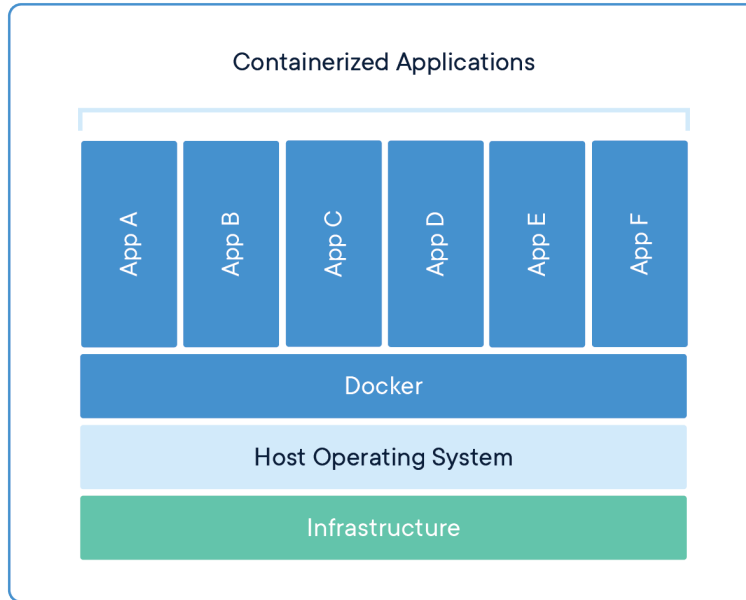
Containers are a progression of the ideas introduced by virtual machines. Docker.com (the company associated with the open-source project that defined containers) has a more concise take:

Containers: A standardized unit of software

Virtual machines allow you to run several operating systems on one server using a hypervisor to mediate and compartmentalize several guest operating systems; each guest can run one or more applications



Using containers is an idea that takes out some of the middle man. No need for several guest operating systems, each with its full suite of libraries and binaries to support its installed applications. The Docker engine runs compartmentalized applications--the compartments (or containers) have only the binaries and libraries they need, and are isolated from each other. This is very helpful when App A needs versions of libraries that conflict with those App B uses.



Containers are very interesting in the era of cloud computing because once you have packaged an application into a container, you can run it in many places without having to make further changes: on your laptop; on a server in your datacenter; in Amazon Cloud; in Azure cloud; in Google Cloud--in fact, most any cloud provider will have a facility to run containers. The metaphor of a standardized shipping container, with well-known dimensions, means that no talk on Docker is complete without a picture of a container ship underway.



## SAS AND CONTAINERS

SAS® software can run in a container. Doug Liming was a force behind helping folks run SAS®9 in containers--often referred to as SAS® Analytics for Containers. The process for building a SAS®9 container was largely to install it on a donor operating system, and then copy the "sashome" folder into the container. De Capite (2018) gives examples for many different cloud providers.

We have improved the approach for doing this with SAS Viya; the container recipes are available to the public on SAS GitHub. In 2018 my talk showed how to build an all-in-one container for a data scientist. This remains as the first building block today.

## SAS® Viya® Container Recipes



Framework to build SAS Viya Docker images and create deployments using Kubernetes.



Container deployments are more lightweight and don't have as much overhead as traditional virtual machines. By running a SAS engine inside a container, you can provision resources more efficiently to address a wide variety of SAS workloads. Select a base SAS recipe and create custom containers with specific products or configurations – e.g, access to data sources, in-database code and scoring accelerators, or specific analytic capabilities. For more information see [SAS for Containers](<http://support.sas.com/rnd/containers/>).

Once you have your SAS order email (the SAS order and license details are wrapped up into a single ZIP file), these steps (from a shell in your home directory) are sufficient to build your licensed SAS software into a container and then run it.

```
$ cd $HOME
$ git clone git@github.com:sassoftware/sas-container-recipes.git
$ ./build.sh --type single \
  --zip ~/path/to/SAS_Viya_deployment_data.zip \
  --addons addons/auth-demo
```

It takes a while (typically 20-30 minutes, but longer when I'm at home on the VPN). When it is finished, it tells you the names of the containers it has built. You'll need this name to run the container, or to promote it to a shared repository/catalog or to a *registry* as Docker people like to call this.

