



Top new CNCF projects to look
out for



SWETUGGT
GÖTEBORG

What value do you get by attending this talk?

- Get inspired
- Get to know cool CNCF projects



@AnnieTalvasto

Who am I?

@AnnieTalvasto

Sr. Product Marketing Mgr at Camunda



- CNCF Ambassador
- Azure MVP

- Kubernetes & CNCF meetup co-organizer
- Startup-coach
- Co-host of Cloudgossip podcast - cloudgossip.net



@AnnieTalvasto



**CLOUD NATIVE
COMPUTING FOUNDATION**

Building sustainable ecosystems for cloud native software

The Cloud Native Computing Foundation (CNCF) hosts critical components of the global technology infrastructure. CNCF brings together the world's top developers, end users, and vendors and runs the largest open source developer conferences. CNCF is part of the nonprofit Linux Foundation.



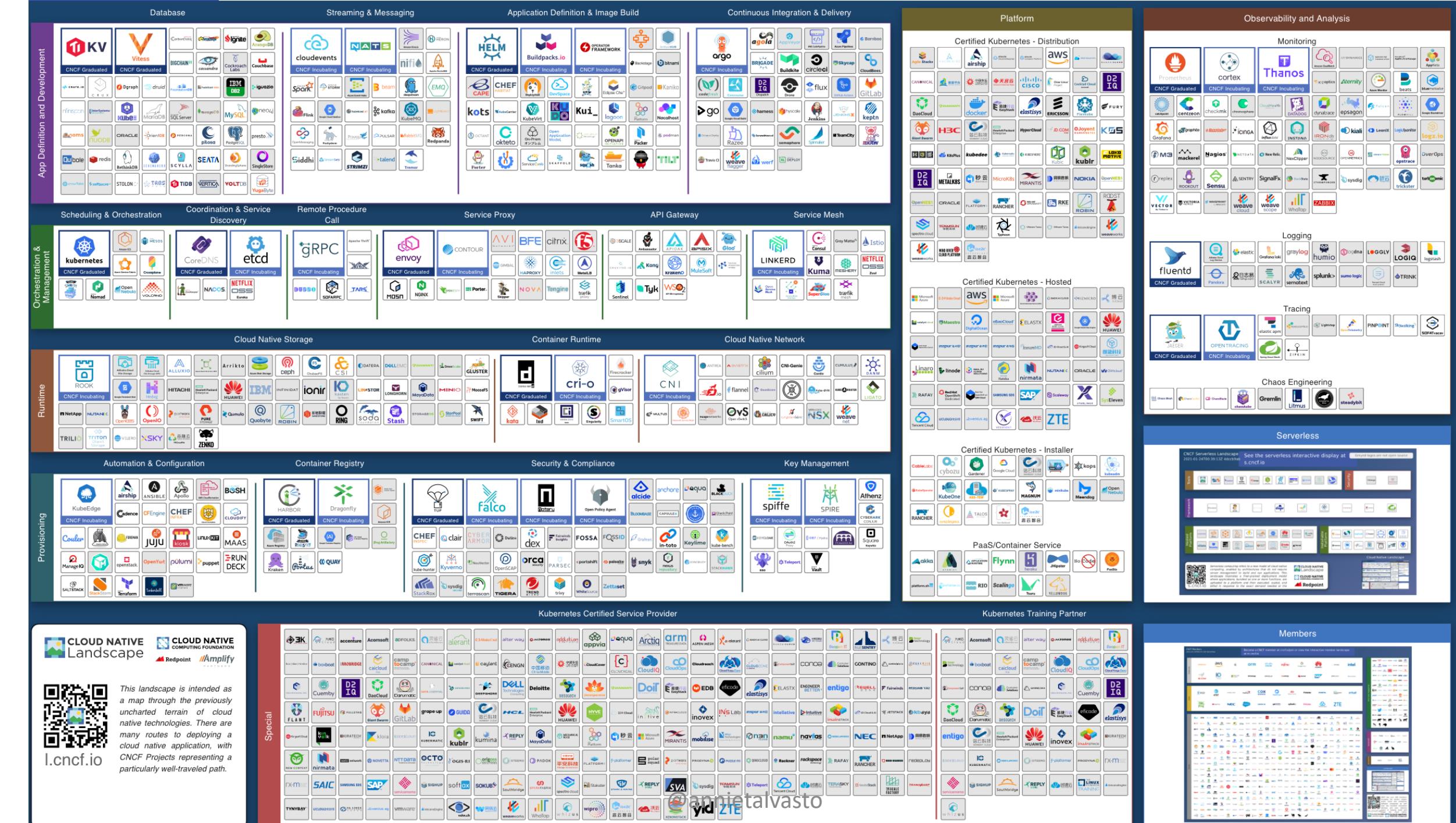
@AnnieTalvasto

Impact of cloud native

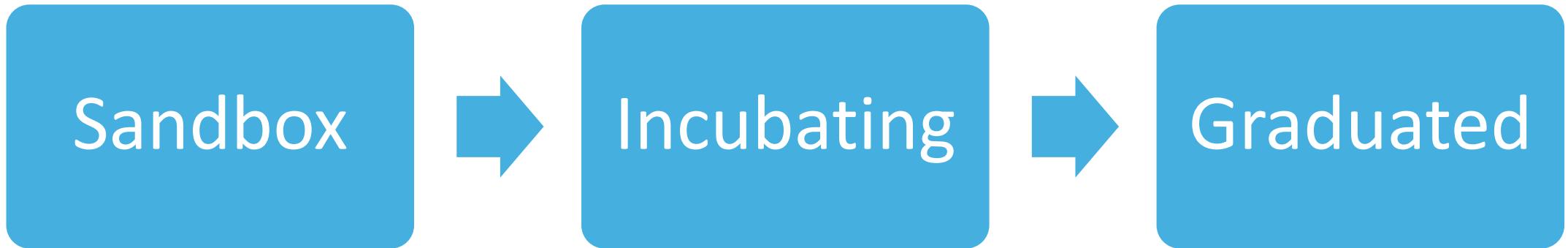
- Kubernetes has crossed the adoption chasm to become a mainstream global technology.
- According to CNCF's respondents, 96% of organizations are either using or evaluating Kubernetes – a record high since the surveys began in 2016.
 - CNCF 2021 Survey



