DEPLOYMENT ORCHESTRATION

POKÉMON GO USE-CASE

- Pokemon Go is an augmented reality game developed by Niantic for Android & iOS devices
  - ”People are healthier when they go outside and have a reason to be connected to others.”, Edward Wu, Director of Software Engineering at Niantic Labs

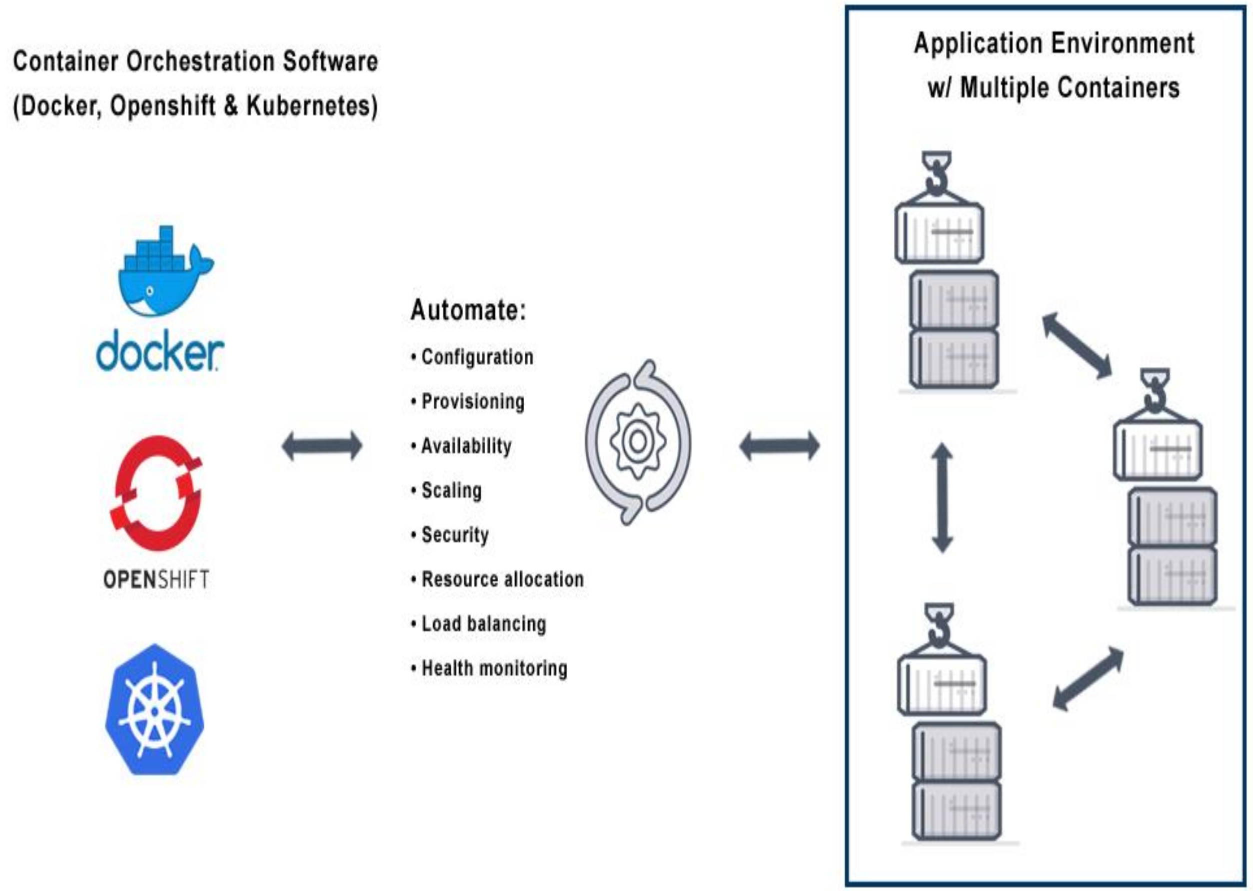
- 500+ million downloads, 20+ million daily active users
- Inspired users to walk over 5.4 billion miles in a year

