Performing Monitoring and Troubleshooting Tasks

Dan Wahlin
WAHLIN CONSULTING

@danwahlin www.codewithdan.com

Module Overview

Monitoring and Troubleshooting Overview

Web UI Dashboard in Action

Prometheus, Metrics Server, and kube-state-metrics in Action

Grafana in Action

Troubleshooting Techniques in Action

Monitoring and Troubleshooting Overview

How important is it to monitor Deployments, Pods, and other resources in your Kubernetes cluster?

Q What should you monitor?

A Short answer....everything!

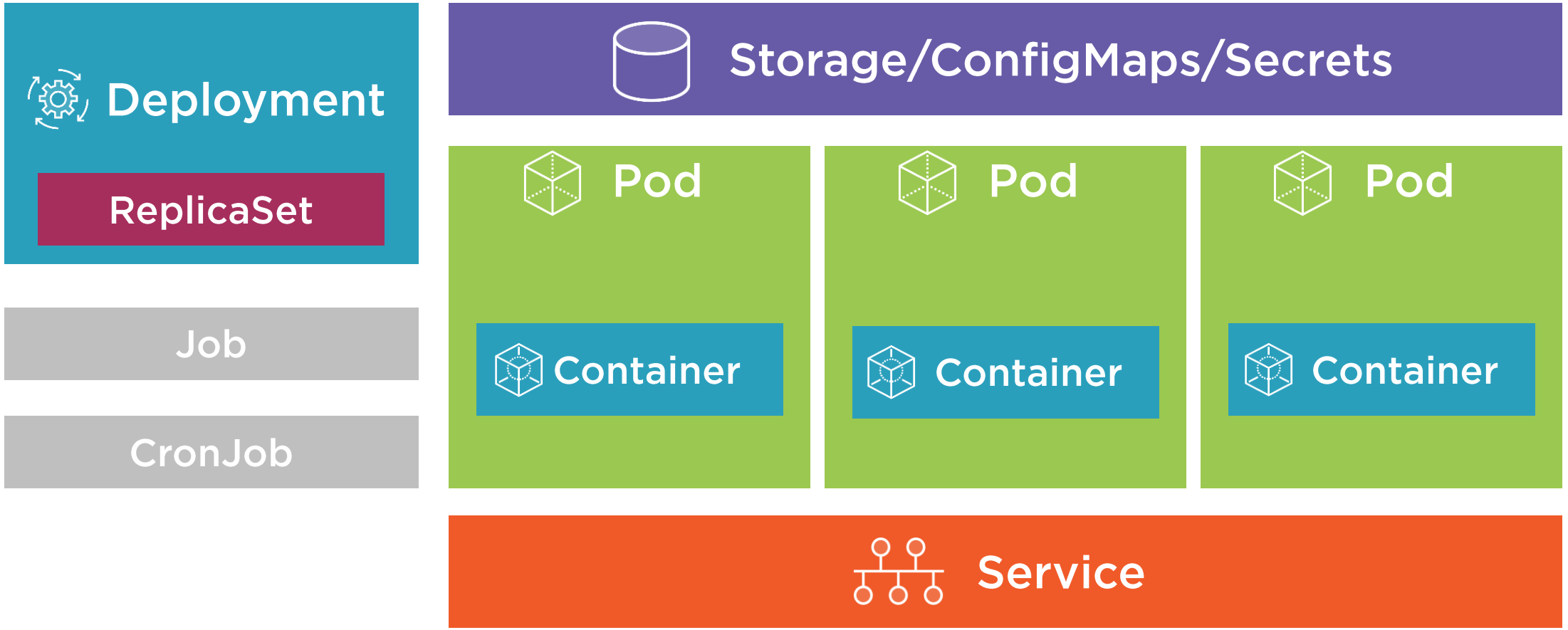
"Kubernetes makes managing a containerized infrastructure much easier by creating levels of abstractions such as pods and services. We no longer have to worry about where applications are running or if they have enough resources to work properly."

