OpenShift Serverless & Serverless functions

Naina Singh
Product Manager
Openshift Serverless
@madhatter_ns

Lance Ball
Functions Architect
Openshift Serverless
@lanceball

Red Hat

What is Serverless?

[hint: it's not just functions]

[not just FaaS]   [it's the platform]

Red Hat

Serverless is a Deployment Model

- Scale to/from Zero
- Often event driven
- Distributed
- Stateless

Red Hat

Why Serverless?

Technology is a **utility**.

The **Serverless** approach abstracts complexity delivering on the utility premise.

"Business as usual, speed as usual"

↓ Extensive Architecture and Design

↓ Management and Provisioning

↓ Capacity Planning Sessions

↓ Monoliths and Application Server

The Serverless approach

↑ Simplified approach to Kubernetes

↑ Workloads dynamic scale

↑ Sensible defaults

↑ More experimentation, more granular

Where Serverless?

- Application with unpredictable or bursty number of requests.

- Maximum resource utilization to reduce the carbon footprint

- Building event-driven, loosely coupled systems

- Low barrier for Developers (Kubernetes is hard)

- A/B testing or canary deployments

- Seasonal or periodic workload.

- microservices or containers and want to leverage serverless

The "Serverless Pattern"

Event

trigger

Your Application

produce

Results
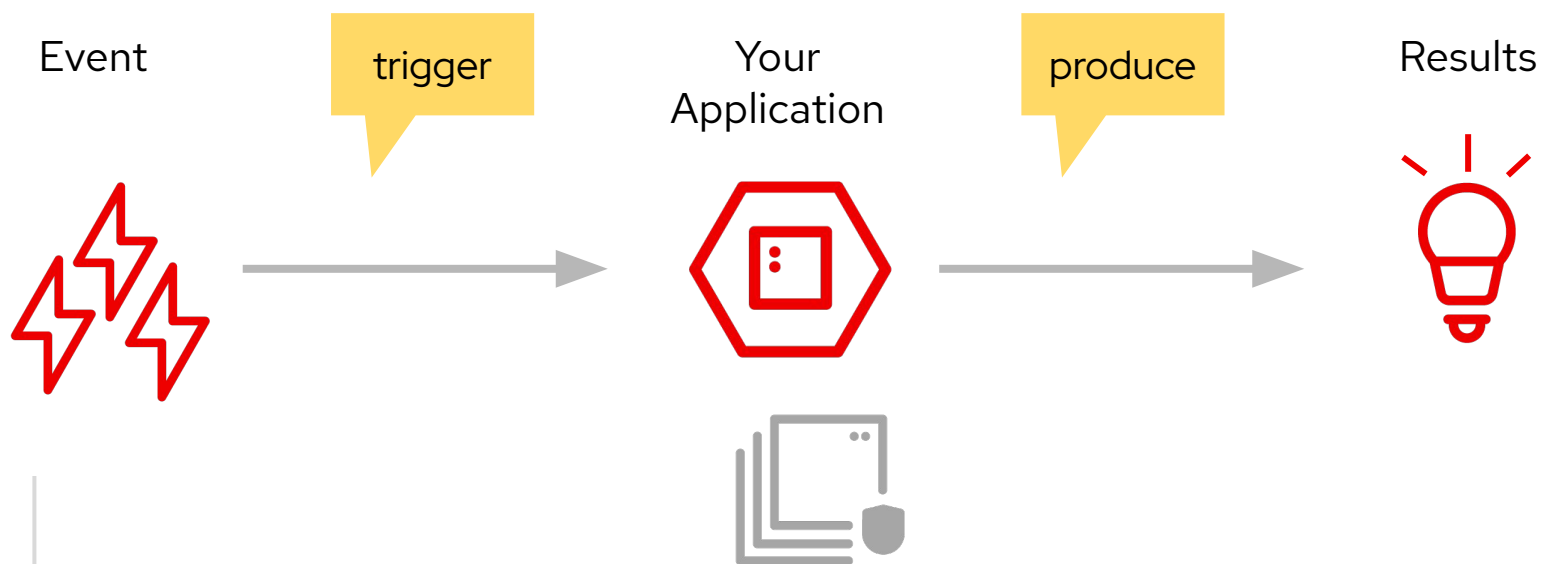
HTTP Requests
Kafka Messages
Image Uploaded
New Order
Login from user