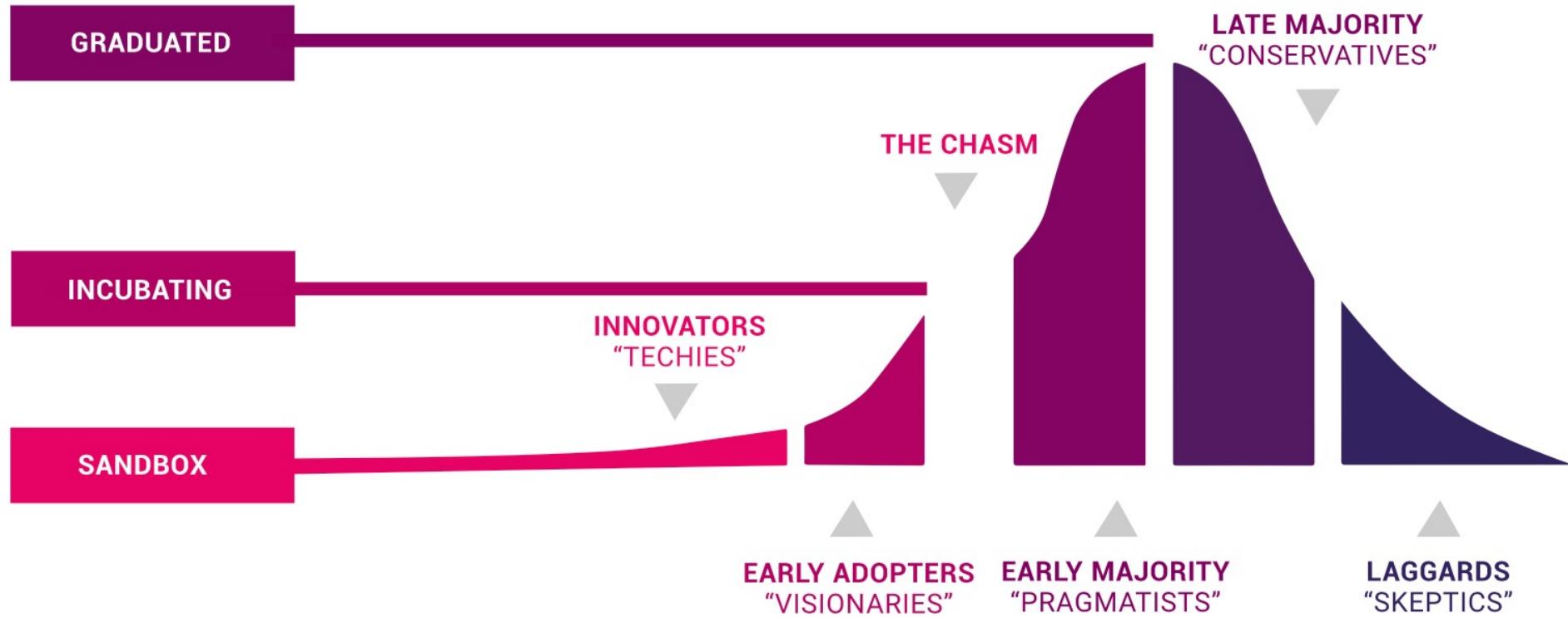
@AnnieTalvasto



3 stages of CNCF projects



@AnnieTalvasto



@AnnieTalvasto

The projects in this session

- This is not fortune telling
- Expectation management: usually CNCF intro to projects talks are around 30 to 45 minutes – this is a shorter talk covering many projects.
- Helm, Linkerd, Keda, Flux, Kubevirt, Kudo, Meshery, Kyverno



@AnnieTalvasto



The package manager for Kubernetes

Helm is the best way to find, share, and
use software built for Kubernetes.



What is Helm?

- Package manager for Kubernetes
- Homebrew, snap or chocolatey for kubernetes
- **Helm maintainer:**
- Package management: Tooling that enables someone who has knowledge of an application and a platform to package up an application so that **someone else who has neither** extensive knowledge of the application or the way it needs to be run on the platform **can use it.**



@AnnieTalvasto

What are the benefits of Helm?

- Manage Complexity
- Easy Updates
- Simple Sharing
- Rollbacks



@AnnieTalvasto

What are the principles of Helm?

- Helm takes security very seriously
- Multiple maintainers, multiple companies.
- Power user email lists, release candidates.
- Supports mac, linux, windows
- Passed 1 million downloads a month already in 2019



@AnnieTalvasto

How is Helm used?

- Charts
- What are the prerequisites?
 - A Kubernetes cluster
 - Deciding what security configurations to apply to your installation, if any
 - Installing and configuring Helm.



@AnnieTalvasto



Artifact HUB

- CNCF sandbox project

- Find Helm charts easily
- Find, install and publish Kubernetes packages
- The Artifact Hub goal is to provide a single experience for consumers that any CNCF project can leverage.