SCALING UP CONTAINER DEPLOYMENT

- Container – reliable, fast efficient, light-weight
- Easy to instantiate many containers
- But difficult to manage
  - They need to talk to each other – they need networking
  - They need to be deployed appropriately – at the right place
  - They need to be managed carefully – IP address are setup and accessible
  - They need to be able to scale up and down based on demands – auto scaling
  - They need to be able to detect a crash and restart new one if needed
  - Traffic distribution is challenging

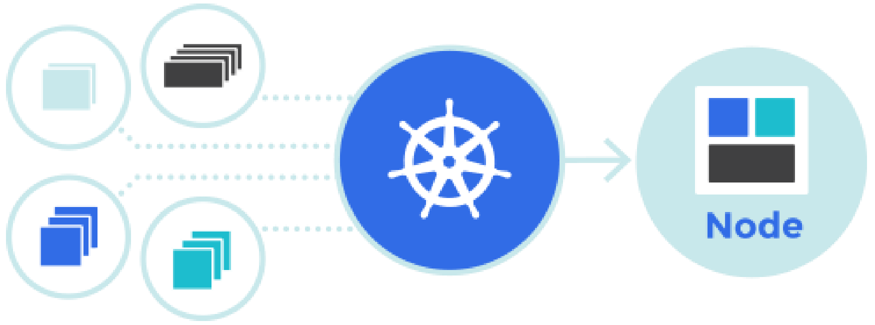
CONTAINER ORCHESTRATION

- Container Orchestration is the automatic process of managing or scheduling the work of individual containers for applications based on microservices within multiple clusters.
- Orchestration Tools:
  - Kubenestes
  - Docker Swarm
  - Appache Mesos
  - CoreOS rkt



KUBERNETES

- An open-source container management tool which automates container deployment, container (de)scaling, container load balancing
- Written on Golang, it has a huge community support
  - It was first developed by Google and later donated to Cloud Native Computing Foundation (CNCF)
- Works with most cloud providers
- Can group 'n' number of containers into one logical unit called '**POD**',
  - easy to manage and deploy



FEATURES OF KUBERNETES

- Automatic Bin-packing
  - Package software and place to container based on resource requirement
- Service Discovery and Load Balancing
  - Auto network and load balance configuration
- Storage Orchestration
  - Auto mount to different storages for the cluster
- Self Healing
  - Detect the crash and restart containers automatically
- Secret and Configuration management
- Batch Execution
- Horizontal Scaling
  - Through command line, dashboard, or autoscaling
- Automatic Rollbacks and Rollouts
  - Make sure an update or rollback would not disrupt the ongoing traffic