OpenShift Serverless Is Built on Knative

A cloud native execution model for compute resources in which resources are allocated on demand

- Serving
- Eventing
- CLI

Red Hat

Event Driven Applications with CloudEvents

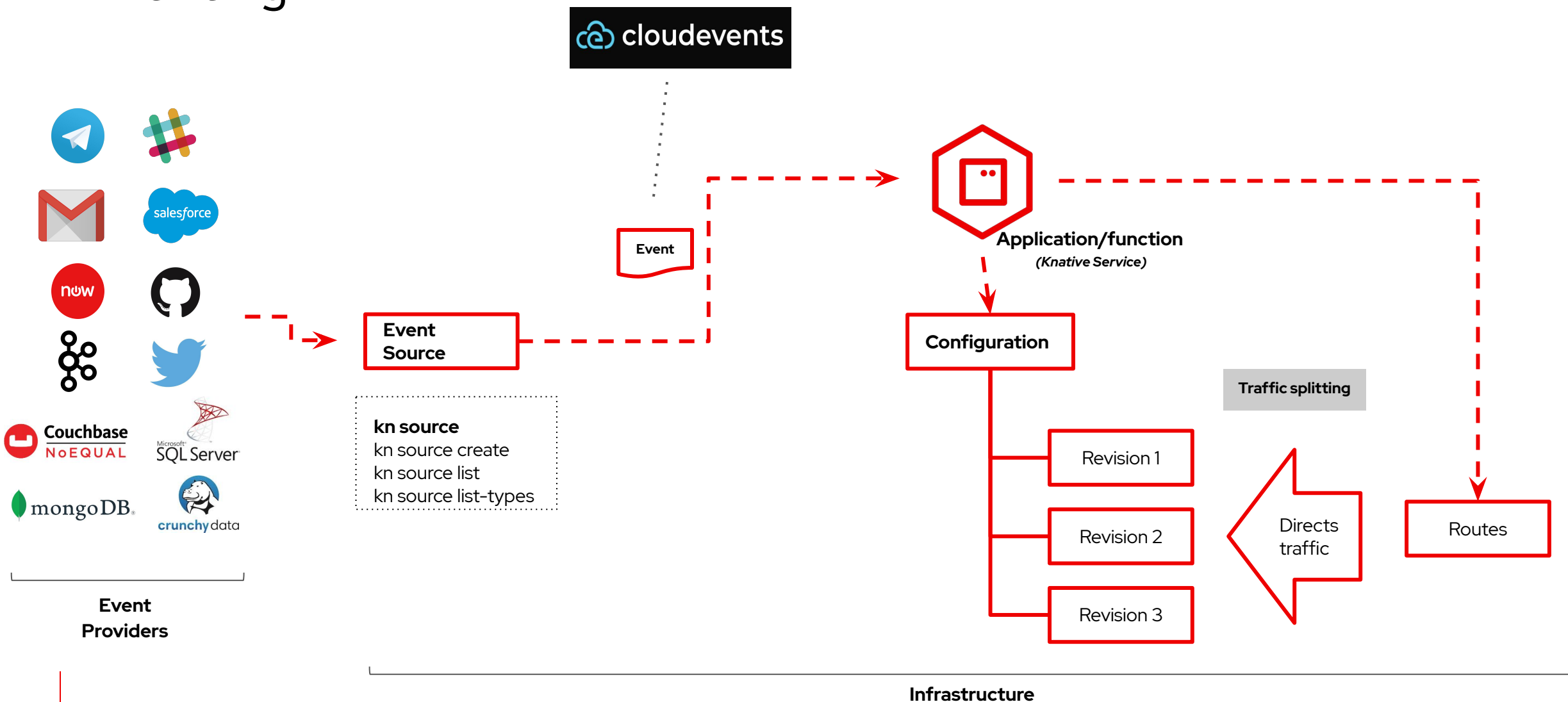**Consistency**   **Accessibility**   **Portability**

https://cloudevents.io/

- CNCF Project
- Backbone for Messaging in Knative Eventing
- Common format for message and event interchange across disparate and heterogeneous services

Red Hat

Eventing

Serverless Functions
[a programming model]

- Single Responsibility

- Event Driven

- Easily Tested

- Cloud Native

- Container Hosted

- Functions are innately serverless, but not all serverless applications are functions

Red Hat

OpenShift Serverless Functions

Augments and extends OpenShift Serverless with Node.js, Python, Quarkus and Go function projects

Functions are deployed as Knative Services and invoked by Knative Event sources or simple HTTP

Red Hat

What consist OpenShift Serverless functions ?