@AnnieTalvasto

Helm Demo:
Easily deploy complex
application (WordPress) to
Kubernetes using a helm chart



@AnnieTalvasto

2021-09-30 Announcing Linkerd 2.11 with authorization policy! [Read more »](#)



A different kind of service mesh

Ultra light, ultra simple, ultra powerful. Linkerd adds security, observability, and reliability to Kubernetes, *without* the complexity. CNCF-hosted and 100% open source.

[Get Started](#)[Get Involved](#)

 Star 7,784  Watch 190  Fork 906

Instant platform health metrics

Instantly track success rates, latencies, and request volumes for every meshed workload, without changes or config.

Zero-config mutual TLS

Transparently add mutual TLS to any on-cluster TCP communication with no configuration.

Drop-in reliability features

Simpler than any other mesh

Minimalist, Kubernetes-native design. No hidden magic, as little YAML, and as few CRDs as possible.

Designed by engineers, for engineers

Self-contained control plane, incrementally deployable, and lots of diagnostics and debugging tools.

Linkerd in production 101: what you need to know

Join our hands-on workshop
Tue, Dec 14, 9-10 am PT

[Register today](#)

Linkerd

- Service mesh
- Ultralight, ultrafast, security-first service mesh for Kubernetes.
- The overall goal is to reduce mental overhead of having service mesh
- What does it do?
 - Observability
 - Reliability
 - Security



@AnnieTalvasto

What are the benefits of Linkerd?

- Thriving open source community
- Simple, minimalist design
- Deep Runtime Diagnostics
- Ultralight and ultra fast
- Installs in seconds with zero config
- Actionable service metrics



@AnnieTalvasto

What are the Linkerd principles?

