Kubernetes, Istio e Knative: The new cloud stack

Luram Archanjo

Who am I?

- Software Engineer at Sensedia

- MBA in Java projects

- Java and Microservice enthusiastic

- Serverless

- Knative & Components

- Why Kubernetes & Istio?

- Demo

- Questions

What is Serverless?

Serverless computing, is an execution model where the **cloud provider** is responsible for executing a **piece of code** by dynamically allocating the resources.
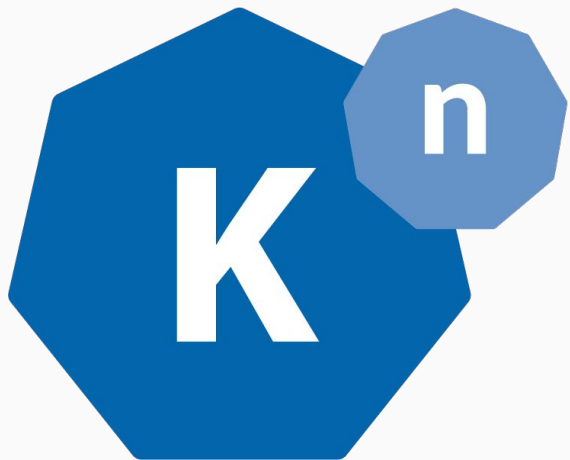
Reduced operational cost

Scalability

Time to market

Source: https://serverless-stack.com & https://martinfowler.com/articles/serverless.html

What is Knative?

Kubernetes based **open source** building blocks for **serverless**

Source: https://knative.dev

Knative Components

Knative Components

Users and Systems (IoT) use applications developed and deployed by developers and hosted by operators

Istio

Knative

Kubernetes

Developers build and deploy apps

API

GitHub

Contributors develop and contribute code and docs to the OSS project

Operators deploy and manage Knative instances using Kubernetes API and tools

Source: https://knative.dev

Knative Components

Serving

How your code receives requests and scales with them:

- Automatic scaling up and down to zero

- Multiple revisions of same function

- Route traffic across revisions

Eventing

How your code is triggered by events:

- Functions consume and publish events

- Multiple event sources available

- Encourages asynchronous

Serving components

Service

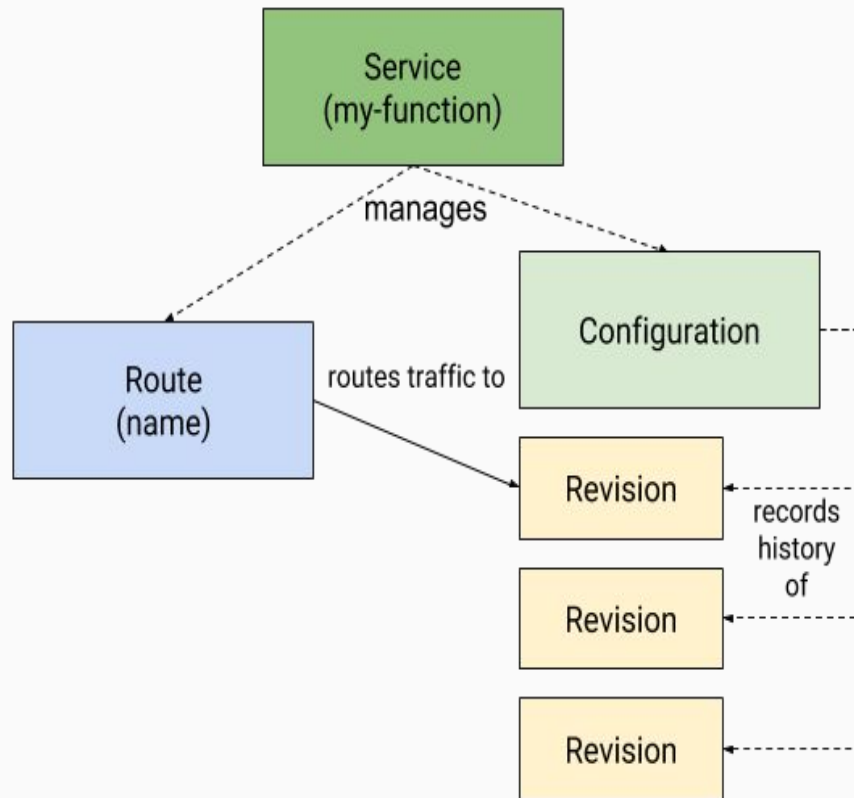Manages the whole lifecycle of your workload.

