NetApp

Kubernetes Service

Cloud Native Application and Infrastructure Management

Matt Baldwin
@baldwinmathew
Dir. Cloud Native & Kubernetes Engineering

NetApp

Why are we here today?

Today we'll discuss:

1) What is Cloud Native?
2) Know thy users - taking a user-centered approach.
3) Cloud Native Anywhere - the Operator Experience
4) Cloud Native App Management - giving developers what they want.

NetApp

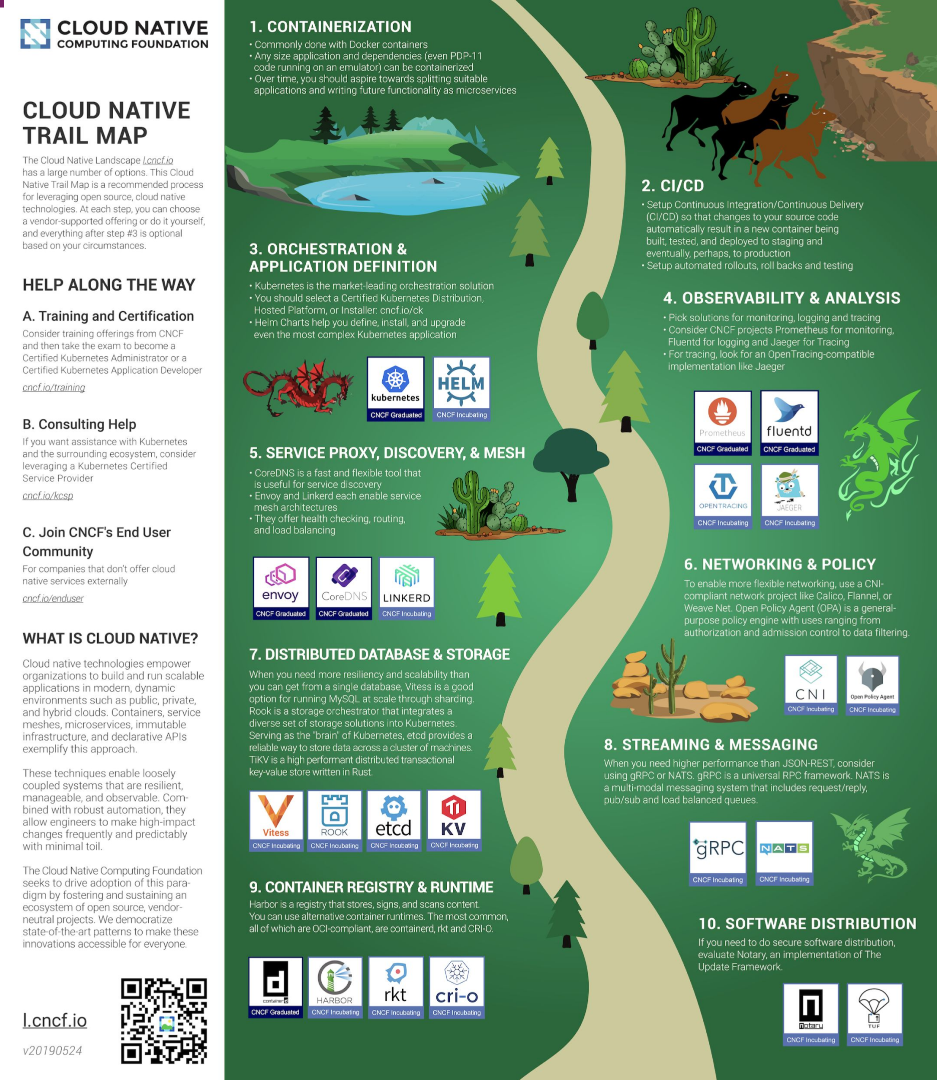
CNCF Cloud Native Definition v1.0

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

Cloud Native Trail Map

Trail Map: l.cncf.io



CLOUD NATIVE COMPUTING FOUNDATION
kubernetes CNCF Graduated
HELM CNCF Incubating
Prometheus CNCF Graduated
fluentd CNCF Graduated
OPENTRACING CNCF Incubating
JAEGER CNCF Incubating
envoy CNCF Graduated
CoreDNS CNCF Graduated
LINKERD CNCF Incubating
CNI CNCF Incubating
Open Policy Agent CNCF Incubating
Vitess CNCF Incubating
ROOK CNCF Incubating
etcd CNCF Incubating
KV CNCF Incubating
gRPC CNCF Incubating
NATS CNCF Incubating
containerd CNCF Graduated
HARBOR CNCF Incubating
rkt CNCF Incubating
cri-o CNCF Incubating
Notary CNCF Incubating
TUF CNCF Incubating

CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape l.cncf.io has a large number of options. This Cloud Native Trail Map is a recommended process for leveraging open source, cloud native technologies. At each step, you can choose a vendor-supported offering or do it yourself, and everything after step #3 is optional based on your circumstances.

HELP ALONG THE WAY

A. Training and Certification

Consider training offerings from CNCF and then take the exam to become a Certified Kubernetes Administrator or a Certified Kubernetes Application Developer

cncf.io/training

B. Consulting Help

If you want assistance with Kubernetes and the surrounding ecosystem, consider leveraging a Kubernetes Certified Service Provider

cncf.io/kcsp

C. Join CNCF's End User Community

For companies that don't offer cloud native services externally

cncf.io/enduser

WHAT IS CLOUD NATIVE?

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

The Cloud Native Computing Foundation seeks to drive adoption of this paradigm by fostering and sustaining an ecosystem of open source, vendor-neutral projects. We democratize state-of-the-art patterns to make these innovations accessible for everyone.

l.cncf.io

v20190524

1. CONTAINERIZATION

- Commonly done with Docker containers
- Any size application and dependencies (even PDP-11 code running on an emulator) can be containerized
- Over time, you should aspire towards splitting suitable applications and writing future functionality as microservices

2. CI/CD

- Setup Continuous Integration/Continuous Delivery (CI/CD) so that changes to your source code automatically result in a new container being built, tested, and deployed to staging and eventually, perhaps, to production
- Setup automated rollouts, roll backs and testing

3. ORCHESTRATION & APPLICATION DEFINITION

- Kubernetes is the market-leading orchestration solution
- You should select a Certified Kubernetes Distribution, Hosted Platform, or Installer: cncf.io/ck
- Helm Charts help you define, install, and upgrade even the most complex Kubernetes application

4. OBSERVABILITY & ANALYSIS

- Pick solutions for monitoring, logging and tracing
- Consider CNCF projects Prometheus for monitoring, Fluentd for logging and Jaeger for Tracing
- For tracing, look for an OpenTracing-compatible implementation like Jaeger

5. SERVICE PROXY, DISCOVERY, & MESH

- CoreDNS is a fast and flexible tool that is useful for service discovery
- Envoy and Linkerd each enable service mesh architectures
- They offer health checking, routing, and load balancing

6. NETWORKING & POLICY

To enable more flexible networking, use a CNI-compliant network project like Calico, Flannel, or Weave Net. Open Policy Agent (OPA) is a general-purpose policy engine with uses ranging from authorization and admission control to data filtering.

7. DISTRIBUTED DATABASE & STORAGE

When you need more resiliency and scalability than you can get from a single database, Vitess is a good option for running MySQL at scale through sharding. Rook is a storage orchestrator that integrates a diverse set of storage solutions into Kubernetes. Serving as the "brain" of Kubernetes, etcd provides a reliable way to store data across a cluster of machines. TiKV is a high performant distributed transactional key-value store written in Rust.

8. STREAMING & MESSAGING

When you need higher performance than JSON-REST, consider using gRPC or NATS. gRPC is a universal RPC framework. NATS is a multi-modal messaging system that includes request/reply, pub/sub and load balanced queues.

9. CONTAINER REGISTRY & RUNTIME

Harbor is a registry that stores, signs, and scans content. You can use alternative container runtimes. The most common, all of which are OCI-compliant, are containerd, rkt and CRI-O.

10. SOFTWARE DISTRIBUTION

If you need to do secure software distribution, evaluate Notary, an implementation of The Update Framework.

Cloud Native Pillars

1) Portability - multi-cloud, on-premise, data anywhere.

