Docker and Kubernetes

I. **Docker:**
   1. What is Docker?
   2. Why use Docker?
   3. When is using Docker necessary?
   4. Containers VS Virtual machines
   5. How are containers created?

II. **Kubernetes:**
   1. What is Kubernetes?
   2. Why use Kubernetes?
   3. When to use Kubernetes?

III. **The Unix team's achievements:**
   1. CAS migration to AKS
   2. Bioinformatics apps containerization
   3. Solaris web content migration to AKS

IV. **Useful links and online resources:**

I. Docker:

1. What is Docker?

Docker is an open-source platform used to develop and run encapsulated application within containers.

2. Why use Docker?

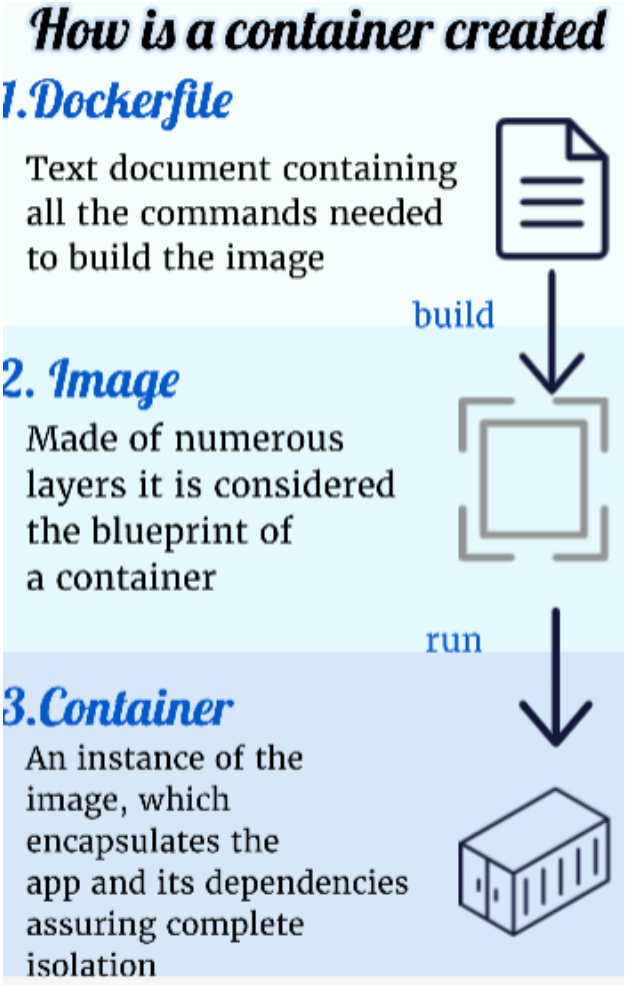
Docker containers are appealing for many reasons, but mainly it's because they are:

- Portable
- Flexible
- Sharable
- OS-independent
- Fast

3. How to use docker?

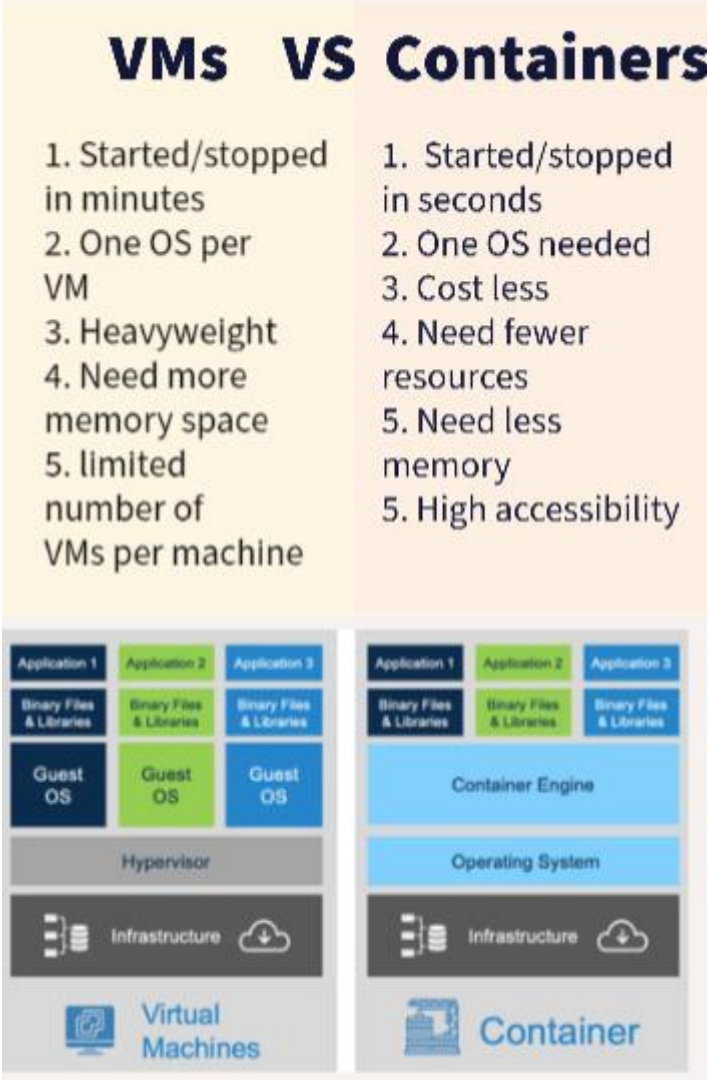
Using docker is quick and simple . After creating a dockerfile and building the image , an instance of that image (container) can then be run.

