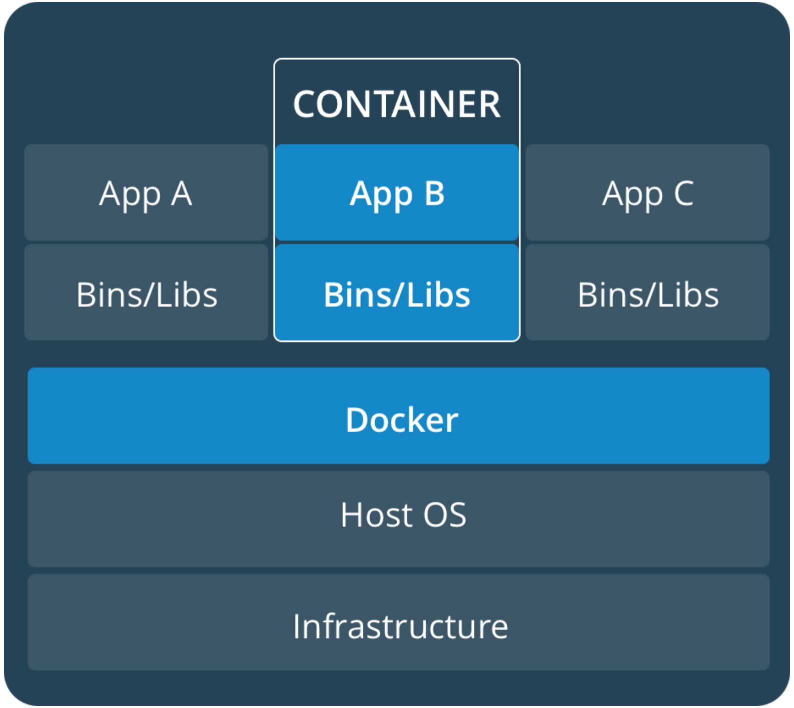
Docker

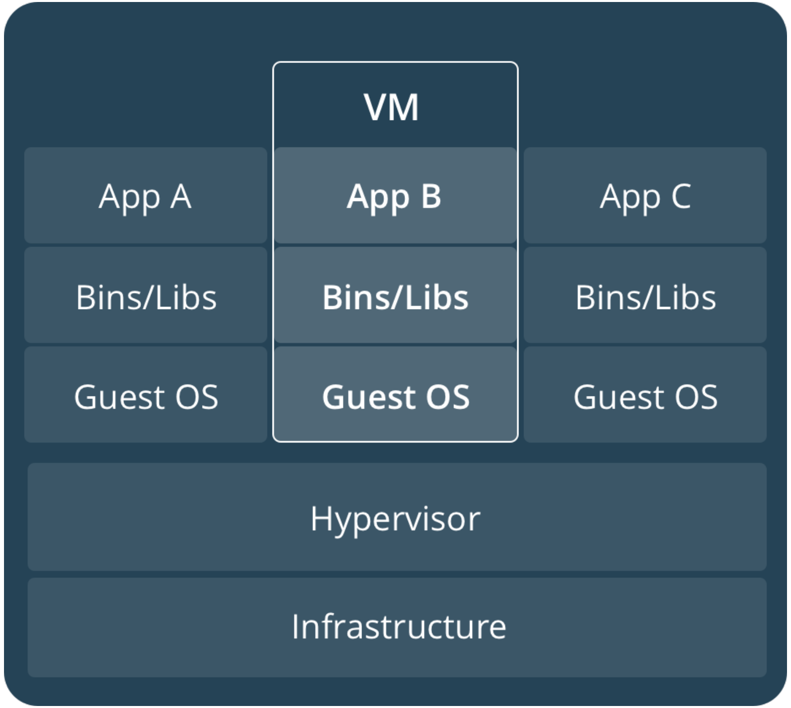
Enterprise Architectures for Big Data

Container vs. Virtual Machine

Container



Virtual Machine



Source: https://docs.docker.com/get-started/

Container vs. Virtual Machine

Container

- Lightweight
- Shares the kernel
- Runs natively on Linux
- Runs as a discrete process
- Fast startup time

Virtual Machine

- Emulates “guest” operating system with virtual access to host resources through a hypervisor
- Slow startup time
- Resource hungry