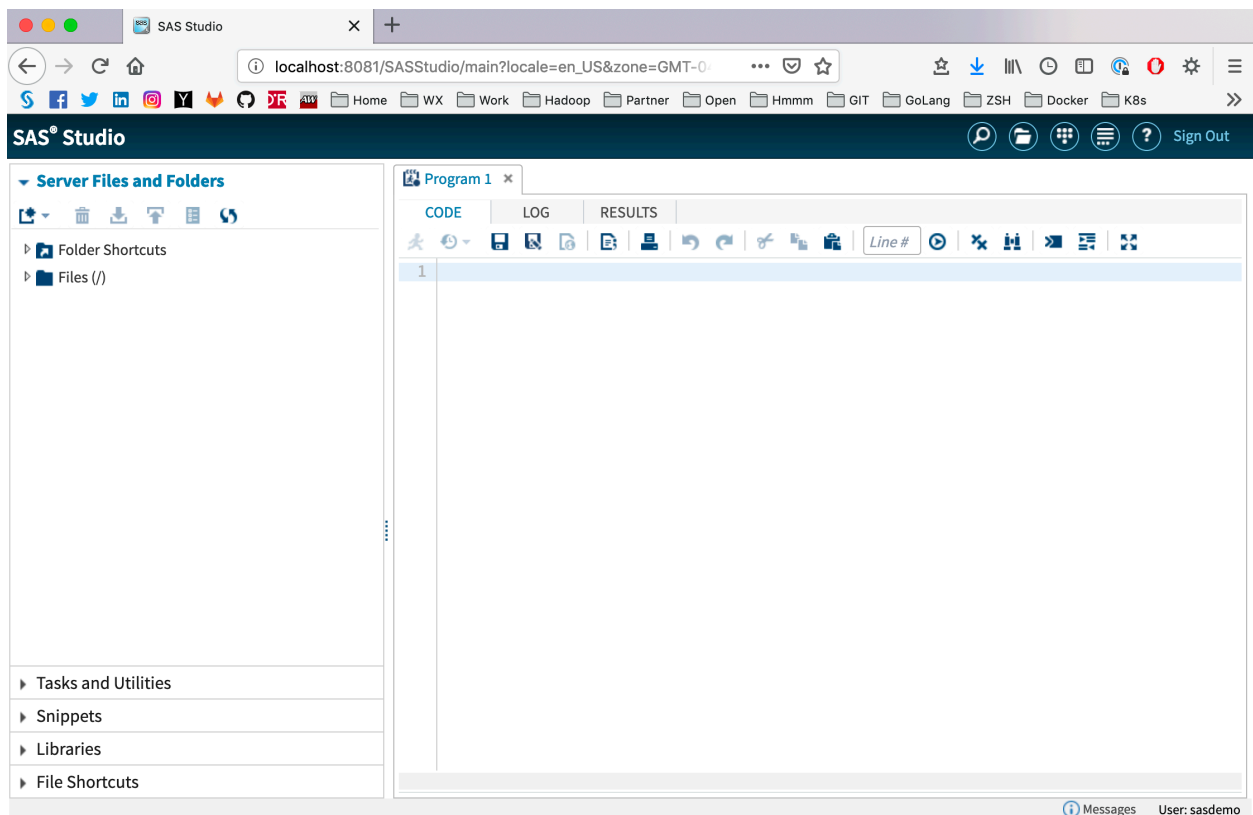
[INFO] : Listing Docker images:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
sas-viya-programming	19.03.0-20190325113734-2e3bf7c	ebd8c0df96b9	Less than a second ago	13.6GB
svc-auth-demo	latest	ebd8c0df96b9	Less than a second ago	13.6GB
viya-single-container	latest	2d33d4be7b01	2 seconds ago	13.6GB
sas-viya-programming	18.11.0-20181026091441-1b90437	fa763dc395f2	5 months ago	14.4GB
<none>	<none>	d8b8103a32cd	5 months ago	13.2GB
394378624171.dkr.ecr.us-east-2.amazonaws.com/kent/reg	latest	1ba683e2ed61	7 months ago	13.1GB
sasawb	build	1ba683e2ed61	7 months ago	13.1GB
<none>	<none>	45bcad48e740	7 months ago	9.55GB
centos	latest	5182e96772bf	7 months ago	200MB
build	latest	56a764cb464e	7 months ago	8.77GB
docker/dtr	latest	dfc50eadd6a7	8 months ago	110MB
docker.sas.com/anbise/sasbase	0.0.2	1a2790e78560	8 months ago	497MB
centos	7	49f7960eb7e4	9 months ago	200MB
hello-world	latest	f2a91732366c	16 months ago	1.85kB

