Nikhil Barthwal
Tech Lead – Tools & Infrastructure, Google
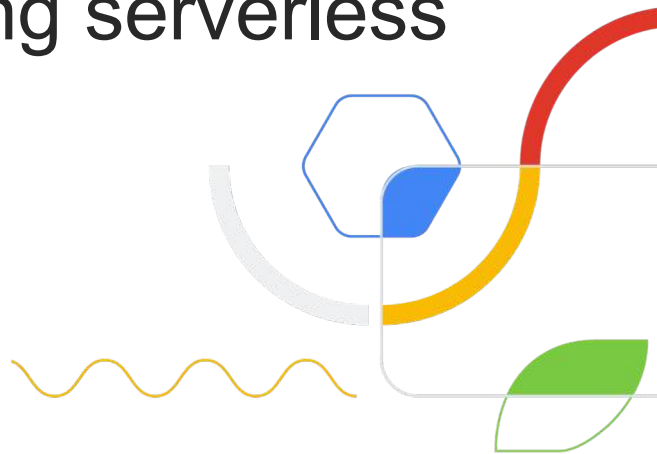
@nikhilbarthwal

nikhilbarthwal@yahoo.com

www.nikhilbarthwal.com

Knative:

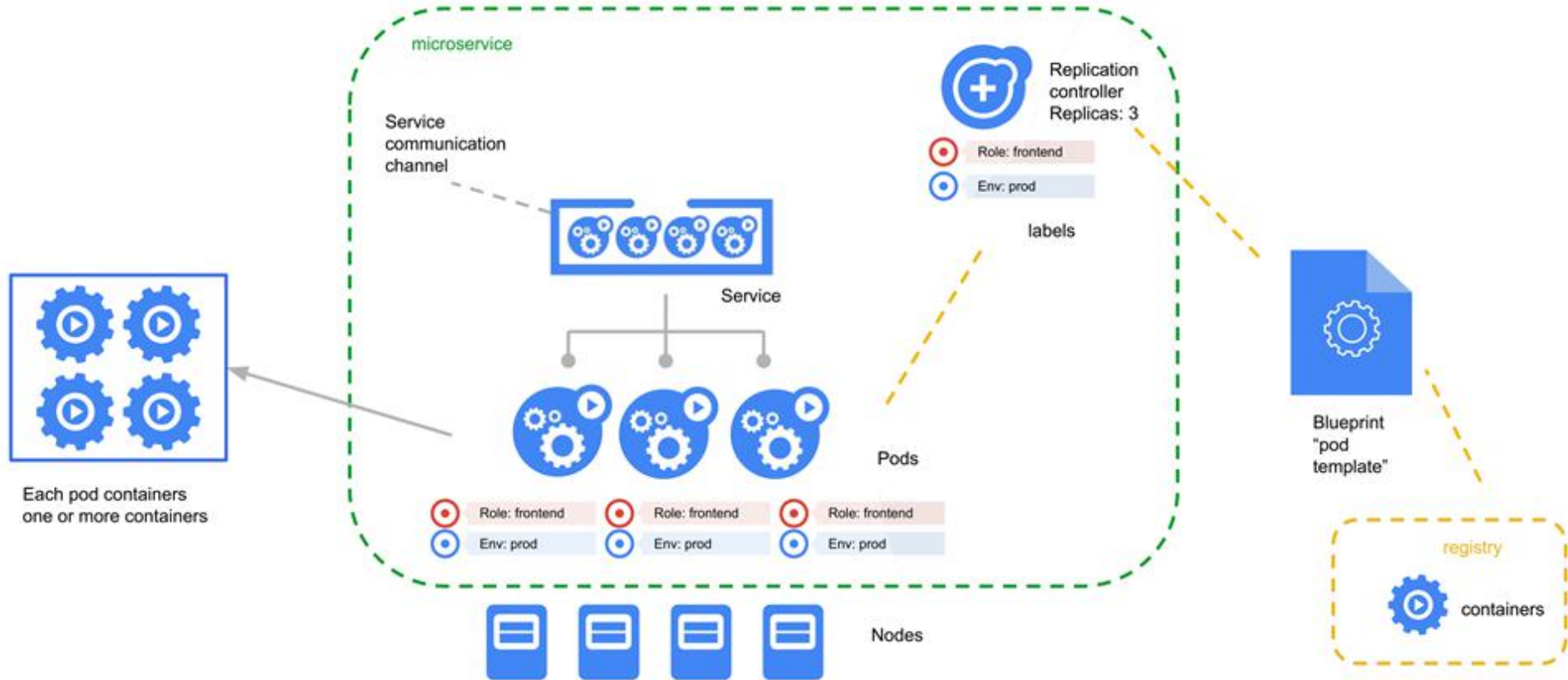Essential Components for building serverless Applications on Kubernetes

Kubernetes: The de facto platform

microservice

Service communication channel

Service

Replication controller Replicas: 3

Role: frontend

Env: prod

labels

Blueprint "pod template"

Pods

Each pod containers one or more containers

Role: frontend Env: prod

Role: frontend Env: prod

Role: frontend Env: prod

registry

containers

Nodes

Kubernetes handles...

