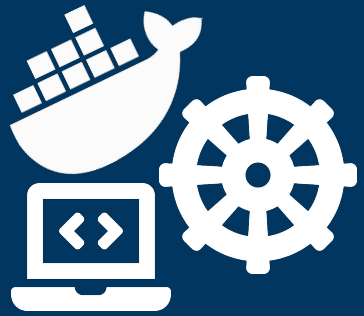
Runtime Auto Deploy (RAD)

Aarti Jivrajani
Daniel Shu



Department of Computer Science

UC SANTA BARBARA

```
  (Total limits may be over 100 percent, i.e., overcommitted.)
  CPU Requests  CPU Limits     Memory Requests  Memory Limits
  ------------  ----------     ---------------  -------------
  7910m (98%)   20100m (251%)  10736Mi (33%)    22762Mi (71%)
Events:
  Type     Reason                   Age                From                      Message
  ----     ------                   ----               ----                      -------
  Normal   NodeNotReady             7h (x14 over 5d)   kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeNotReady
  Warning  ContainerGCFailed        7h (x6 over 4d)    kubelet, 20.20.208.6      rpc error: code = DeadlineExceeded desc = context deadline exceeded
  Normal   NodeHasNoDiskPressure    7h (x40 over 5d)   kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeHasNoDiskPressure
  Normal   NodeReady                7h (x33 over 5d)   kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeReady
  Normal   NodeHasSufficientMemory  7h (x33 over 5d)   kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeHasSufficientMemory
  Normal   Starting                 7h                 kube-proxy, 20.20.208.6   Starting kube-proxy.
  Warning  Rebooted                 7h                 kubelet, 20.20.208.6      Node 20.20.208.6 has been rebooted, boot id: 65cc6a0a-c6c1-4ffc-990d-453e6
b44a1ea
  Normal   Starting                 7h                 kubelet, 20.20.208.6      Starting kubelet.
  Normal   NodeAllocatableEnforced  7h                 kubelet, 20.20.208.6      Updated Node Allocatable limit across pods
  Normal   NodeHasSufficientDisk    7h                 kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeHasSufficientDisk
  Normal   NodeNotReady             36m (x5 over 7h)   kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeNotReady
  Warning  ContainerGCFailed        16m (x4 over 5h)   kubelet, 20.20.208.6      rpc error: code = DeadlineExceeded desc = context deadline exceeded
  Normal   NodeHasNoDiskPressure    16m (x25 over 7h)  kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeHasNoDiskPressure
  Normal   NodeHasSufficientMemory  16m (x25 over 7h)  kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeHasSufficientMemory
  Normal   NodeReady                16m (x24 over 6h)  kubelet, 20.20.208.6      Node 20.20.208.6 status is now: NodeReady
  Warning  ImageGCFailed            6m                 kubelet, 20.20.208.6      wanted to free 4481718681 bytes, but freed 0 bytes space with errors in im
age deletion: rpc error: code = Unknown de                                       a (cannot be forced) - image is being use
d by running container ba2a24e78f85
  Warning  ImageGCFailed            1m                                           but freed 0 bytes space with errors in im
```

UC SANTA BARBARA

Motivation

- Deployment is hard
- Learning new technologies just for deployment slows development
- Installation, bootstrapping, and configuration is very error-prone

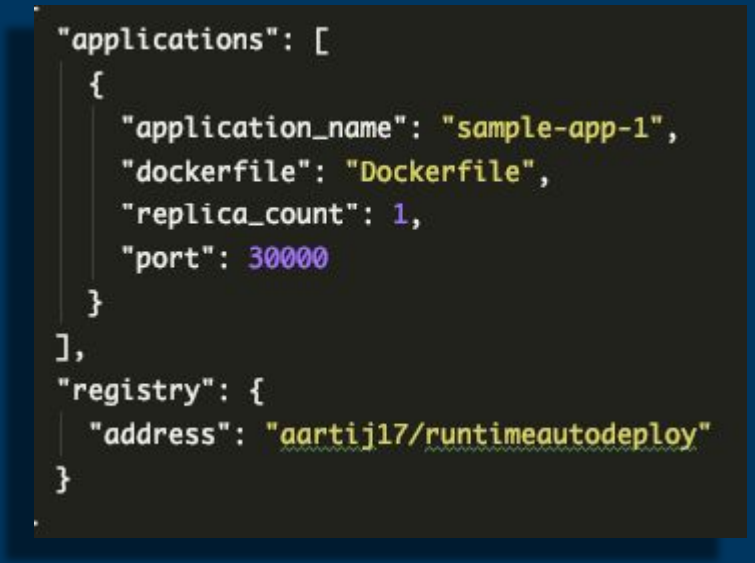
What is a container? What is kubernetes?

- A container is like a mini VM packaged with application software and everything needed to run that application - code, runtime, system tools, system libraries and settings.