2) Cloud Native Application Management

3) Security - at rest, at run-time, in-flight

4) Storage - across clouds & onprem

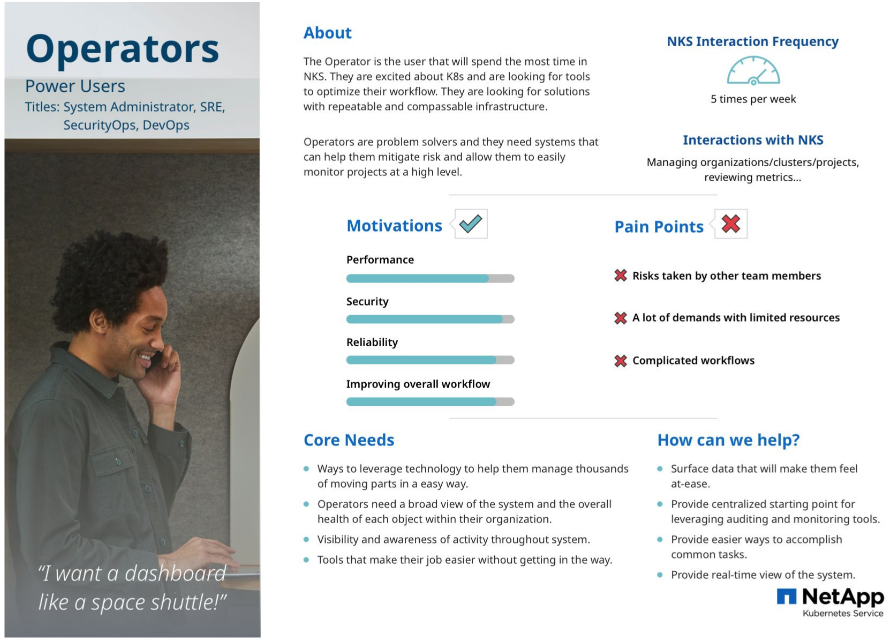
5) Managed Data Services - Kafka, Postgres, MySQL

6) Personas -- who are your users? Operator, Developer, Executive

NetApp

User-centered

Who are the users? What are their motivations and pain points?



Operators

Power Users

Titles: System Administrator, SRE, SecurityOps, DevOps

"I want a dashboard like a space shuttle!"

About

The Operator is the user that will spend the most time in NKS. They are excited about K8s and are looking for tools to optimize their workflow. They are looking for solutions with repeatable and compassable infrastructure.

Operators are problem solvers and they need systems that can help them mitigate risk and allow them to easily monitor projects at a high level.

NKS Interaction Frequency

5 times per week

Interactions with NKS

Managing organizations/clusters/projects, reviewing metrics...

Motivations ✓

- Performance
- Security
- Reliability
- Improving overall workflow

Pain Points ✗

- ✗ Risks taken by other team members
- ✗ A lot of demands with limited resources
- ✗ Complicated workflows

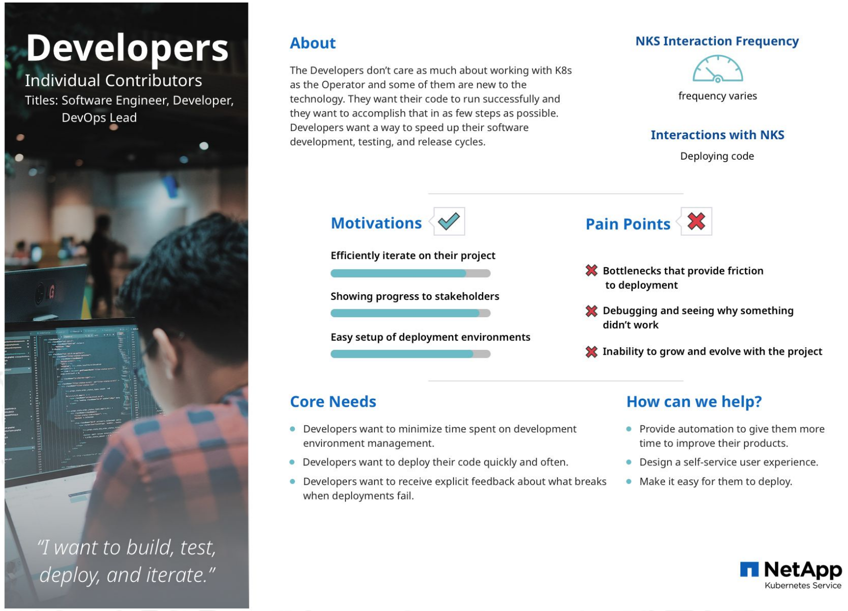
Core Needs

- Ways to leverage technology to help them manage thousands of moving parts in a easy way.
- Operators need a broad view of the system and the overall health of each object within their organization.
- Visibility and awareness of activity throughout system.
- Tools that make their job easier without getting in the way.