- Just works
- Ultralight
- Simple
- Security first
- Linkerd has a custom proxy, Linkerd2-proxy.



@AnnieTalvasto

What is needed to use Linkerd:

```
cat deployment.yml | linkerd inject - | kubectl apply -f -
```



@AnnieTalvasto

Linkerd in practice

Projects: Linkerd,
Prometheus, Flux, Thanos

Case study: Nordic retail giant's (Elkjøp)Kubernetes & Linkerd-based platform reduced hosting costs by 80%

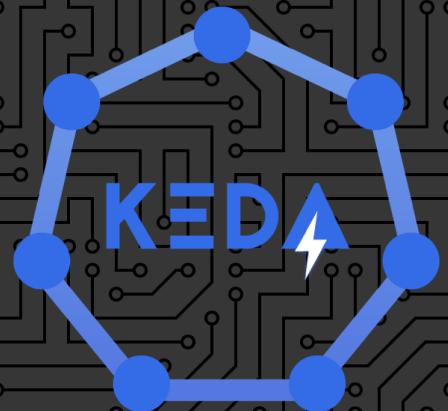
Challenge

Solution

Impact

Scale: 200+ microservices in production

Agility: Ability to create architecture changes with a smaller infrastructure footprint



Kubernetes Event-driven Autoscaling

Application autoscaling made simple

[Concepts](#)[Deploying KEDA](#)[Architecture](#)[Scalers](#)[Blog](#)

What is KEDA?

KEDA is a [Kubernetes](#)-based Event Driven Autoscaler. With KEDA, you can drive the scaling of any container in Kubernetes based on the number of events needing to be processed.

KEDA is a single-purpose and lightweight component that can be added into any Kubernetes cluster. KEDA works alongside standard Kubernetes components like the [Horizontal Pod Autoscaler](#) and can extend functionality without overwriting or duplication. With KEDA you can explicitly map the apps you

What is Keda?

- Default Kubernetes Scaling is not well suited for event driven applications, kubernetes is more for resource based scaling (CPU and memory).
- Keda: Event driven scale controlling that can run inside any kubernetes cluster.
- You can install it into new or existing clusters.



@AnnieTalvasto

What are the Keda principles?