Scheduling
Decide where my containers should run

Lifecycle and health
Keep my containers running despite failures

Scaling
Make sets of containers bigger or smaller

Naming and discovery
Find where my containers are now

Load balancing
Distribute traffic across a set of containers

Storage volumes
Provide data to containers

Logging and monitoring
Track what's happening with my containers

Debugging and introspection
Enter or attach to containers

Identity and authorization
Control who can do things to my containers

Kubernetes ecosystem

442 Years of effort*

5,000+ Contributors

47k+ GitHub stars

*Sources: COCOMO Model, CNCF.io Certified Providers

www.nikhilbarthwal.com

The state of affairs!

Developers want serverless

- Just want to run their code

- Want to use their favorite languages and dependencies

- Don't want to manage the infrastructure

Operators want Kubernetes

- Kubernetes is great orchestrating microservices

- They love using GKE and not having to do operations for Kubernetes

- Kubernetes is not the right abstraction for their developers

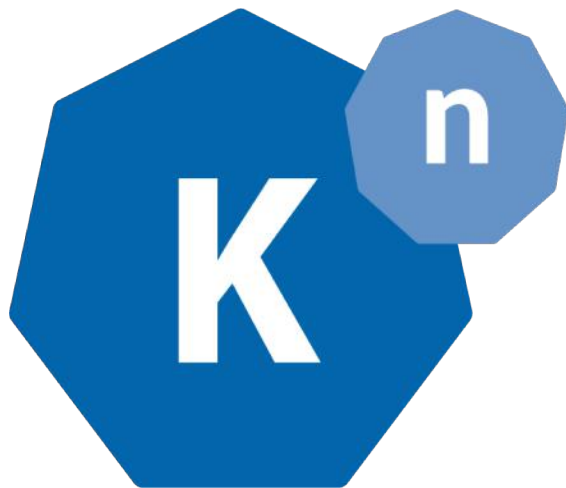# **Serverless:** Characteristics

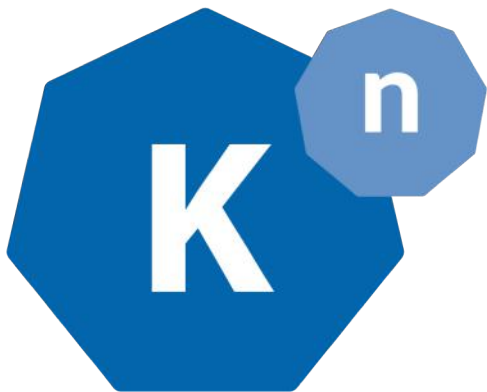Serverless

Event-Driven

Pay as you Go

Portable

Knative

[kay-native]

Kubernetes based open source
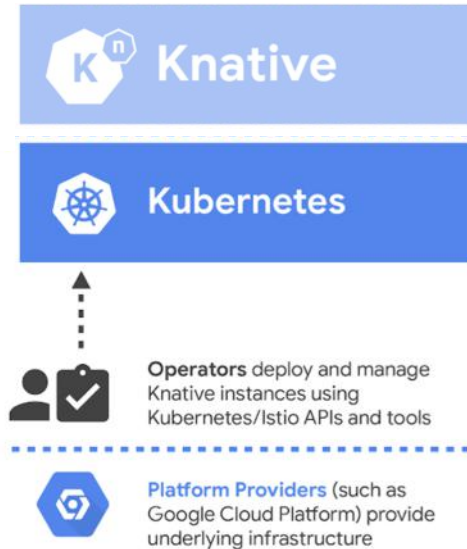building blocks for serverless