How can we help?

- Surface data that will make them feel at-ease.
- Provide centralized starting point for leveraging auditing and monitoring tools.
- Provide easier ways to accomplish common tasks.
- Provide real-time view of the system.

NetApp Kubernetes Service



Developers

Individual Contributors

Titles: Software Engineer, Developer, DevOps Lead

"I want to build, test, deploy, and iterate."

About

The Developers don't care as much about working with K8s as the Operator and some of them are new to the technology. They want their code to run successfully and they want to accomplish that in as few steps as possible. Developers want a way to speed up their software development, testing, and release cycles.

NKS Interaction Frequency

frequency varies

Interactions with NKS

Deploying code

Motivations ✓

- Efficiently iterate on their project
- Showing progress to stakeholders
- Easy setup of deployment environments

Pain Points ✗

- ✗ Bottlenecks that provide friction to deployment
- ✗ Debugging and seeing why something didn't work
- ✗ Inability to grow and evolve with the project

Core Needs

- Developers want to minimize time spent on development environment management.
- Developers want to deploy their code quickly and often.
- Developers want to receive explicit feedback about what breaks when deployments fail.

How can we help?

- Provide automation to give them more time to improve their products.
- Design a self-service user experience.
- Make it easy for them to deploy.

NetApp Kubernetes Service

NetApp

Cloud Native Anywhere - Ops!

Cloud Native Anywhere

A unified management platform regardless of where your Kubernetes cluster lives.

1) Multi-cloud ready

2) What public cloud services do you consume?
   Hint : more than you think.

3) Lifecycle - are you really working with cattle?

4) Chaos - deploying to prod.

5) Managing access

6) Scaling

NetApp

Cloud Native Anywhere

A unified management platform regardless of where your Kubernetes cluster lives.

1) Managed Kubernetes on:
   - Microsoft, Google, Amazon
   - NetApp HCI, General VMware, FlexPod

2) AnydayOps - Day 1, Day 2 for infrastructure & applications workload.

3) Istio Service Mesh Management - canary, blue/green, A/B

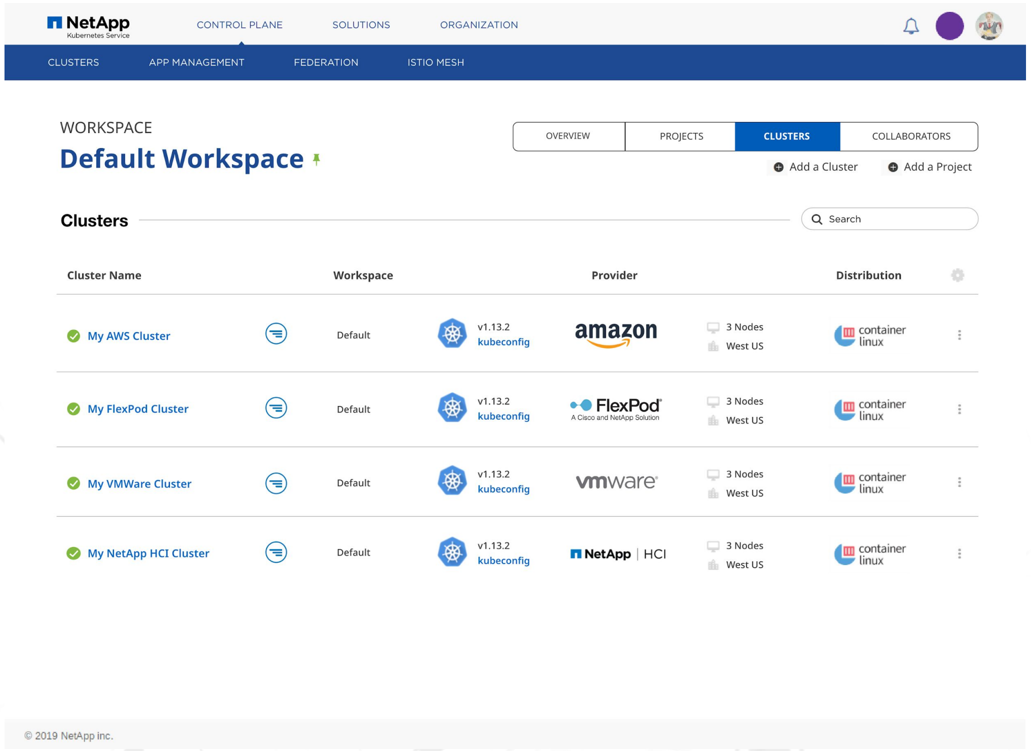
4) Roles-based Controls for Users and Teams