4. How are containers created?



5. Containers VS Virtual machines :

While containers can be seen as type of virtual machine, the way they operate is distinct as can be seen on the figure below.



II. Kubernetes:

1. What is Kubernetes?

Kubernetes is an open-source container orchestrater . In other words, it deploys , runs, and manages containers in the cloud and on-premises. Assuring that the number of resources needed is always satisfied no matter the demand .

2. Why use Kubernetes?

Kubernetes has some interesting advantages which play a massive role in its constantly growing popularity. Naming a few:

- Automation and load balancing
- Scalability
- High availability
- Portability
- Security

3. When to use Kubernetes?

Kubernetes can be used to manage any container , if required but it is usually used for :

- Applications with high traffic
- Big data analytics
- Multiservice applications

III. The Unix team's achievements:

1. CAS migration to AKS:

The central authentication service (CAS) is an open-source software UWE uses to enable single sign-on for its web apps. Earlier this year J.Wilde and T.Elliott were able to migrate CAS to Azure Kubernetes Services (AKS) . The lengthy and complex procedure of containerizing the software with all of it dependencies and deploying it in Azure , helped reduce resource utilization massively. Transitioning from using 2 VMs and needing 2 CPUs as well as 4 GB of memory to having 2 containers and approximatively 0.03 CPUs and 1.2 GB of memory needed per container.

if CAS ever needs more resources , Kubernetes can scale up and create 3 additional containers to meet the demand when traffic is high . One of AKS benefits is its ability to auto-scale ensuring that the minimum resource usage for a specific workload is employed at all times .

2. Bioinformatics apps containerization:

This project was a collaboration between the Unix team and one of UWE's Molecular Biology senior lecturers D.Turner . S.Tadlaoui and J.Wilde were able to containerize 32 Bioinformatics related applications using docker. The initiative will offer future students enrolled in that class an unlimited access to the containerized apps from any machine they wish to use by simply pulling the required image from the Bioinformatics GitHub repository .

While building the different containers images was not easy , the team took their time and rigorously researched the various dependencies to include in the dockerfile of each package built .

3. Solaris web content migration to AKS:

Due to the Solaris operating system being no longer supported and posing grave security concerns. The Unix team started migrating the content part of the WWW Solaris legacy environment to another environment hosted in the Azure cloud . Content such as ./enews, ./logincomms and ./tour-360 will now be accessed by their owners via the Azure storage explorer .

IV. Useful links and online courses:

1. Docker: Accelerated, Containerized Application Development
2. Kubernetes
3. CTOs' Guide to Containers and Kubernetes — Answering the Top 10 FAQs (gartner.com)
4. Learn Docker Despite Your Busy DevOps/Developer Schedule | Udemy
5. Docker for the Absolute Beginner - Hands On - DevOps | Udemy
6. Kubernetes Courses - Get your Kubernetes Certification!
7. Getting Started with Docker | Pluralsight
8. Online Kubernetes Course: Beginners Guide to Kubernetes | Pluralsight