Knative project

- Set of primitives (build, events, serving...)
- Solves for modern development patterns
- Implements learnings from Google, partners
- Ingredient or platform for Serverless

github.com/knative
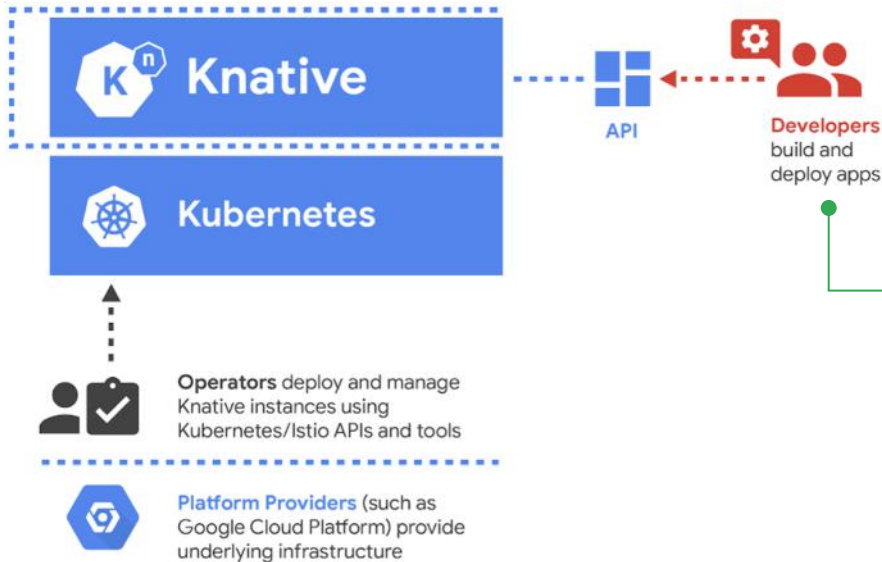
Knative for operator

Abstracts operational complexity, smooth infrastructure surface

Universal supported by all major Cloud providers, enables portability

Extendable platform with clear separation of concerns between operator and developer

Knative for developer

Knative

Kubernetes

API

Developers
build and
deploy apps

Operators deploy and manage
Knative instances using
Kubernetes/Istio APIs and tools

Platform Providers (such as
Google Cloud Platform) provide
underlying infrastructure

Want to
Write code

Don't have to

- Build docker image
- Upload image to registry
- Deploy service
- Expose to the internet
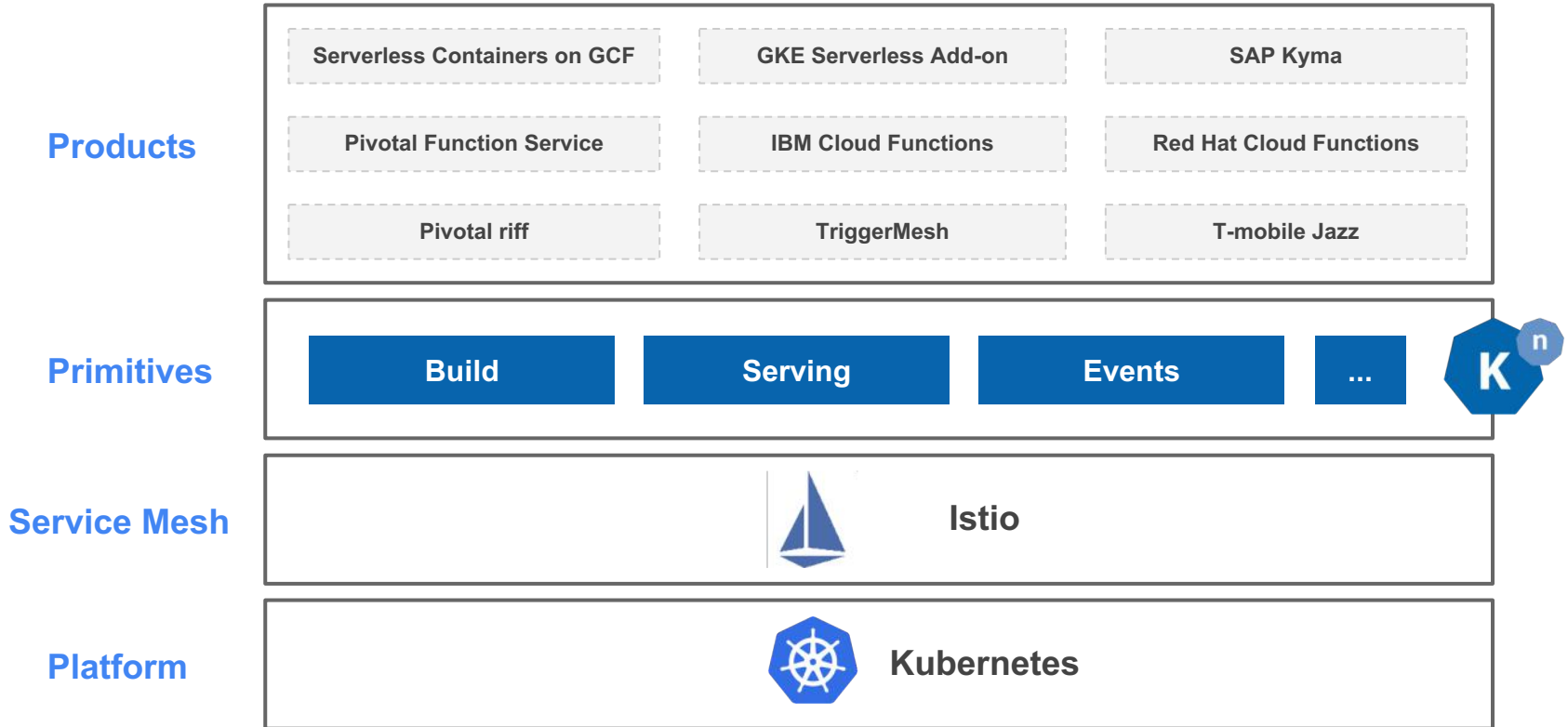- Setup logging & monitoring
- Scale workload...