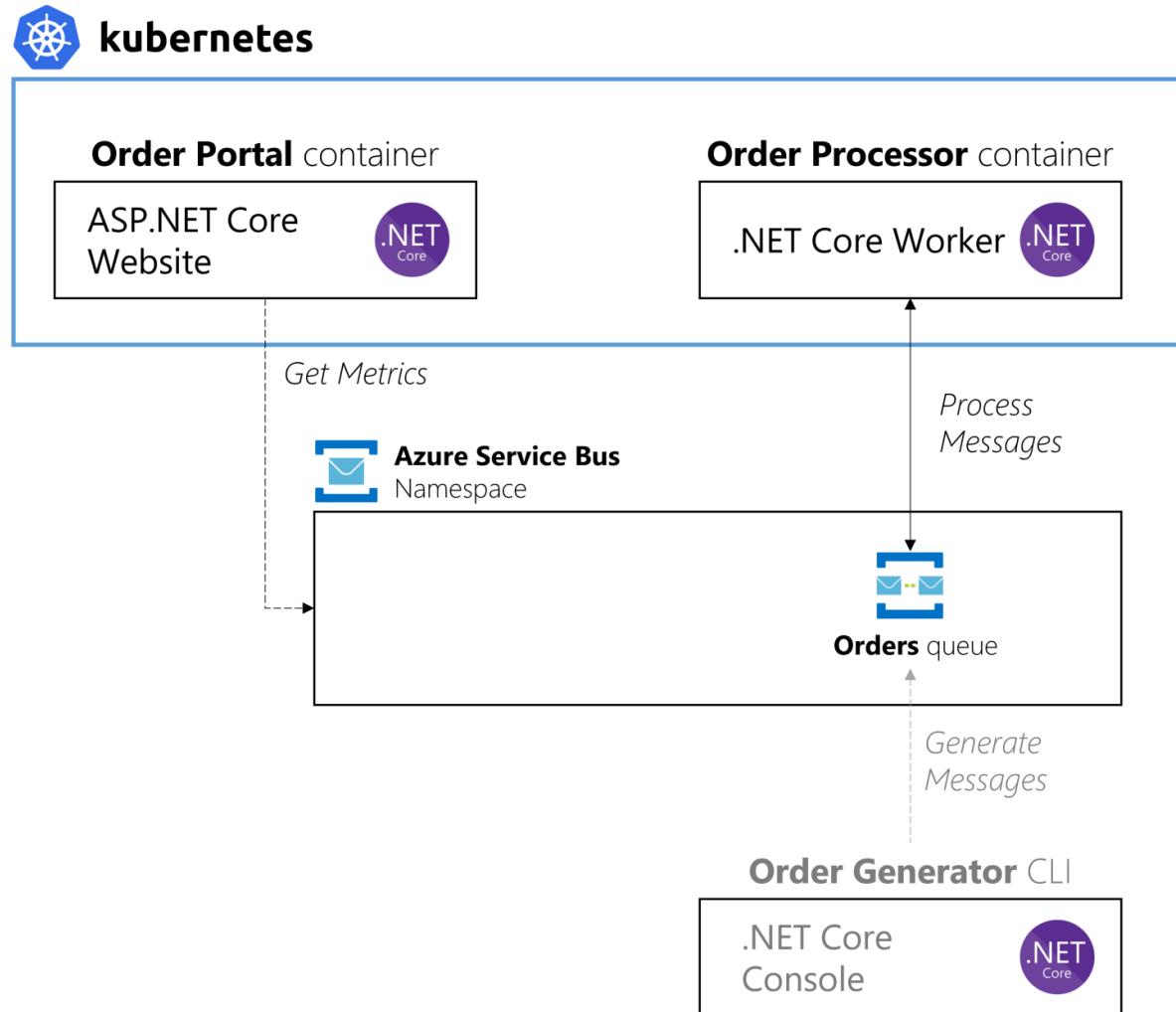
- Not rebuilding anything that Kubernetes offers out of the box.
- Single purpose, simple, non-intrusive.
- Works with any container and any workload



@AnnieTalvasto

Demo

KEDA:
Scaling .NET
Core worker
with Azure
Service Bus



@AnnieTalvasto

 **Server-Side Apply has landed in Flux!**

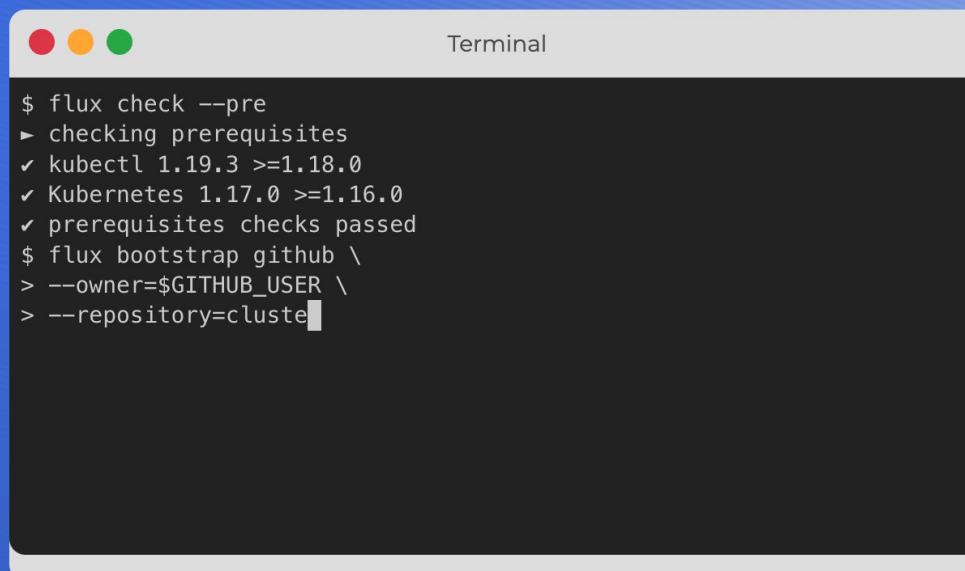
Flux - the GitOps family of projects

Flux is a set of continuous and progressive delivery solutions for Kubernetes that are open and extensible.

The latest version of Flux brings many new features, making it more flexible and versatile.

