Bigdatastack.eu

@bigdatastackeu

BigDataStack

Holistic stack for big data applications and operations

# Application and Data Services

**Dr. Richard McCreadie**

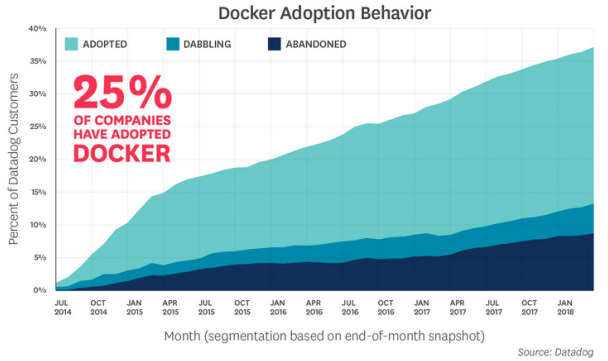
University of Glasgow

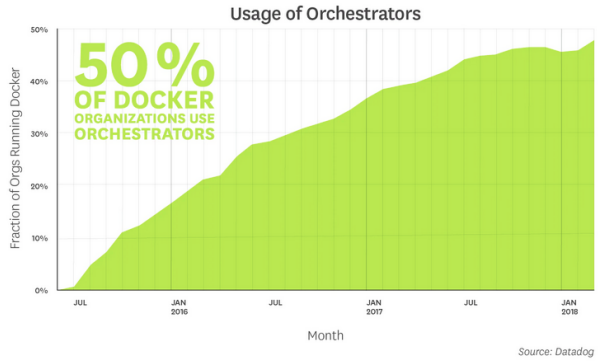
Co-funded by the European Commission
Horizon 2020 - Grant # 779747

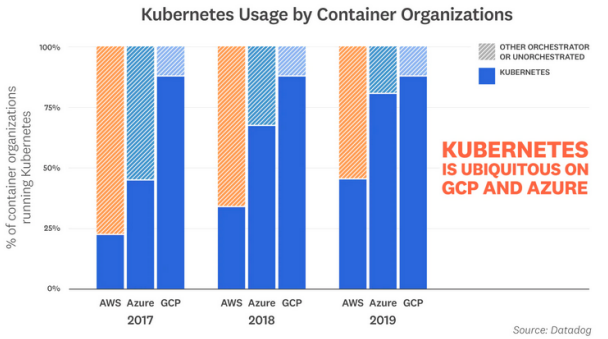
European Commission

# Background in Container Orchestration

- Current best practices in industry state that application deployment should be managed using containers where possible
  - Provide a standard format for all compute
  - Sharable and many turn-key solutions already exist for common applications
- In cloud/cluster environments, an additional container management platform is needed
  - Schedule container execution on available resources
  - Enable communication between different containers
  - Manage storage
  - Enable application scaling and load balancing
- Kubernetes is currently the most popular of these with ~45% market share