5) GPU Support

6) Scale from POC to High Availability

7) Private Topology

8) And moaaaaar!



WORKSPACE

Default Workspace

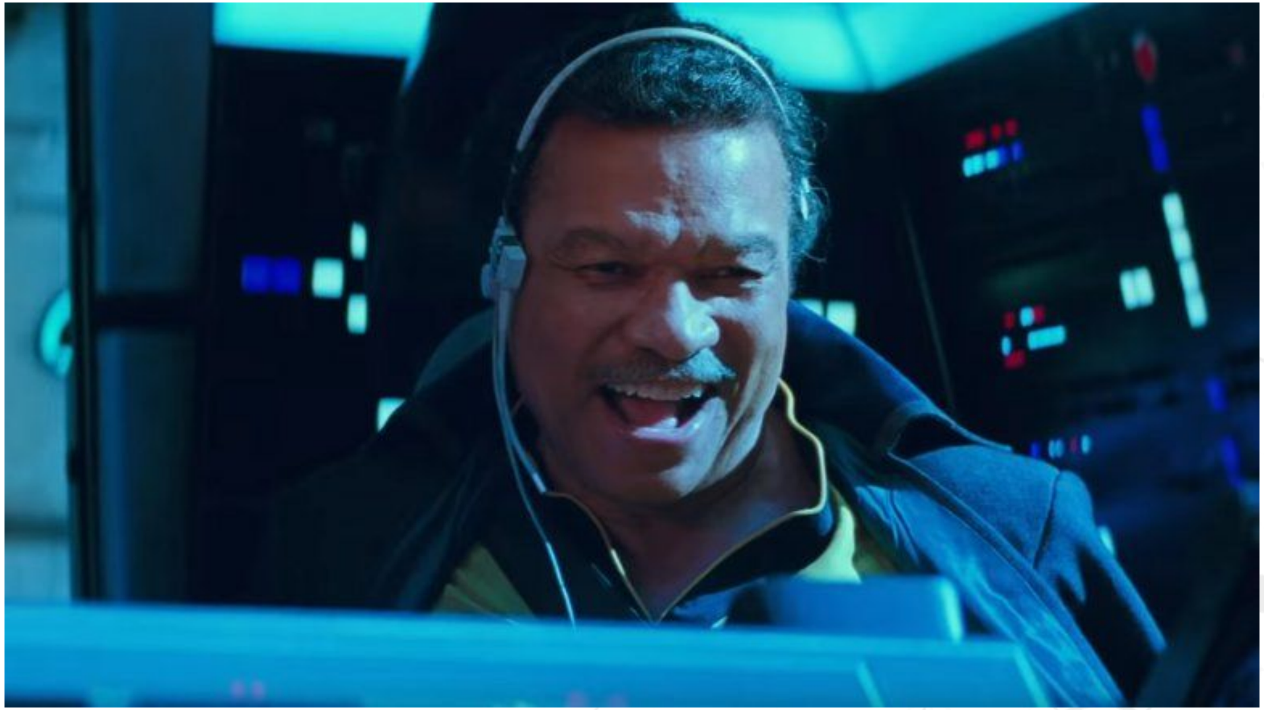
Clusters

Cluster Name	Workspace		Provider		Distribution
My AWS Cluster	Default	v1.13.2 kubeconfig	amazon	3 Nodes West US	container linux
My FlexPod Cluster	Default	v1.13.2 kubeconfig	FlexPod A Cisco and NetApp Solution	3 Nodes West US	container linux
My VMWare Cluster	Default	v1.13.2 kubeconfig	vmware	3 Nodes West US	container linux
My NetApp HCI Cluster	Default	v1.13.2 kubeconfig	NetApp | HCI	3 Nodes West US	container linux

© 2019 NetApp inc.