Knative launch partners

Knative stack

Products	Serverless Containers on GCF	GKE Serverless Add-on	SAP Kyma
	Pivotal Function Service	IBM Cloud Functions	Red Hat Cloud Functions
	Pivotal riff	TriggerMesh	T-mobile Jazz

Primitives	Build · Serving · Events · ...

Service Mesh	Istio

Platform	Kubernetes

Knative impact

"With Kubernetes, SAP Cloud Platform is making it easy for customers to embrace multi-cloud application development and deployment. **Knative further enables SAP to accelerate enterprise-grade serverless and event-based applications.** SAP Kyma is the first open source project being a lighthouse application for SAP on Knative."

Michael Wintergerst
Vice President, SAP

Knative impact

"IBM Cloud is built on open technology, and the future of cloud will be created in the open. Combining what we've learned from the creation of **Apache OpenWhisk**, building serverless projects for enterprises, and our experience with Kubernetes and Istio, we plan to evolve Knative to **bring serverless to a wider range of applications, developers and industries than ever before**."

Jason McGee
IBM Fellow and CTO, IBM Cloud

IBM Cloud

Knative impact

"Red Hat's focus on the hybrid cloud is rooted in customer choice and we carry that same perspective to our work to make hybrid serverless a reality. By joining the Knative community, we are combining our deep enterprise Kubernetes and open source expertise to help **create a common building block for serverless on top of Kubernetes across the hybrid cloud**."

Chris Wright
Red Hat Vice President and Chief Technology Officer

Knative impact

"We believe in providing choice for our internal customers (developers) who are building and deploying cloud native applications. And with Knative, we have been able to rapidly integrate Kubernetes-based building blocks in Jazz, our own open-source serverless toolchain. **Knative and Kubernetes and Jazz are working beautifully together to accelerate T-Mobile's move to cloud native applications**."

Nicolas Criss
Sr. Manager - Cloud Center Of Excellence, T-Mobile

Knative Functionality

- Wakes up workloads (activates) on request
- Scales workloads up/down based on requests (0 pods when no use)
- Turns developer app/function into a container/live URL (transparent build)
- Collects runtime metrics and exports user-space telemetry
- Automatically handles revision of code+config (rolls back if needed)
- Can split traffic and enable gradual rollouts
- Includes eventing system with configurable sources (triggers)
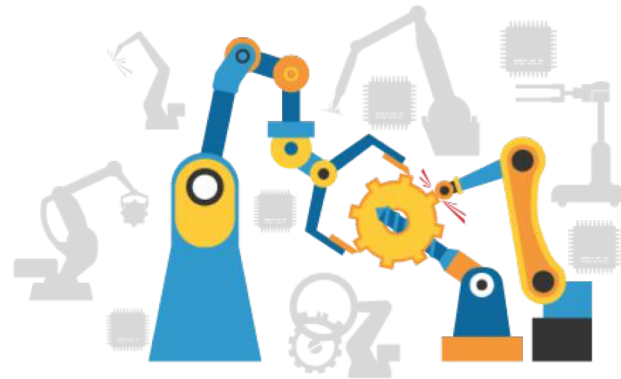
Knative serving