For the 'sas-viya-programming' docker image, you can run the following command to create and start the container:  
docker run --detach --rm --env CASENV\_VIRTUAL\_HOST=HomeMac --env CASENV\_VIRTUAL\_PORT=8081 --publish-all --publish 8081:80 --name <docker container name> --hostname <docker hostname> sas-viya-programming:19.03.0-20190325113734-2e3bf7c

To create and start a container with the 'viya-single-container' image and no addons, submit:  
docker run --detach --rm --env CASENV\_VIRTUAL\_HOST=HomeMac --env CASENV\_VIRTUAL\_PORT=8081 --publish-all --publish 8081:80 --name sas-viya-programming --hostname sas.viya.programming viya-single-container

I can now run a SAS® Studio session using the “Docker run command” suggested by the build process, and visit the application at “server:port/SASStudio” – for example on localhost as shown below.



You probably want to go and explore the addons folder some more; there are examples for adding different SAS/ACCESS engines to your container, and for connecting the username/password prompting to a corporate resource like Active Directory.

## ORCHESTRATION (AND KUBERNETES)

The container we built above is limited to the resources available to one Docker engine—granted, this can be a nice beefy Linux server with 1TB of RAM. If your problem-size starts to outpace a single server, you can explore Step 2 in the progression of sas-container-recipes where you build and use several containers. There’s one for the SAS Studio application, one for a CAS controller, and several for CAS workers. Now your SAS problem solving can scale out over several Docker engines, and access the resources of more than one server.

Once you have more than one container, you start to need a way to launch each of the containers that makes up your application, and to manage the interrelationships between the containers. What if one of them dies? Can something monitor that, and restart as needed? These tasks are often labeled collectively as “Orchestration.” Enter Kubernetes—a distributed operating system for containers.

## SAS AND KUBERNETES

```
$ ./build.sh --type multiple \  
--zip ~/path/to/SAS_Viya_deployment_data.zip \  
--docker-url {your-docker-registry} \  
--docker-namespace {your-namespace-in-registry} \  
--virtual-host {your-application-ingress-url} \  
--addons addons/auth-demo
```

This command builds a set of containers and pushes them to a Docker registry. (When the application is scheduled across several hosts they need somewhere to “fetch” the container images.)

```
[INFO] : Listing Docker images where "label=sas.recipe.version=19.03.0" :
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
docker.sas.com/kent/sas-viya-sas-casserver-primary	19.03.0-20190325125523-2e3bf7c	1783f1c557ff	2 seconds ago	4.17GB
svc-auth-demo	latest	1783f1c557ff	2 seconds ago	4.17GB
docker.sas.com/kent/sas-viya-programming	19.03.0-20190325125523-2e3bf7c	0703621fc0ed	8 seconds ago	9.29GB
docker.sas.com/kent/sas-viya-sas-casserver-primary	19.03.0-20190325125523-2e3bf7c-base	35a21dfc2b5a	6 minutes ago	4.17GB
sas-viya-sas-casserver-primary	20190325171331	35a21dfc2b5a	6 minutes ago	4.17GB
sas-viya-sas-casserver-primary	latest	35a21dfc2b5a	6 minutes ago	4.17GB
docker.sas.com/kent/sas-viya-programming	19.03.0-20190325125523-2e3bf7c-base	04fa3a7354a1	9 minutes ago	9.29GB
sas-viya-programming	20190325171010	04fa3a7354a1	9 minutes ago	9.29GB
sas-viya-programming	latest	04fa3a7354a1	9 minutes ago	9.29GB
docker.sas.com/kent/sas-viya-httpproxy	19.03.0-20190325125523-2e3bf7c	bc74dffe123b	21 minutes ago	279MB
docker.sas.com/kent/sas-viya-httpproxy	19.03.0-20190325125523-2e3bf7c-base	bc74dffe123b	21 minutes ago	279MB
sas-viya-httpproxy	20190325165844	bc74dffe123b	21 minutes ago	279MB
sas-viya-httpproxy	latest	bc74dffe123b	21 minutes ago	279MB

Kubernetes manifests can be found at [viya-programming/viya-multi-container/manifests/kubernetes](#).

One should review the settings of the yaml files in [viya-programming/viya-multi-container/manifests/kubernetes/configmaps](#) to make sure they contain the desired configuration

To deploy the SAS Viya environment, run the following