Demo!

But what about the devs?!



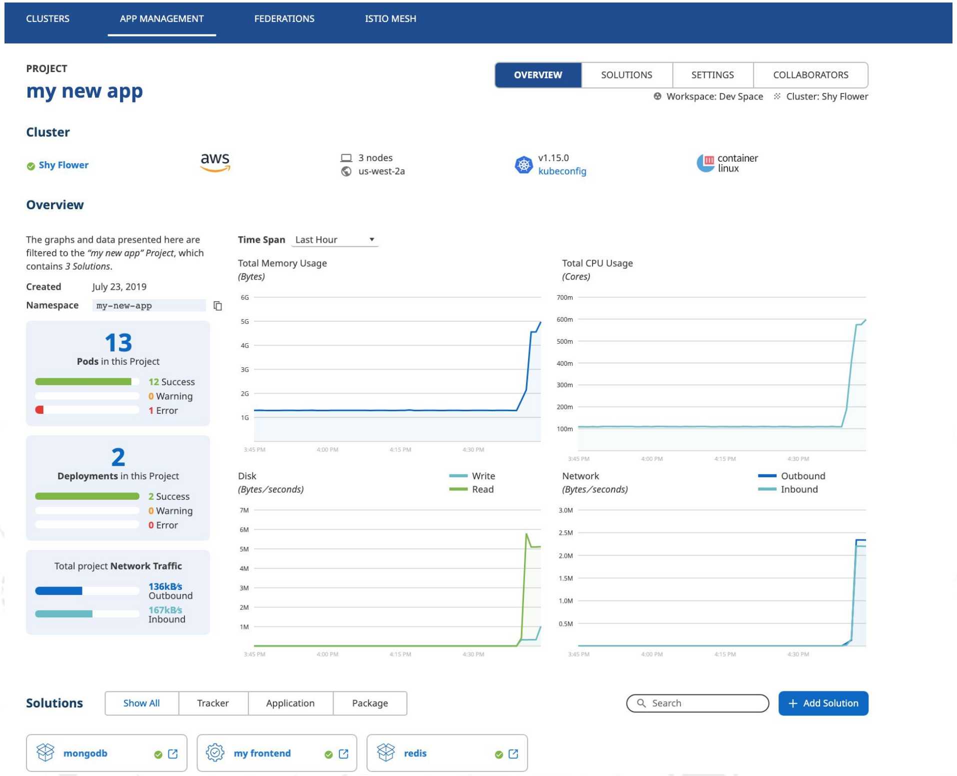
What do they want?

What do they want? TO PUSH CODE!

Cloud Native App Management

NetApp's App Management includes:

1) git based app deployment into any cluster anywhere - onprem, offprem.
2) Developers build, test, ship, and run applications in Projects, which are tenanted namespaces.
3) Choose your own adventure - deploy using git, Helm, kubectl, or other.
4) Lifecycle for your App regardless of how it is spun up in K8s - Tracker, Workflow, Package
5) Simplicity of Heroku built into Kubernetes.
   - ***git push nks master***
6) App Health, Metrics, and Logging Dashboards per Project
7) Configure Autoscaling by App, Project, or Cluster
8) Set Resource and Quota Configuration
9) And moaaaaar!



PROJECT
my new app

Cluster
Shy Flower · aws · 3 nodes us-west-2a · v1.15.0 kubeconfig · container linux

Overview

The graphs and data presented here are filtered to the *"my new app" Project*, which contains *3 Solutions*.