POKEMON GO – SCALING UP ARCHITECTURE

A container for Pokémon Go

- Java
- Google BigTable
- MapReduce
- Cloud DataFlow

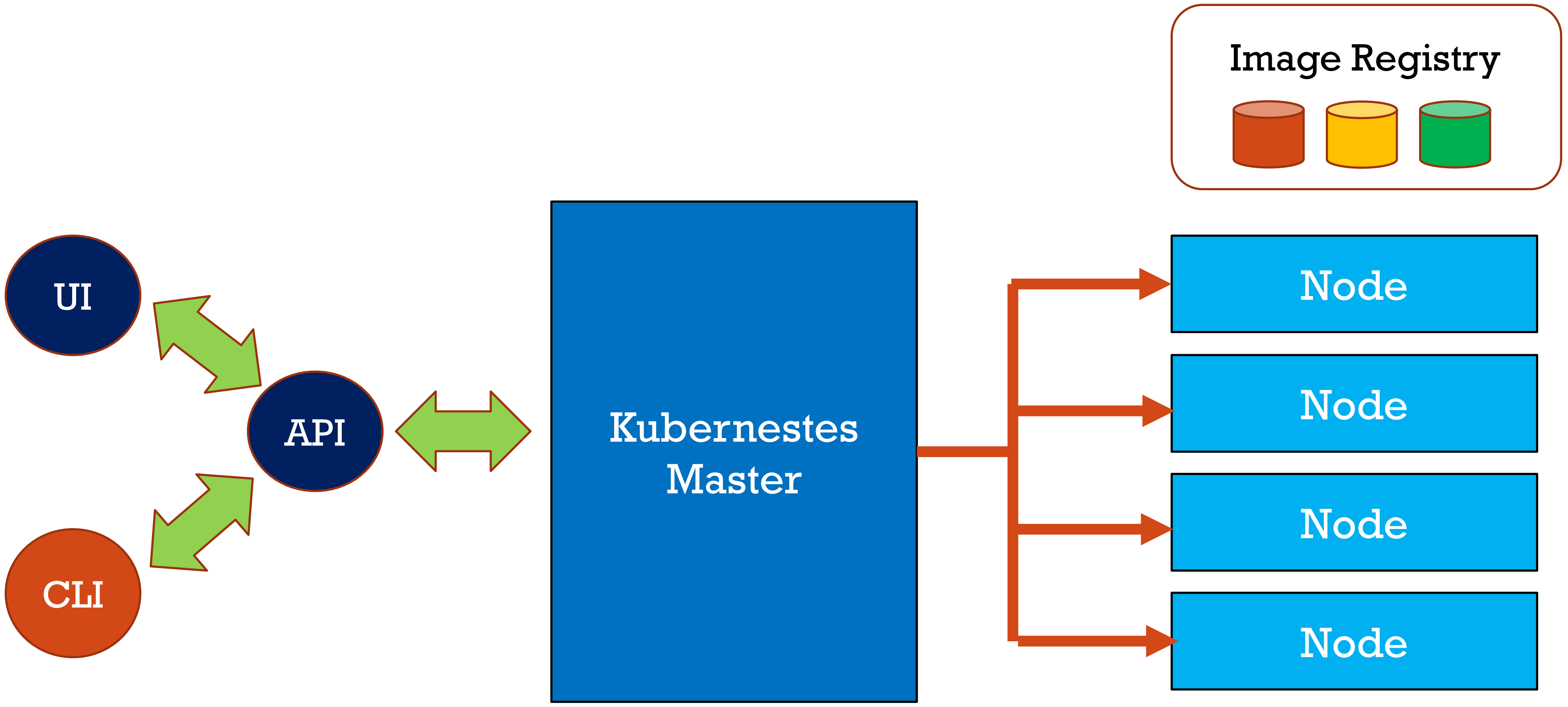
- Tested in one country – Australia and New Zealand
- Initially started with 5 containers
- Crashed because it could not handle the load