Benefits

- Seamlessly scale up and down

- Built-in traffic splitting between revisions

- Integrates networking and service mesh automatically

- Easy to reason about object model

Pluggable

- Connect to your own logging and monitoring platform, or use the built-in system

- Auto-scaler can be tuned or swapped out for custom code

Serving Primitives

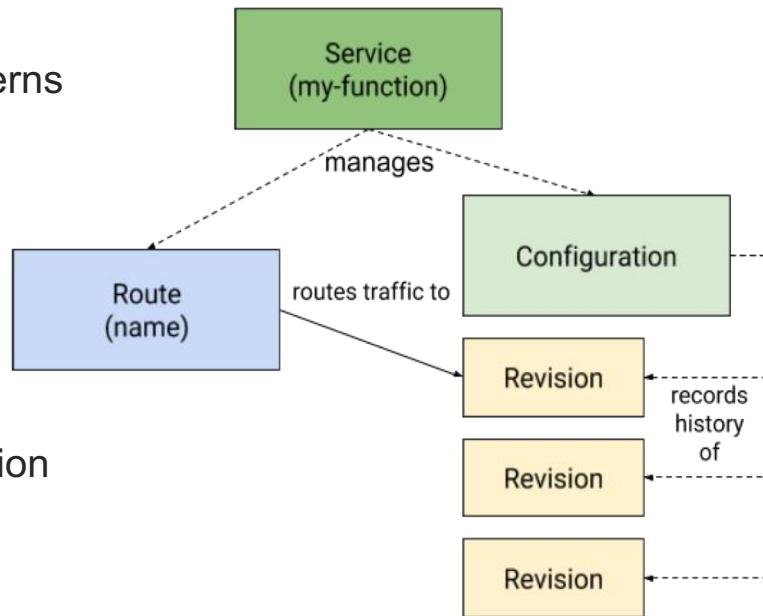Knative defines primitives with clear separation of concerns

Configuration
- Current/desired state for your application
- Code & configuration (separated, ala 12 factor)

Revision
- Point in time snapshots for your code and configuration

Route
- Maps traffic to a revisions
- Supports fractional, named routing

Knative Eventing

What is it?

- For loosely coupled, event-driven services

- Declaratively bind between event producers and Knative services

- Scales from just few events to live streams

- Custom event pipelines to connect with your own existing systems

KubernetesEventSource
GitHubSource
GcpPubSubSource
AwsSqsSource
ContainerSource
CronJobSource