Created July 23, 2019
Namespace `my-new-app`

13 Pods in this Project: 12 Success, 0 Warning, 1 Error

2 Deployments in this Project: 2 Success, 0 Warning, 0 Error

Total project Network Traffic: 136kB/s Outbound, 167kB/s Inbound

Time Span: Last Hour

NetApp

What You Are Going To See

We’re taking an application-centric view of deploying to Kubernetes

- *Solutions* define how you want to deploy your application
- Applications are deployed to *Projects* as *Solutions*
- *Solutions* come with dashboards for things like metrics, logs, configuration
- *Solutions* can be delivered through Git

NetApp

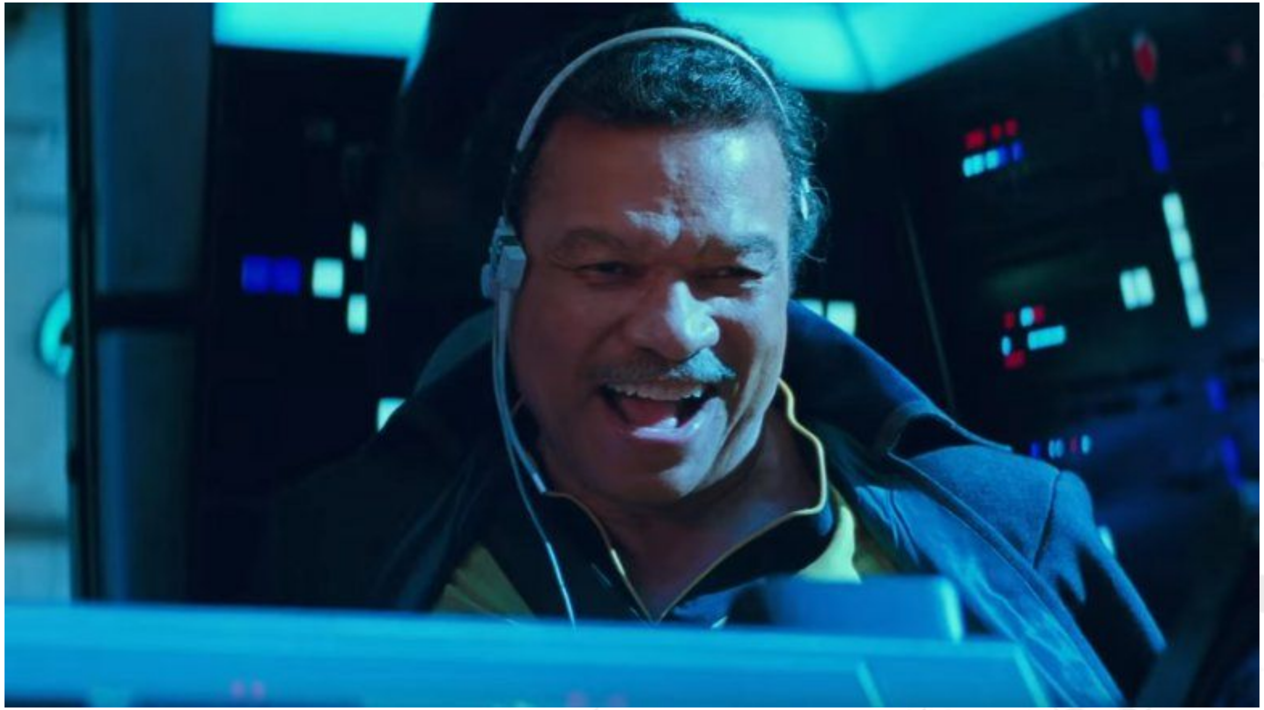
Projects

- Container for grouping components of your application:
- Ex. wordpress
  - Mysql
  - Wordpress
- Kubernetes namespace + additional features (RBAC)
- Default NetworkPolicy
- Resource Quotas and Limits

Solutions

1) Tracker
   - Bring your own application workload
   - Label selectors
     - Ex. app:staging
2) Git Workflow
   - ***git commit***
   - ***git push***
3) Helm Charts
   - Define where your chart lives, changes you want to make to values.yaml
   - We take care of the rest (without tiller!)
   - Lifecycle deployed Packages
4) Default PodSecurityPolicy
5) Automatic Management of Workload Autoscaling

DevX DEMO TIME!



Questions

NetApp

NetApp

Kubernetes Service

Thank You!

Try it out. You can sign-in at nks.netapp.io and begin building.

Send a DM - @baldwinmathew

Or drop an email: matt.baldwin@netapp.com

… oh yeah - we're hiring!

NetApp