https://kubernetes.io/blog

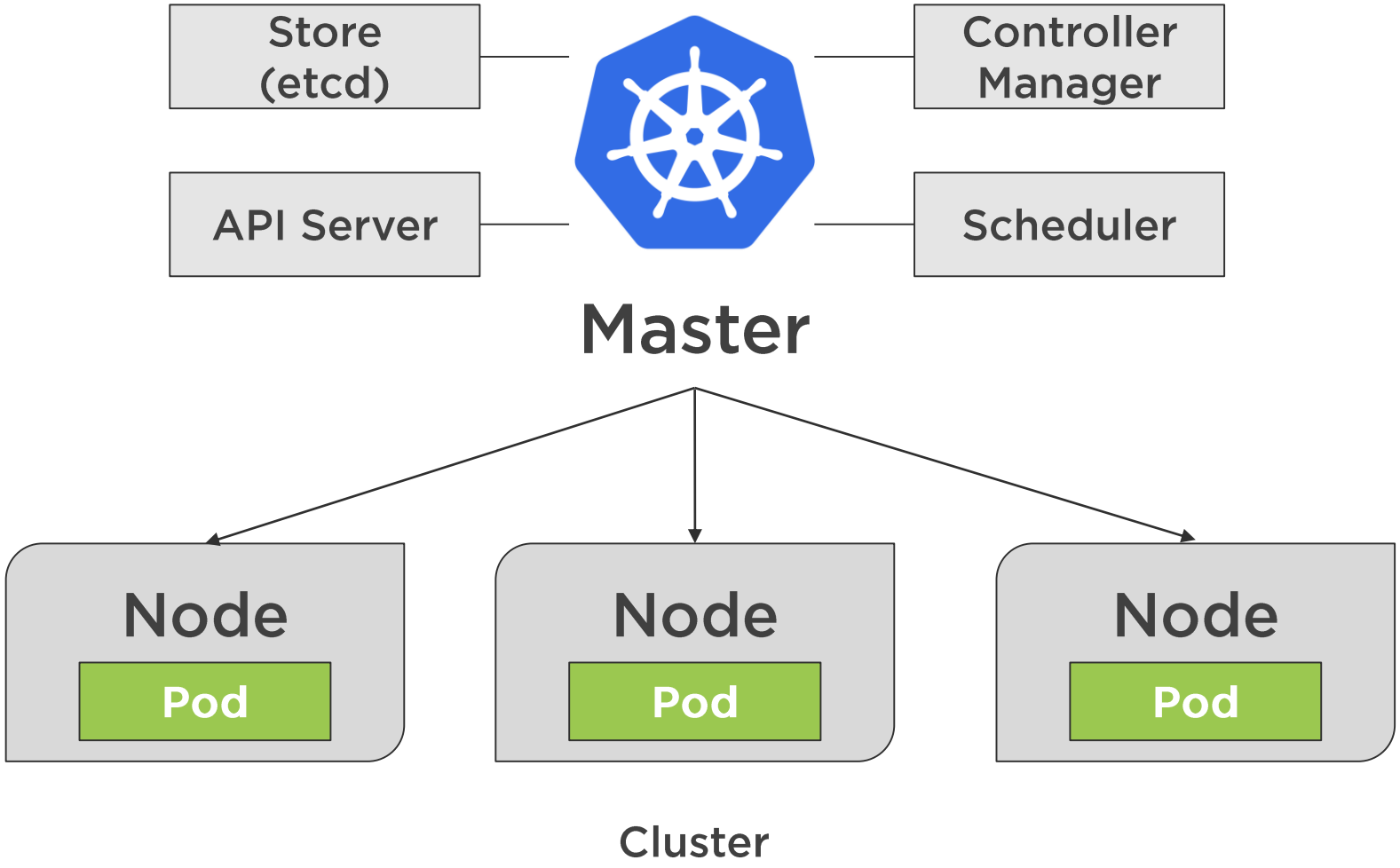
"Kubernetes makes managing a containerized infrastructure much easier by creating levels of abstractions such as pods and services. We no longer have to worry about where applications are running or if they have enough resources to work properly. But that doesn't change the fact that, in order to ensure good performance, we need to monitor our applications, the containers running them, and Kubernetes itself."

https://kubernetes.io/blog

What Should You Monitor?



What Should You Monitor?



Monitoring and Troubleshooting Options

Key monitoring and alerting options:

- Web UI Dashboard
- Metrics Server
- kube-state-metrics
- Prometheus
- Grafana
- Many more...

Web UI Dashboard in Action

"**Web UI (Dashboard)** is a web-based Kubernetes user interface. You can use Dashboard to deploy containerized applications to a Kubernetes cluster, troubleshoot your containerized application, and manage the cluster resources."

https://kubernetes.io

https://github.com/kubernetes/dashboard

Metrics Server, kube-state-metrics, and Prometheus in Action

"Metrics Server is a cluster-wide aggregator of resource usage data. It is deployed by default in clusters created by kube-up.sh script as a Deployment object."