- Buildpacks
- Knative CLI plugin
- Runtimes
- Templates

Red Hat

kn func create

Templates
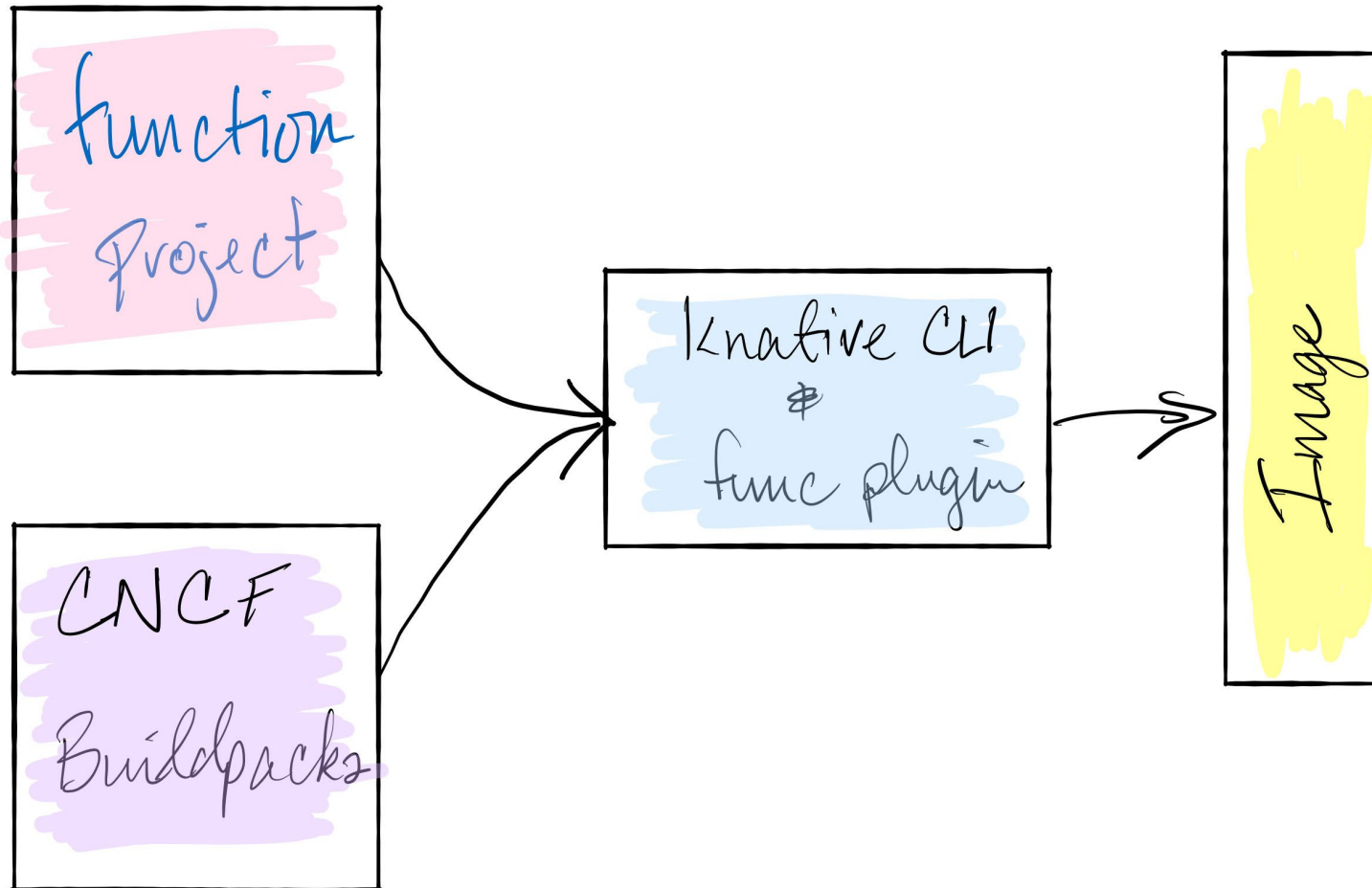
Java, Node
Go