```
kubectl apply -f viya-programming/viya-multi-container/working/manifests/kubernetes/namespace  
kubectl -n sas-viya apply -f viya-programming/viya-multi-container/working/manifests/kubernetes/ingress  
kubectl -n sas-viya apply -f viya-programming/viya-multi-container/working/manifests/kubernetes/configmaps  
kubectl -n sas-viya apply -f viya-programming/viya-multi-container/working/manifests/kubernetes/secrets  
kubectl -n sas-viya apply -f viya-programming/viya-multi-container/working/manifests/kubernetes/services  
kubectl -n sas-viya apply -f viya-programming/viya-multi-container/working/manifests/kubernetes/deployments
```

You deploy the application to Kubernetes by applying the generated manifests. This makes Kubernetes aware of the resources and configuration required for the application, and the Kubernetes scheduler arranges to satisfy those requirements.

The `--type multiple` argument to the `build.sh` command builds a sas programming-only environment. As you gain experience with Kubernetes, you might want to experiment with a `--type full` build. This will construct containers and manifests for all the micro services in a full stack SAS Viya deployment. The `--type full` argument is available for your experimentation, and to learn how a large application suite like SAS can be deployed. At the time of writing (March 2019) this third `--type full` variant is not yet considered "ready for production deployment," and the build process will warn you with words to that effect.

## REFERENCES

Cloud Native Computing Foundation. What is Kubernetes?  
<https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/>

"Container (virtualization)." Wikipedia.  
[https://en.wikipedia.org/wiki/Container\\_\(virtualization\)](https://en.wikipedia.org/wiki/Container_(virtualization))

De Capite, Donna. 2018. Docker Toolkit for Data Scientists – How to Start Doing Data Science in Minutes! *Proceedings of the SAS Global Forum 2018 Conference*. Cary, NC: SAS Institute Inc. <https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/1875-2018.pdf>

Docker, Inc. What is a Container? <https://www.docker.com/resources/what-container>

SAS Institute Inc. SAS Software Open Source Project called "sas-container-recipes"  
<https://github.com/sassoftware/sas-container-recipes>

## RECOMMENDED READING

- Docker.com – What is a Container?
  - <https://www.docker.com/resources/what-container>
- Github.com – the sas-container-recipes project
  - <https://github.com/sassoftware/sas-container-recipes>
  - <https://github.com/sassoftware/sas-container-recipes/wiki>
- Kubernetes – What is Kubernetes?
  - <https://kubernetes.io/docs/concepts/overview/what-is-kubernetes>

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

[Paul Kent](#)

100 SAS Campus Drive  
Cary, NC 27513  
[Paul.Kent@sas.com](mailto:Paul.Kent@sas.com)  
<http://www.sas.com>  
[twitter.com@hornpolish](https://twitter.com/hornpolish)  
<https://github.com/sassoftware/sas-container-recipes/issues>

Note: sas-container-recipes is an open-source project. Contributions from everybody are how open-source projects improve - we welcome yours - our [contributions](#) page has details. If you are using SAS and containers together, I would love to learn from your experience. Please do get in touch!

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.