Flux is a CNCF Incubating project.

[Get started →](#)



A terminal window titled "Terminal" showing a flux command being run. The command checks prerequisites and then boots up the flux controller in a GitHub repository.

```
$ flux check --pre
► checking prerequisites
✓ kubectl 1.19.3 >=1.18.0
✓ Kubernetes 1.17.0 >=1.16.0
✓ prerequisites checks passed
$ flux bootstrap github \
> --owner=$GITHUB_USER \
> --repository=cluster-api
```

Flux in short

What is Flux?

- What is GitOps?

- Kubectl apply

- Kubectl set image

- Helm upgrade

- Kubectl upgrade

- > git push

GitOps provides one model for making infrastructure, apps and Kubernetes add-on changes, you have consistent end-to-end workflow across your entire organization

- Why is Flux great for GitOps?



@AnnieTalvasto

Flux practices & benefits

Defined GitOps practices:

- 1) Describe your system declaratively
- 2) Keep configuration under source control
- 3) Use software agents to reconcile and ensure correctness and alert for drift

Benefits

- Collaboration on infra
- Access Control
- Auditable History
- Drift Correction
- Clear boundaries between dev-team and kubernetes



@AnnieTalvasto

Building a virtualization API for Kubernetes

 [Watch Our Intro Video](#)

Try KubeVirt now

Try KubeVirt right now on the following platforms



KubeVirt on katacoda

[Try Katacoda!](#)



KubeVirt on MiniKube

[Try MiniKube!](#)



KubeVirt on Kind

[Try Kind!](#)



KubeVirt on Cloud Provider

[Try It!](#)

Why KubeVirt?

KubeVirt technology addresses the needs of development teams that have adopted or want to adopt [Kubernetes](#) but possess existing Virtual Machine-based workloads that cannot be easily containerized. More specifically, the technology provides a unified development platform where developers can

Recent posts

[KubeVirt v0.50.0](#)

35

What is Kubenvirt?

- Enabling developers to work with new containerized technologies.
 - Legacy will not go away, but you can work with it.
- >The technology provides a unified development platform where developers can build, modify, and deploy applications residing in both Application Containers as well as Virtual Machines.
- Declarative - Kubectl



@AnnieTalvasto

What are the benefits of Kubevirt?

- Manage virtual machines for impractical-to-containerize apps.
- Combine existing virtualized workloads with new container workloads on the one platform.
- Support development of new microservice applications in containers that interact with existing virtualized applications.



@AnnieTalvasto



The Kubernetes Universal Declarative Operator

↔ Get Started ↔

Focus on your software ...

The Kubernetes Universal Declarative Operator (KUDO) is a highly productive toolkit for writing Kubernetes Operators.

... not on deploying to Kubernetes

Using KUDO you can deploy your applications, have the tools needed to operate them, and understand how they're behaving – all without a Ph.D. in Kubernetes.

Automate Day-2 Operations

KUDO lets you configure an Operator's entire lifecycle using a declarative spec, including things like backup/restore. You don't have to write Go unless you want to.

What is KUDO?

KUDO is a toolkit that makes it easy to build [Kubernetes Operators](#), in most cases just using YAML.

It provides a set of pre-built Operators, that you can use out of the box or easily customize.

Finally, KUDO lets you standardize the way you run Operators.

What is Kudo?

- Stateless vs stateful app
- Kubernetes has been very focused on stateless apps – and stateful apps do not like it -> solution: operators.
- Building operators requires deep expertise and may require thousands of lines of code -> substantial engineering resource needed.
- **Kudo = Kubernetes Universal Declarative Operator**