Knative CLI
&
func plugin

Create

Function
Project
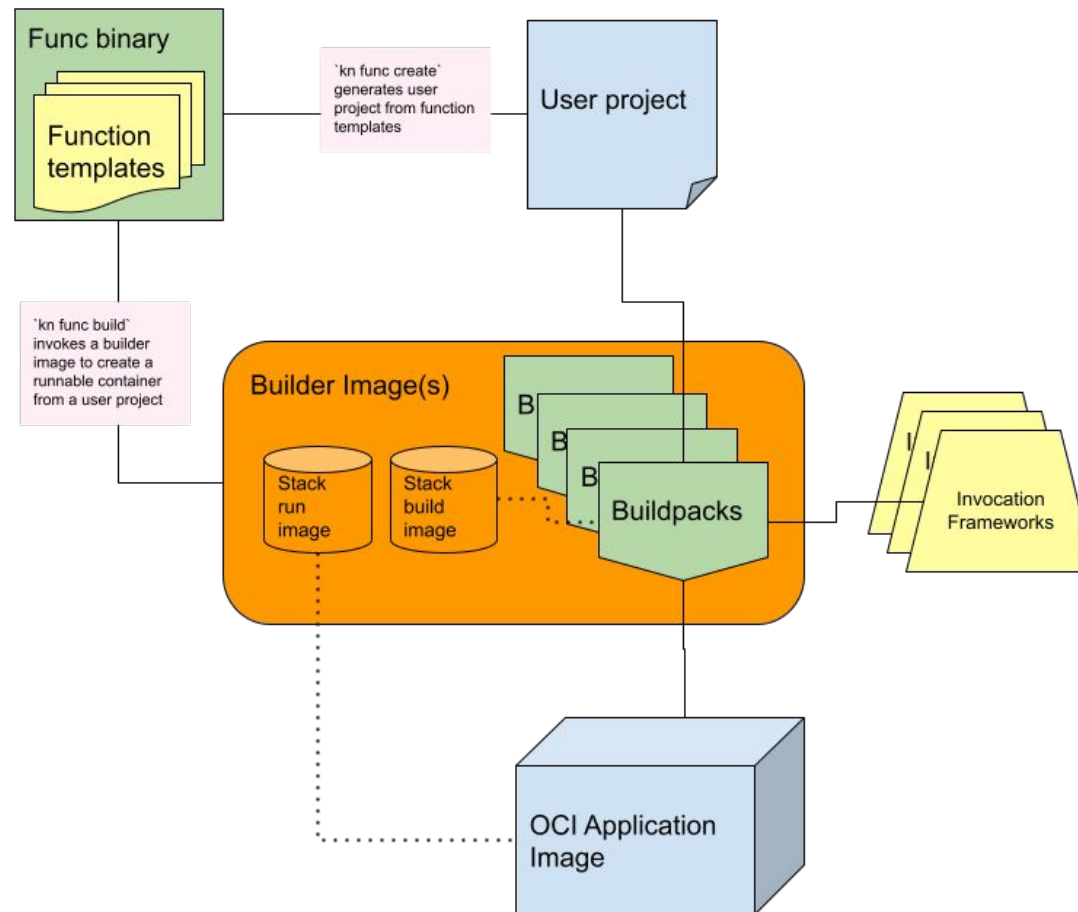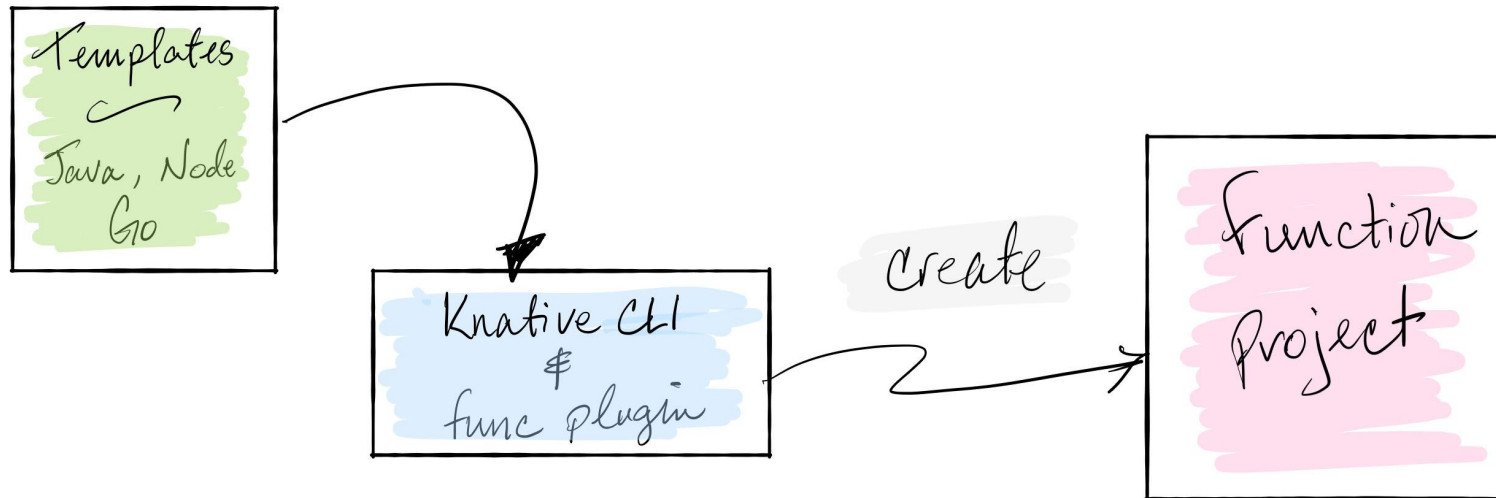
Creating a project