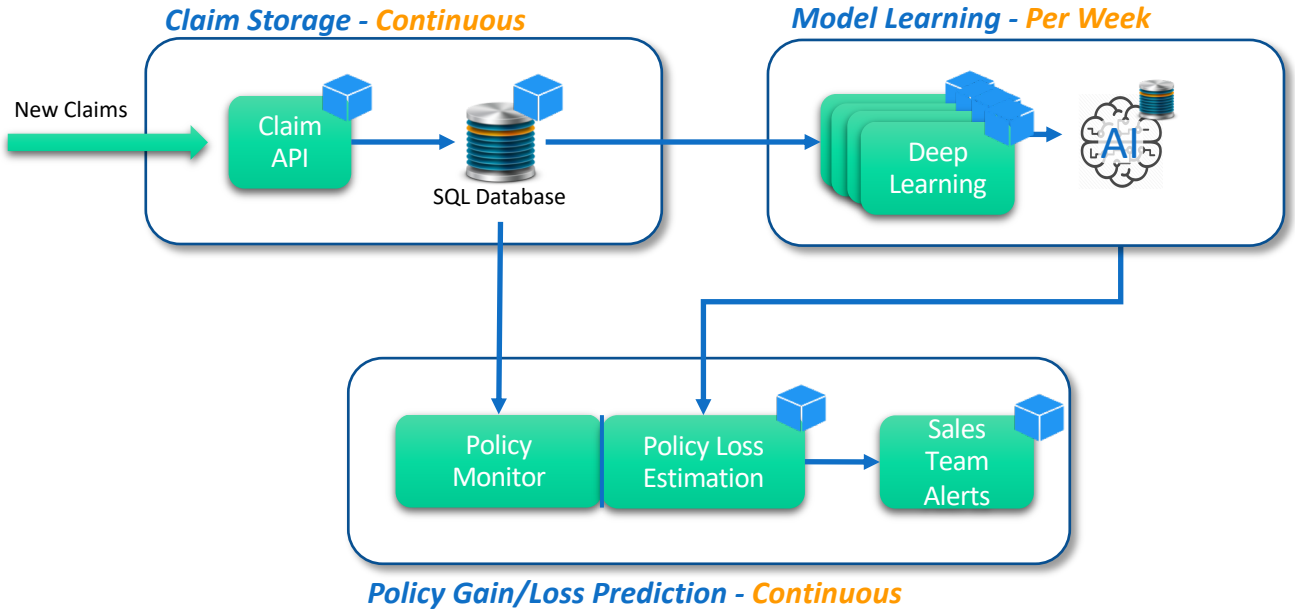




https://www.datadoghq.com/docker-adoption/
https://www.datadoghq.com/container-report/

# Example Application: Insurance Loss Estimation

- **Aim**: For a customer with an insurance policy, and a number of past claims, predict the monetary gain/loss to the insurance company at the end of the policy lifetime
  - Gain/Loss is calculated at the end of each policy year using a deep neural network
  - The deep neural network needs to be updated with new claim information at the end of each week



# Knowledge and Technology Gap

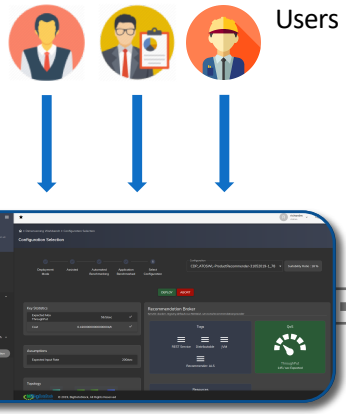
- Platforms like Kubernetes do not provide effective tools to deploy and manage complex applications
  - No central management of multi-component applications
  - No in-built sequencing of operations within an application
  - No identification of the correct amount of resources to assign to application components
  - No monitoring application/container-level Metrics/Quality of Service