Configuration

Maintains the current/desired state for your deployment.

Revision

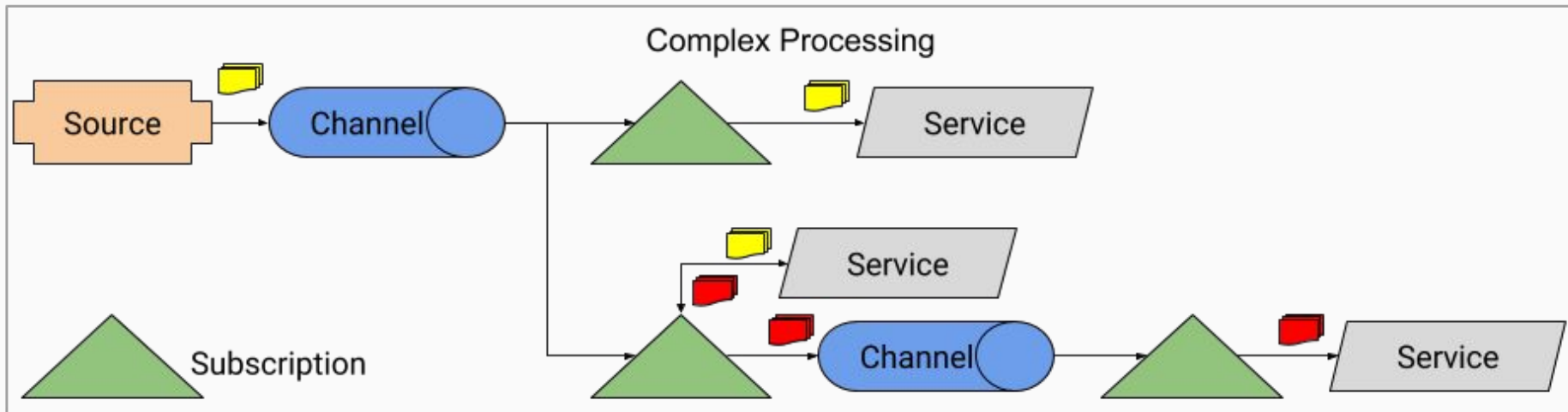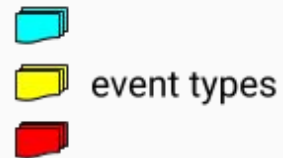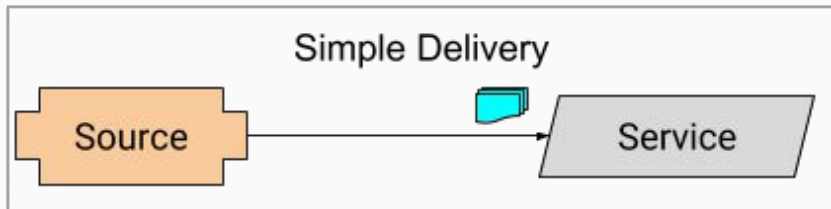A point-in-time snapshot of code and configuration.

Route

Maps and manage traffics to revisions.

Eventing components

How Eventing works?
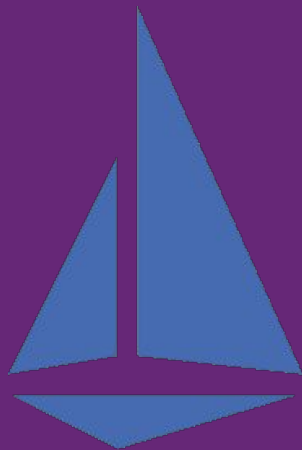

Source: https://knative.dev

Why Kubernetes & Istio?

Kubernetes

- Horizontal scaling

- Auto-scaling

- Service discovery and Load balancing

- Self-healing

Istio

- Traffic management
  - Traffic Routing

- Security
  - East-West
  - North-South

- Observability
  - Metrics
  - Traces
  - Logs

Demo

Summary

2º Place

Flexibility and No Vendor Lock-in

- Cloud
- Premises
- Third-party data center

1º Place

Build on top of Kubernetes and Istio

- Automatic scaling up and down to zero
- Route traffic across revisions
- Load-balancing

3º Place

Serverless Experience in a Containerized Environment

- Immutability
- High Scalability
- Pipeline Management

Thanks a million!
Questions?

 /larchanjo

 /luram-archanjo