It's really easy. The kn func command will create a skeleton project along with the necessary metadata for Go, Java (Quarkus and SpringBoot), and Node.js
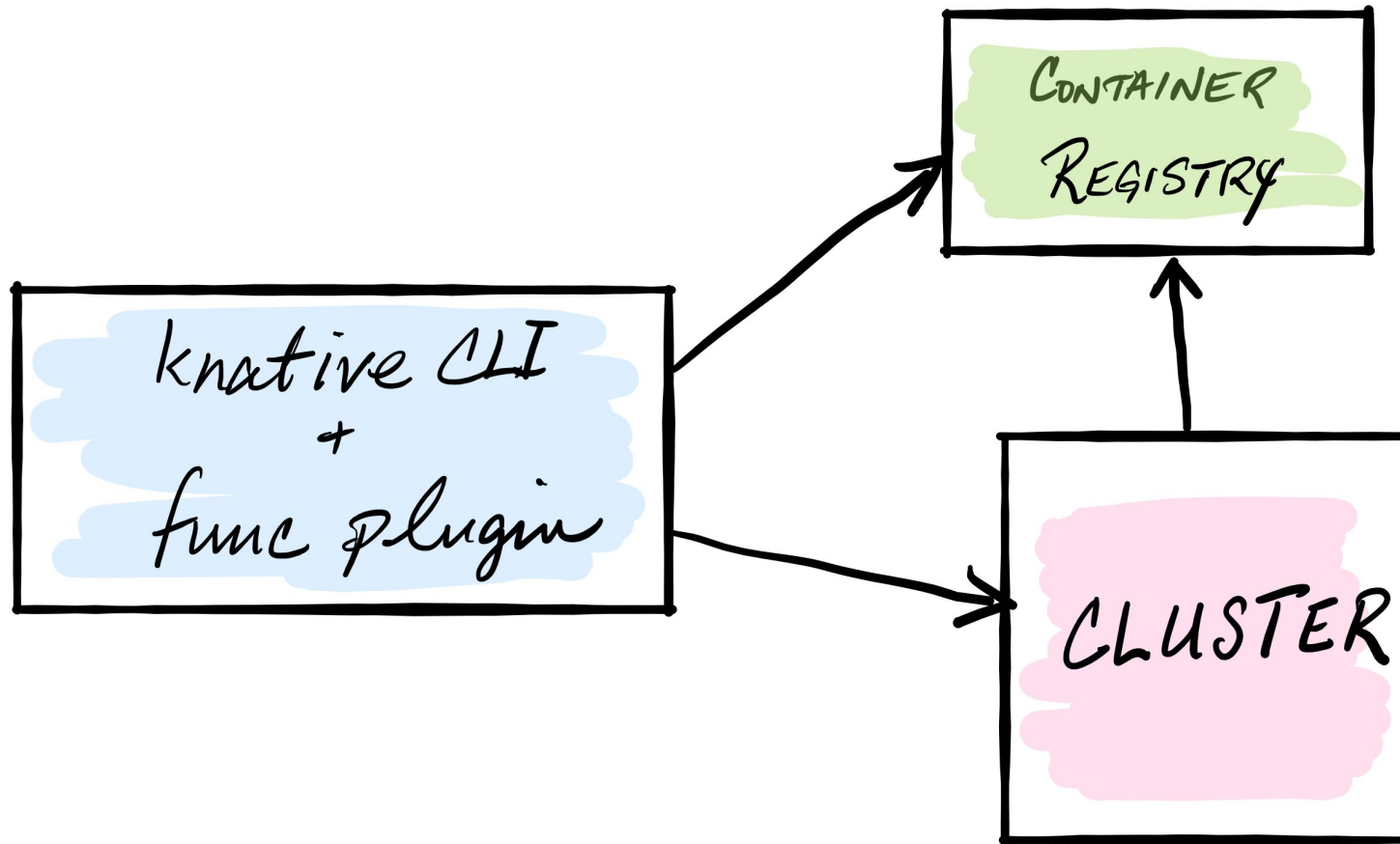
Red Hat