# Application and Data Services

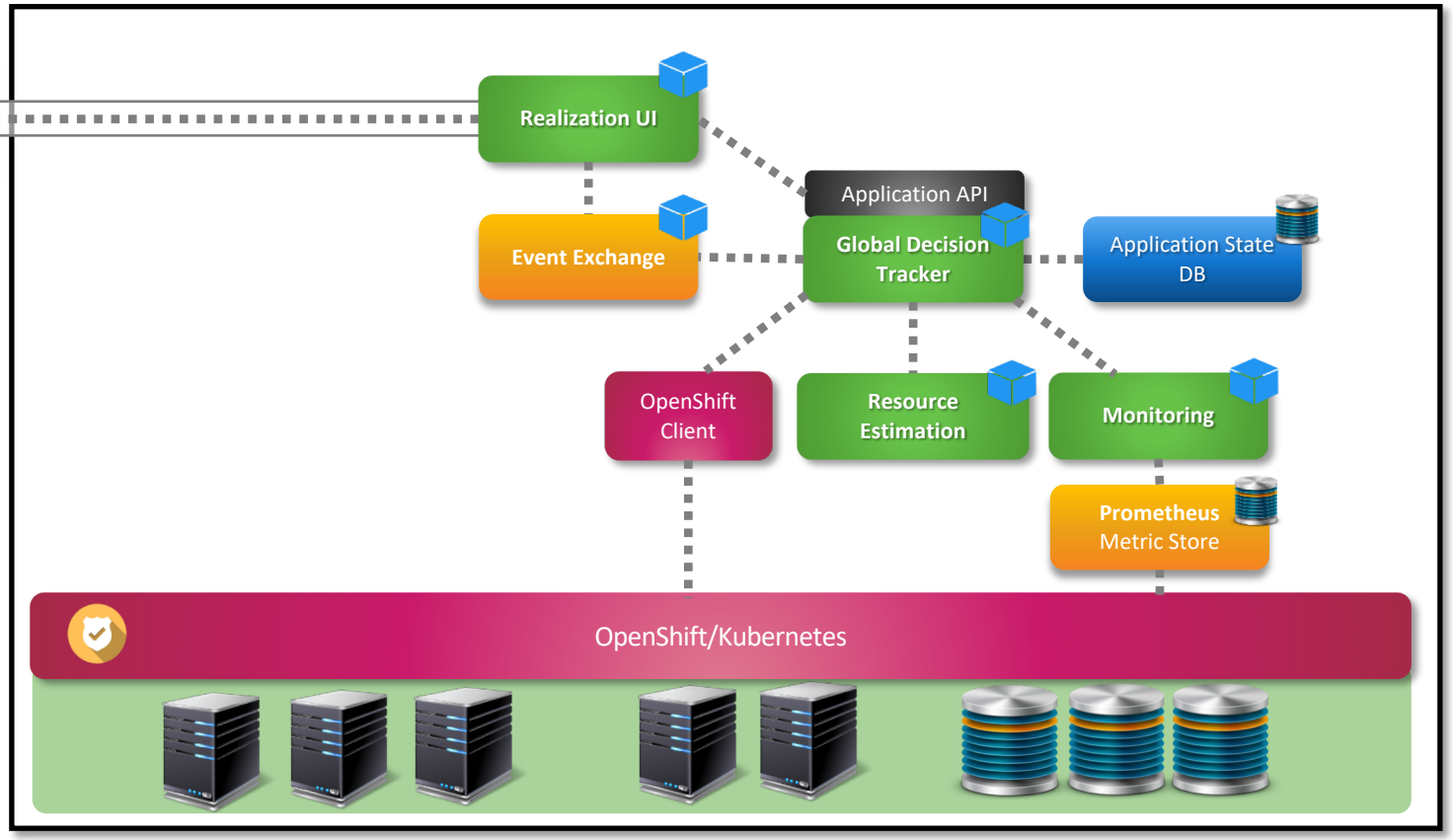
- The BigDataStack Application and Data Services aim to provide an additional tool-set to make tackling these use-cases faster and easier

- Grouping of Kubernetes Components into Applications via BigDataStack Playbooks
- Support for re-usable Operation Sequences with common operations such as Apply, Wait-For, Execute-Command
- Standardized metric collection, storage and visualisation
- Semi-automated resource estimation for containers
- Application API providing direct access to both application-level metrics as well as available operations/sequences

# Architecture

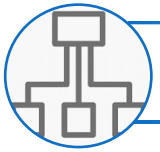



Application and Data Services

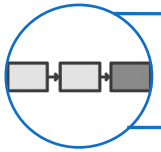


# Example Application: Insurance Loss Estimation

## Advantages:



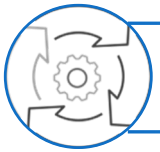
All three components and their desired processing properties can be defined within a single BigDataStack Playbook, then deployed and managed automatically



Operation Sequences can be used to order component deployment dependencies, e.g. the database needs to be running an in 'ready' state before launching model learning


Standardized metric collection enables both component up-time and model performance to be monitored and visualised, in addition to be used as a trigger

During container deployment, resource estimation will set CPU, Memory and GPU requests automatically, and can learn over time how to avoid wasting resources



Application API can be used to enable custom business logic services to control the application at run-time (e.g. scale up/down or trigger model learning)

# Open Source Release

- The current build is currently in alpha testing for product recommendation and insurance use-cases within BigDataStack

- Planned open source release by the **end of the year**