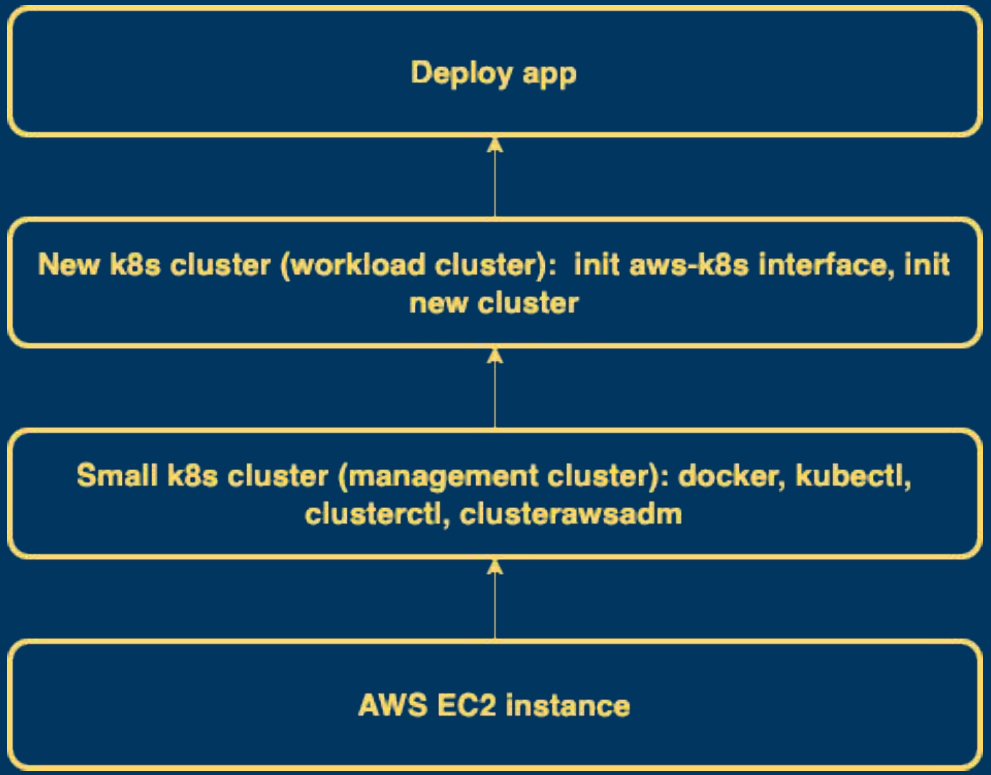
- To maintain these containers, and facilitate easy networking, version and configuration management, kubernetes is used. **TL;DR - CONTAINER LIFECYCLE MANAGEMENT**

Runtime Auto Deploy (R.A.D.)

- Automate the process of spinning off a new kubernetes cluster
- Automate the process of containerizing user applications and deploy them to Kubernetes with minimum user input



Layers of implementation

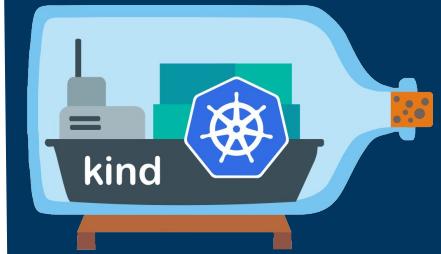


- Installing all the packages and dependencies
- Initializing a management cluster
- Initializing a workload cluster
- Download the repo, build the docker image, deploy to kubernetes