kn func build

function
Project

CNCF
Buildpacks

Knative CLI
&
func plugin

Image

Building a project

Your source code is
combined with a boson
buildpack to create an OCI
container image.

15

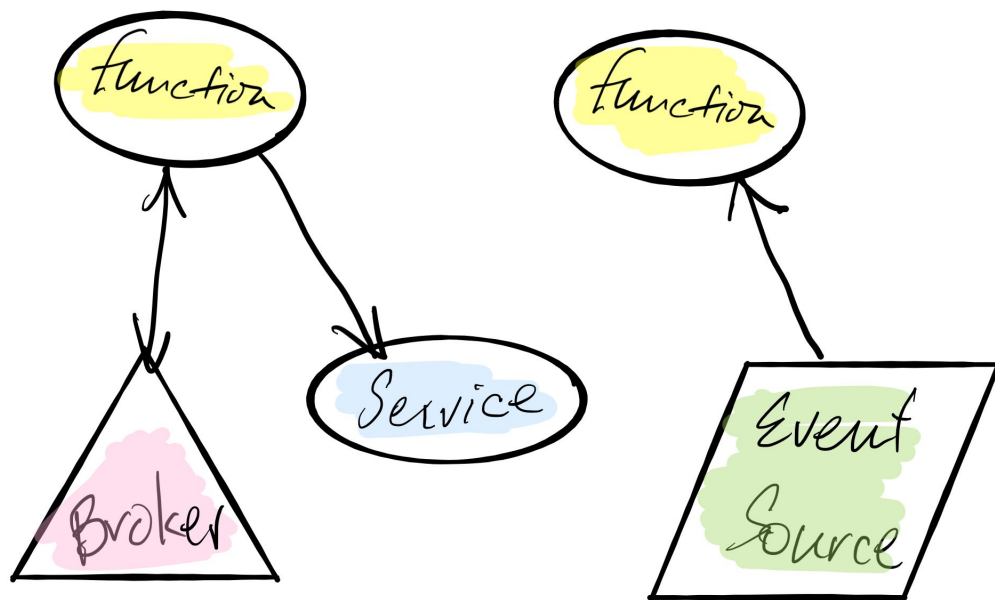Red Hat

kn func deploy

Deploying a project

Using the kn func cli, you can deploy the function to your Kubernetes cluster.

On the cluster

CLUSTER Topology

A Function is a Service

Once your function has been deployed, it's no different in behavior than any other Knative Service

Red Hat

Demo Time

Red Hat

Resources

GitHub

https://github.com/boson-project

GitHub-Knative-Sandbox URL-
Coming soon.

Product Documentation

https://docs.openshift.com/container-platform/4.7/serverless/
functions/serverless-functions-about.html

Telegram Image Analysis Demo

https://www.youtube.com/watch?v=CsYo0SmQ0Uk&t=7s

Red Hat

Q& A

Red Hat