Event type

Event type

Event type

Event source

Flow

Event consumer(s)
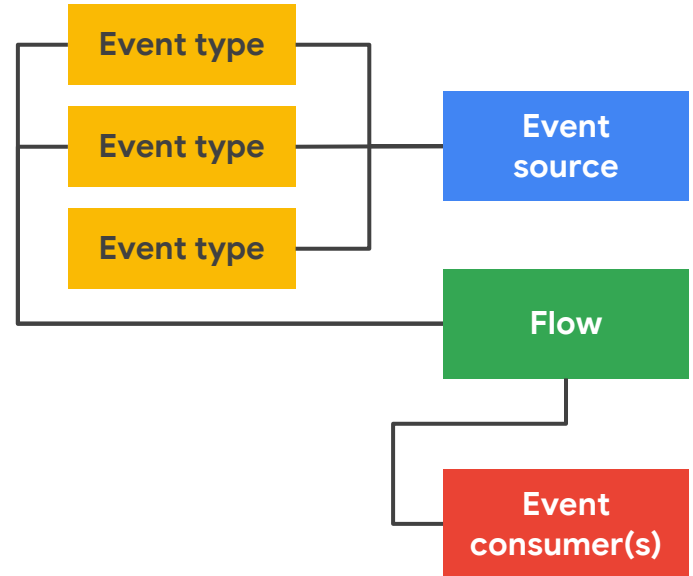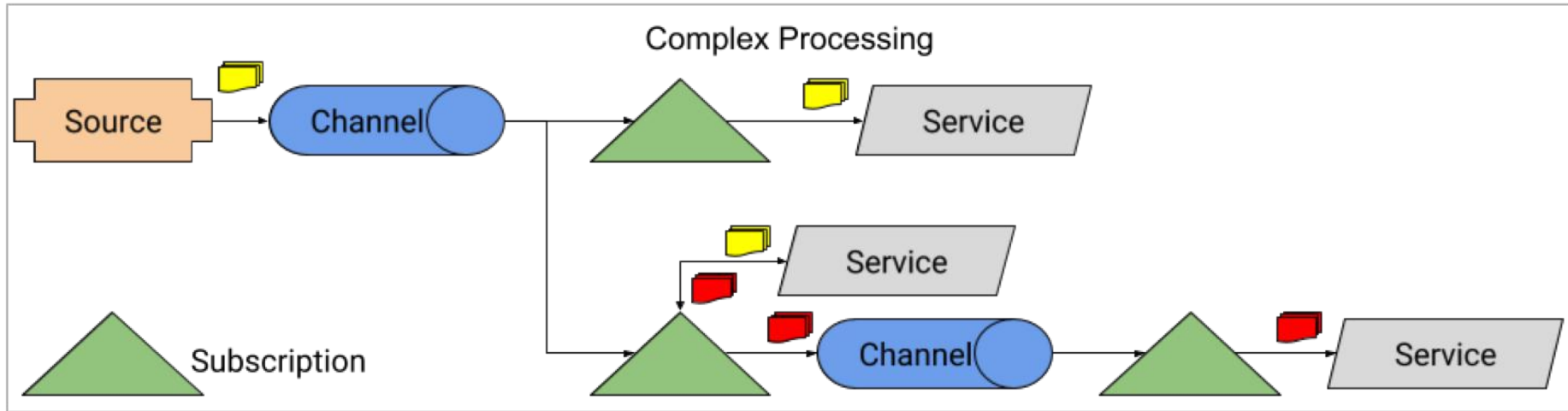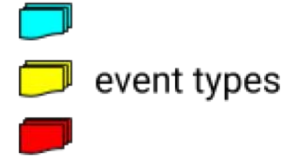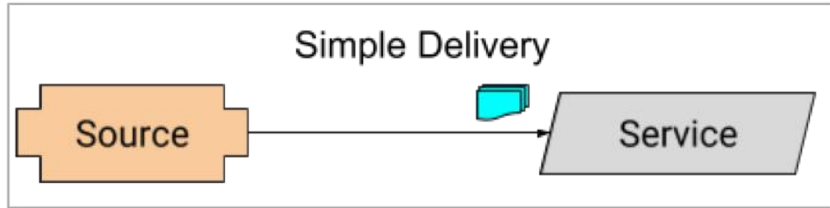
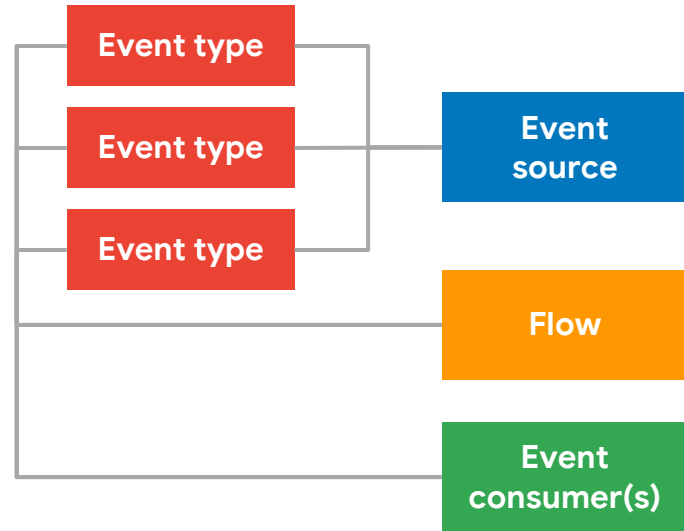# **Knative** Eventing

Knative: Eventing Objects

Eventing constructs :

- **Event Sources** (producer)

- **Channel** (pipelines)

- **Subscription** (configuration)

Warning: volatile development area!

Knative build

Primitives

- **Build:** Represents a single build job with 1 or more steps.
- **BuildTemplate:** A set of ordered and parameterized build steps.
- **ServiceAccount:** For auth with DockerHub etc

What is it?

- Go from source code to container images on repositories
- Build pipelines can have multiple steps and can push to different registries
- Builds run in containers in the cluster. No need for Docker locally

Knative value, today

1-step deploy

Build & deploy with less config/code

Source to container safely in your cluster

Auto-scale

Auto-scale your stateless container based workloads

Scale down to zero

Manage workloads

Automatically deploys containers and provision ingress

Go from source to URL

Knative: Ready for you!

Install, Samples, Docs
github.com/knative/docs

Have questions?
knative.slack.com

Want to contribute?
knative/docs/community

Anything else?
@nikhilbarthwal