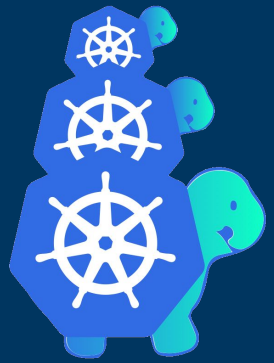
Components of the Pipeline

To deploy / initiate the RAD pipeline

To spin off a new k8s cluster

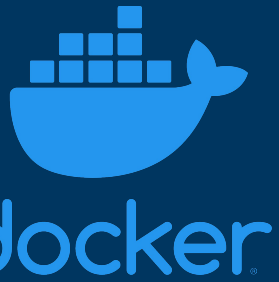


Management cluster



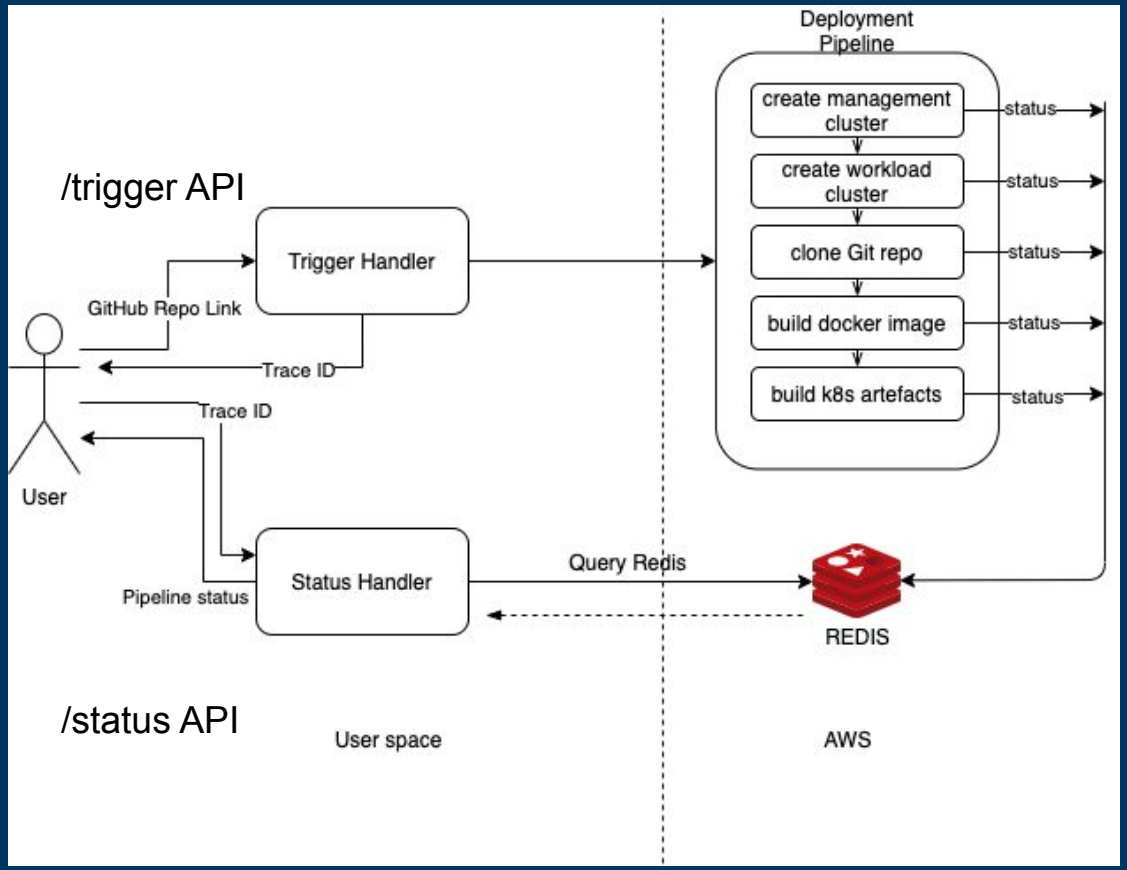
Cluster API - k8s lifecycle manager

To deploy user application



Kubernetes

Pipeline



Management Cluster

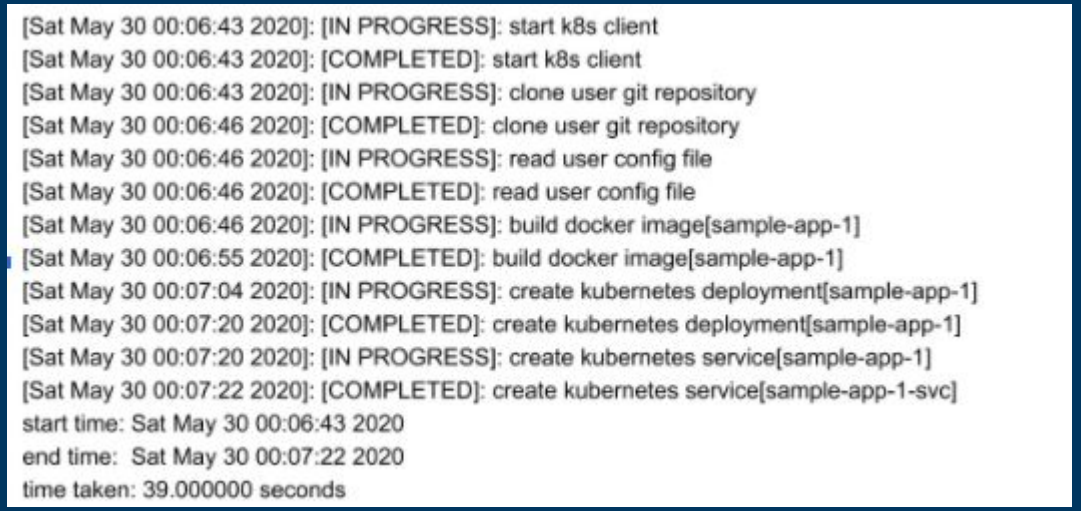
- Base cluster with a small cluster of k8s deployed

Workload Cluster

- New k8s cluster spun off using Cluster API

Status Profiler

- After you trigger the endpoint, you will get back a UID
- Use this UID to poll for the status of deployment while the pipeline is running



Evaluation

- Measure the throughput of our application from trigger to pod being deployed
- Spinning up a new kubernetes cluster is always 1740 seconds
- Tested deployment with two very different sized repositories

Repo 1

Two Images built: 13MB and 16.4MB

20 trials

Times (seconds)

- 44.0, 58.0, 85.0, 75.0, 92.0, 71.0, 112.0, 51.0, 84.0, 52.0, 101.0, 41.0, 67.0, 68.0, 63.0, 71.0, 96.0, 61.0, 61.0, 81.0

AVG: 71.7 seconds

Repo 2

One image built: 1.06 GB

10 trials

Time (seconds)

- 687.0, 621.0, 748.0, 738.0, 735.0, 1464.0, 1749.0, 1352.0, 1503.0, 1395.0

AVG: 1025.7 seconds

Thank you!