POKEMON GO
SCALING UP ARCHITECTURE WITH KUBERNETES

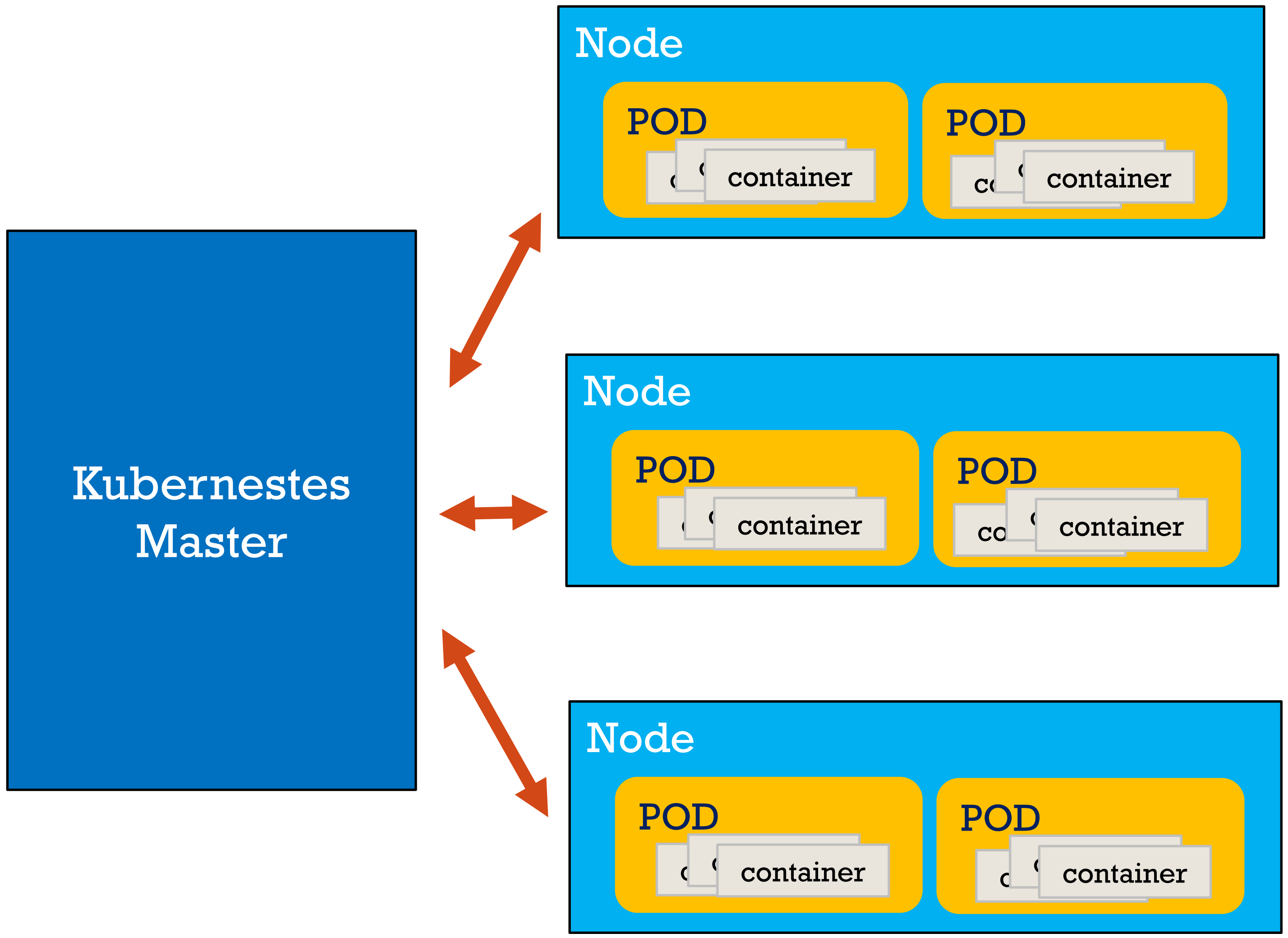
- Java
- Google BigTable
- MapReduce
- Cloud DataFlow

- Challenges
  - Need both horizontal and vertical scaling because of real-time activity in gaming
  - Much higher load than expected
  - Need 50X containers at come point
- Kubernest team at GCP worked with Niantic team to handle all the challenges

KUBERNETES ARCHITECTURE



KUBERNETES OVERVIEW



- Master controls the clusters and the nodes in it
- Nodes host the group of containers called POD
    - Containers in a POD run on the same node and share resources such as filesystems, kernel namespaces, and an IP address.
- Replication Controller at the Master ensure that requested number of PODs are running on nodes
- Load Balancer at the master provide load balancing across a replicated group of PODs

MORE ABOUT KUBERNETES

- Required Readings: https://kubernetes.io/docs/tutorials/kubernetes-basics/
- Optional Readings:
  - Kubernetes YAML Generators: https://k8syaml.com/
  - Best Practices: https://github.com/diegolnasc/kubernetes-best-practices?utm_source=tldrnewsletter
  - Orchestration Platform Comparison: https://newrelic.com/blog/best-practices/best-cloud-infrastructure-automation-tools
  - k0s - Zero Friction Kubernetes: https://github.com/k0sproject/k0s