Source: https://docs.docker.com/get-started/

Docker

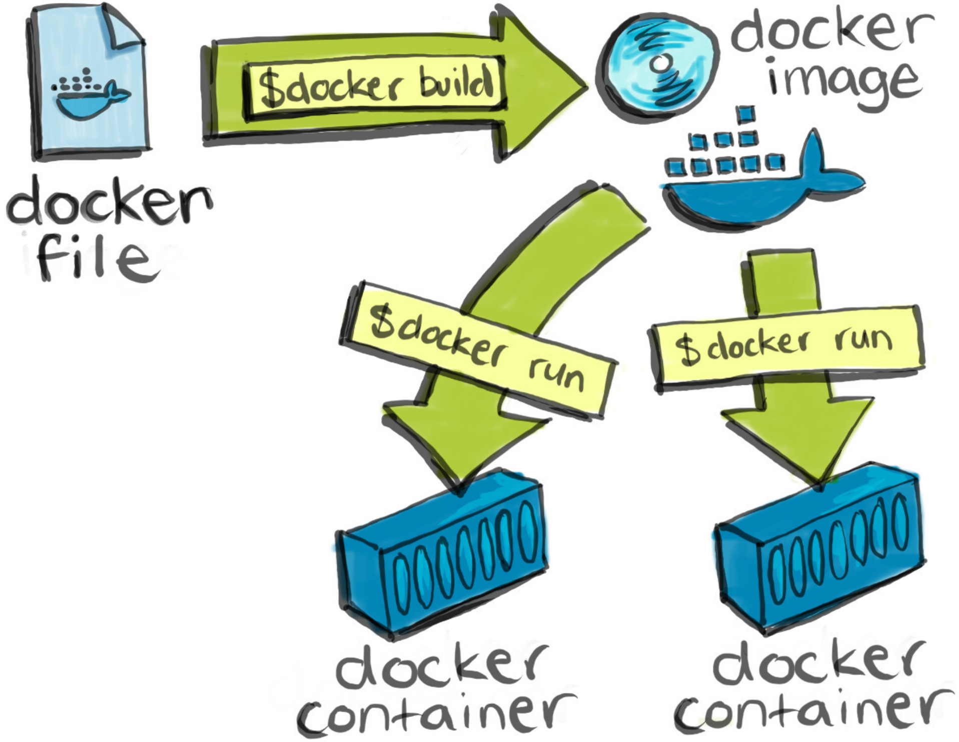
- Docker is the most popular container technology

- Advantage:
  - Reproducibility of your environment
  - Same environment
  - Avoids “It worked on my machine” syndrome
  - Infrastructure as Code
  - Supports Microservices
  - Easy to deploy to Clusters or to the Cloud

Images and Container

- **Image**: Blueprint of an application which forms the basis of a container
- **Container**: Runtime instance of a Docker images. Runs the actual application
- **Docker Hub**: Central Repository of Docker Images
- **Dockerfile:** Text document containing all the commands you would normally execute manually in order to build a Docker image. Docker can build images automatically by reading the instructions from a Dockerfile.
- **Base Image**: An image that has no parent (typical an OS like Ubuntu)
- **Child Image**: Build on a base image and add additional functionality
- **Official image**: Docker sanctioned image on Docker Hub. Has no prefix by an organization or user name (ubuntu, python, ...)
- **User image**: Images from users on Docker Hub. Has user prefix, in the form *user-name/image-name*

Relationships between Dockerfile, Image and Container



Source: https://cultivatehq.com/posts/docker/

Terms

- Docker Daemon
  - (Daemon = Disk And Execution Monitor)
  - Background Docker Process

- Volume:
  - Persists data.
  - Independent of the container’s life cycle.
  - Docker run creates a new container, without the data of a past container (clean slate).
  - If you delete old containers all data will be deleted as well.
  - To persist data (like database data or results of calculation) use volumes.

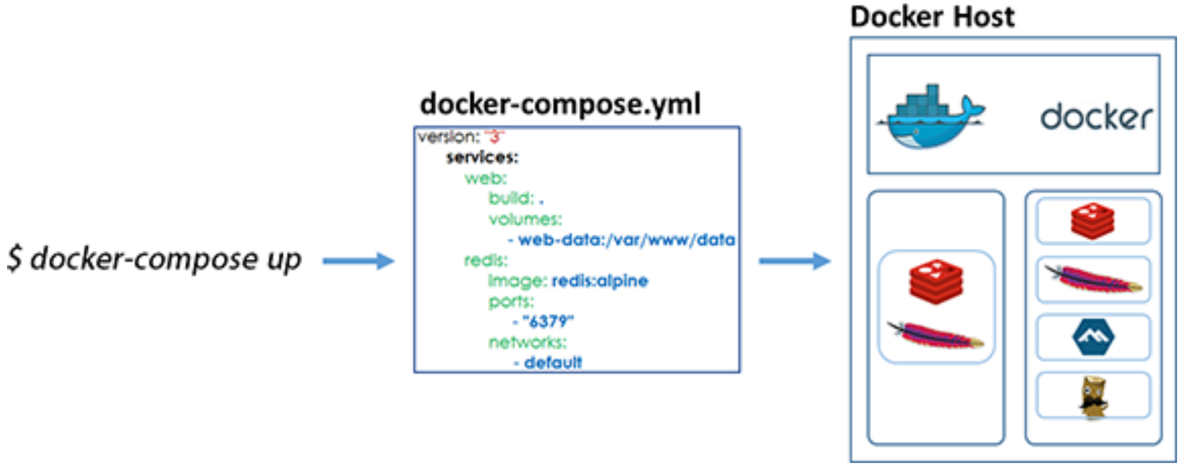
Services & Stacks

- **Service**
  - “container in production“.
  - Codifies the way that image runs (ports, volumes, ...)