@AnnieTalvasto

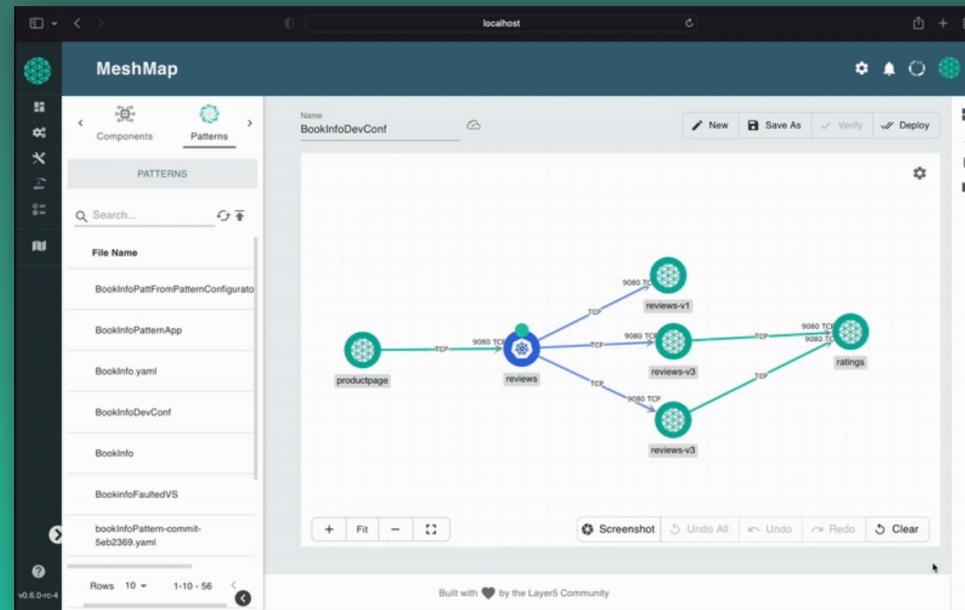
What are the benefits of Kudo?

- Kudo can create operators without needing deep knowledge of kubernetes or coding – by defining lifecycle stages.
- Just kubernetes APIs, a lot easier to learn
- Has kubernetes native management, aka using of kubectl and other familiar tools



@AnnieTalvasto

the extensible service mesh manager



cloud native management with plugins

Meshery

What is Meshery?

The Service Mesh Management Plane

Service mesh:

Control plane

Data plane

+

Management plane



@AnnieTalvasto

What is Meshery?

- Supports over 10 different service meshes
- Multi-mesh management
 - Lifecycle
 - Workload
 - Performance
 - Configuration
 - Patterns and practices
 - Chaos and filters
- Meshery is about halfway to complete architecture



@AnnieTalvasto

[About Kyverno](#)[Documentation](#)[Policies](#)[Resources](#)[Community](#)[Support](#)

1.7.0 ▾

 Search this site...

Kyverno

Kubernetes Native Policy Management

[Learn More](#)[Get Started](#)

What is Kyverno?

- Designed for Kubernetes
- Policies managed as Kubernetes resources
 - Kubectl
 - Git
 - kustomize
- Validate, mutate and generate



@AnnieTalvasto

Wrap up

CNCF overview

Projects:

- Helm
- Linkerd
- Kudo
- Flux
- Keda
- Meshery



@AnnieTalvasto

Learn more

- CNCF survey -<https://www.cncf.io/reports/cncf-annual-survey-2021/>
- All project sites
- Support your favorite projects in GitHub!
- Case studies:
 - <https://www.cncf.io/case-studies/babylon/>
 - <https://www.cncf.io/case-studies/elkjop/>
- CNCF End User Technology Radar: <https://radar.cncf.io>
- Keynote: Predictions from the Technical Oversight Committee (TOC) - Liz Rice, CNCF TOC Chair
- Techworld with Nana – Youtube
- CNCF Youtube
- [Links and slides: github.com/annietalvasto](https://github.com/annietalvasto)



@AnnieTalvasto

[BACK TO EPISODES](#)

Adventures in open source with Tom Kerkhove

JANUARY 14TH, 2021 | 46:06 | S2:E4

[SHARE](#)[EMBED](#)[RECAST](#)[SUBSCRIBE](#)[DOWNLOAD MP3](#)EPISODE DETAILS / [TRANSCRIPT](#)**EPISODE SUMMARY**

Today's guest on Cloud Gossip is Tom Kerkhove!

Tom works as an Azure Architect at Codit, he's a Github Star, CNCF Ambassador, Azure MVP and he's active as maintainer of Promitor and Keda.

Tom is going to talk to us about how the world of Open-Source projects works, the importance of supporting them, and his personal experience as a maintainer.

EPISODE NOTES

We're going to learn about KEDA and CNCF Sandbox projects, what they are and how they work, and learn about some of Tom's insights in the industry.

He's going to talk about how GitHub is helping the world of Open Source projects and how he uses the platform to engage with the users.



One thing to take away



@AnnieTalvasto