https://kubernetes.io

https://github.com/kubernetes-sigs/metrics-server

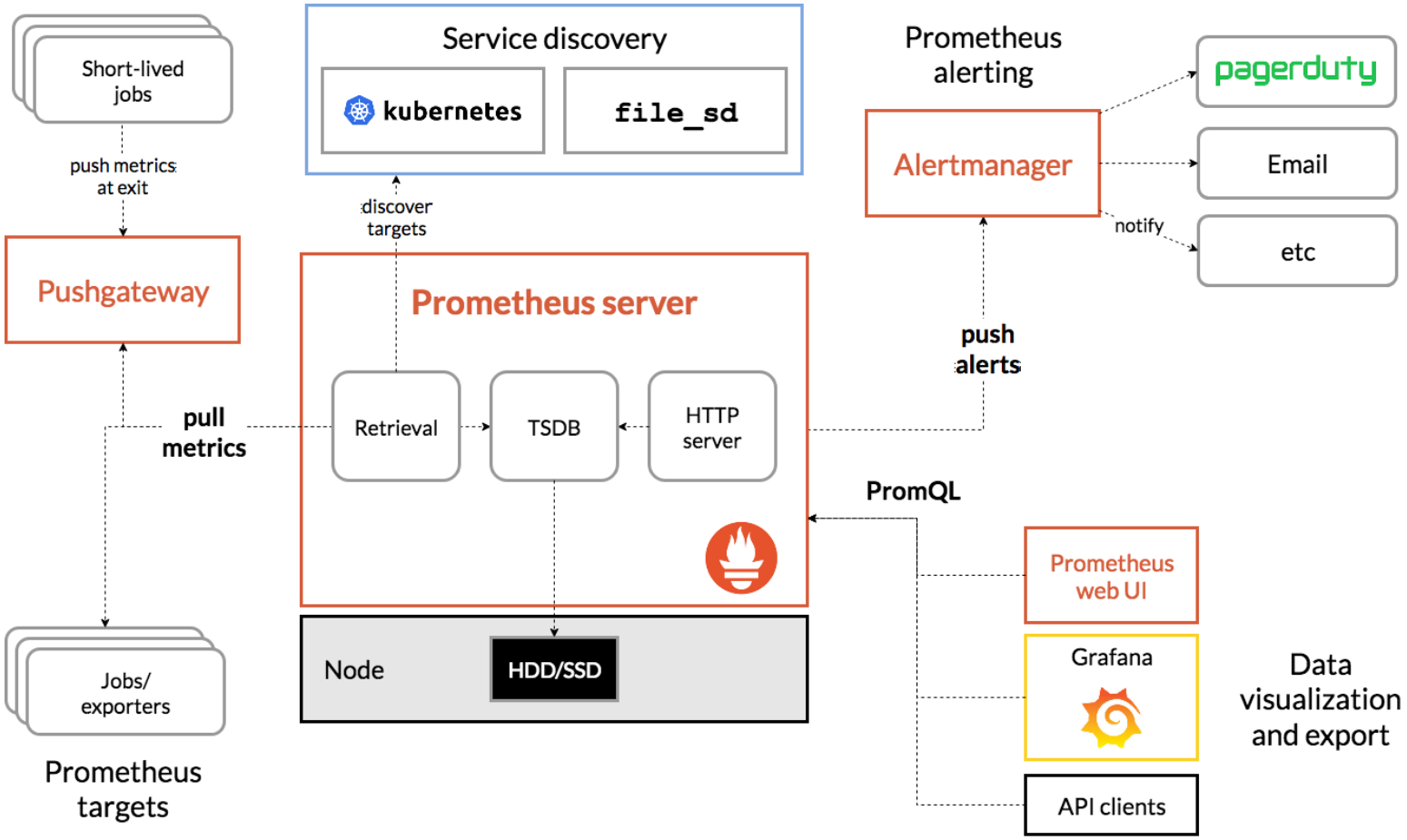
"kube-state-metrics is a simple service that listens to the Kubernetes API server and generates metrics about the state of the objects. It is not focused on the health of the individual Kubernetes components, but rather on the health of the various objects inside, such as deployments, nodes and pods."

https://github.com/kubernetes/kube-state-metrics

"Prometheus is an open-source systems monitoring and alerting toolkit originally built at SoundCloud....Prometheus joined the Cloud Native Computing Foundation in 2016 as the second hosted project, after Kubernetes."

https://prometheus.io

Prometheus Architecture



https://prometheus.io/docs/introduction/overview

Grafana in Action

"Grafana allows you to query, visualize, alert on and understand your metrics no matter where they are stored. Create, explore, and share dashboards with your team and foster a data driven culture."

https://grafana.com

Troubleshooting Techniques with kubectl

Key Troubleshooting Commands

Key troubleshooting commands:

- kubectl get pod [pod-name] -o yaml
- kubectl describe pod [pod-name]
- kubectl exec [pod-name] –it sh

Viewing Pod logs

- kubectl logs [pod-name]
- kubectl logs [pod-name] –c [container-name]
- kubectl logs –p [pod-name]
- kubectl logs –f [pod-name]

Troubleshooting Techniques in Action

Summary

Monitoring should be a critical part of any Kubernetes rollout plan

Several tools can be used to provide monitoring and alerts:

- Web UI Dashboard
- Prometheus
- Grafana
- Many more...

Master kubectl commands that can help troubleshoot issues with Kubernetes resources