- **Stack**
  - Multiple services (containers) that define an application.
  - E.g. one container with PHP Wordpress Blog and one container with the MySQL Database

Docker Compose

- **Docker Compose**: Tool for assembling services consisting of multiple containers (**Stack**). Can spin up the application in a single command which does everything that needs to be done to get it running.
- **docker-compose.yml**: YAML file with interactions of multiple Docker containers. Defines a multi-container application in a single file.



```
$ docker-compose up
```

```
version: "3"
    services:
      web:
        build: .
        volumes:
          - web-data:/var/www/data
      redis:
        image: redis:alpine
        ports:
          - "6379"
        networks:
          - default
```

Source: https://docs.docker.com/glossary/ http://neokobo.blogspot.com/2017/04/docker-compose.html

Container Orchestration

- Manage stacks of containers in production in computer clusters and in the cloud
- Different Tools:
  - Docker Machine
  - Docker Swarm
  - Docker Cloud
  - Kubernetes (K8)
  - Google Kubernetes Engine (GKE)
  - Amazon Elastic Kubernetes Service (Amazon EKS)
  - Azure Kubernetes Service (AKS)
  - Amazon Elastic Container Service (Amazon ECS)
  - Apache Mesos

Function as a Service (FAAS) aka “Serverless”

- Container will be sleeping, if there is not request
- Only pay per request
- Pro:
  - Scalable
  - Ideal for varying workload or experiments
- Contra:
  - Small startup time for waking up the container

- Cloud Offerings:
  - AWS Lambda
  - Azure Functions
  - Google Cloud Run

Deployment of Machine Learning Models

In Development
(on Data Scientist’s Machine):

- Train and Test Machine Learning Model (e.g. with Jupyter Notebook)

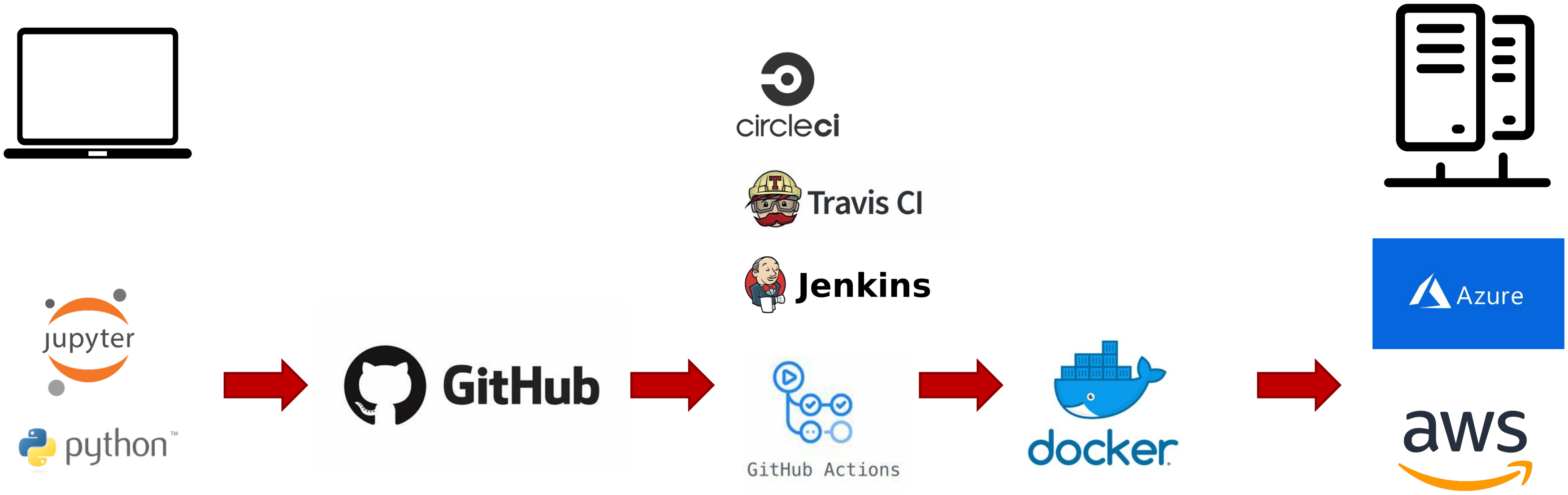
After Development
(on Data Scientist’s Machine):

- Save Machine Learning Model (e.g. with Python Pickle)
- Create a Docker Container
  - Saved Model
  - Flask (Python Web Framework)
  - Expose Model via an API (Application Programming Interface) that communicates JSON (JavaScript Object Notation)

In Production:

- Deploy Container to:
  - Internal Server
  - Internal Cluster
  - Cloud
- Constantly measuring the effectiveness of the model in production

Machine Learning DevOps Pipeline



Development

Shared Repo

Continues Integration (CI)

Continues Deployment (CD)