#RedHatOSD

RED HAT FORUMS

IL FUTURO DELL'INTEGRAZIONE

Knative, Camel-K e low code

Giuseppe Boncore	Solution Architect
Nicola Ferraro	Principal Software Engineer
Andrea Tarocchi	Senior Software Engineer

Red Hat

```
apiVersion: redhat/v2
kind: SolutionArchitect
metadata:
   name: Giuseppe Bonocore
   namespace: Italy / FSI
   websites:
     linkedin: giuseppebonocore
     github: bonocore
     twitter: @gbonocore
   annotations:
      specialist: openshift, cloudnative, integration
   labels:
      sports: gym, professional eating
spec:
   replicas: 1
   containers:
         - image: redhat.io/giuseppe:latest
```

APPLICATION RUNTIMES

VERT.X
node
MICROPROFILE
Java EE
THORNTAIL

RED HAT DATA GRID

OpenJDK

RED HAT AMQ BROKER

COMPREHENSIVE TOOLS TO BUILD & MIGRATE APPS

INTEGRATION

RED HAT FUSE

RED HAT AMQ

RED HAT 3SCALE API MANAGEMENT

COMPOSE AND INTEGRATE MICROSERVICES ACROSS AN ENTERPRISE SERVICE NETWORK

PROCESS AUTOMATION

RED HAT PROCESS AUTOMATION MANAGER

RED HAT DECISION MANAGER

AUTOMATE AND OPTIMIZE BUSINESS PROCESSES

APPLICATION SERVICES

SERVICE MESH

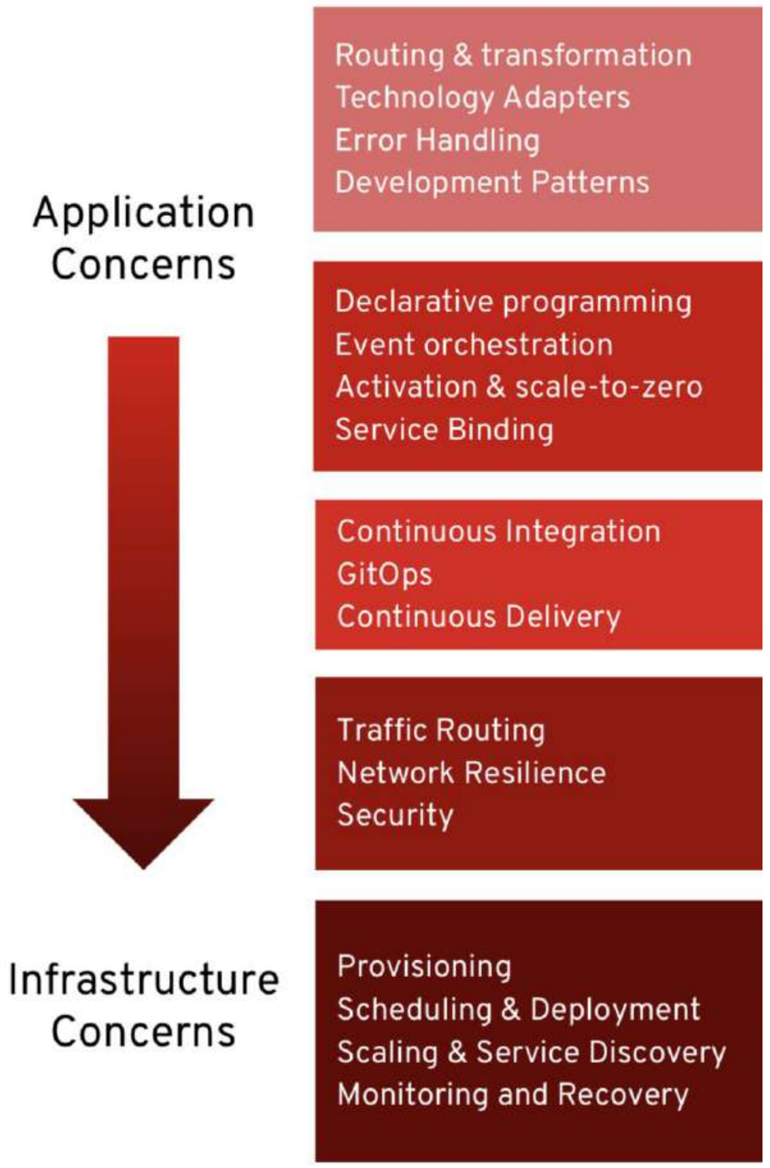
ENTERPRISE KUBERNETES

RED HAT OPENSHIFT

Serverless Defined

“Serverless computing refers to a new model of cloud native computing, enabled by architectures that ***do not require server management*** *to build and run applications. "*

CLOUD NATIVE
COMPUTING FOUNDATION



FAAS (*AAS)

SERVERLESS

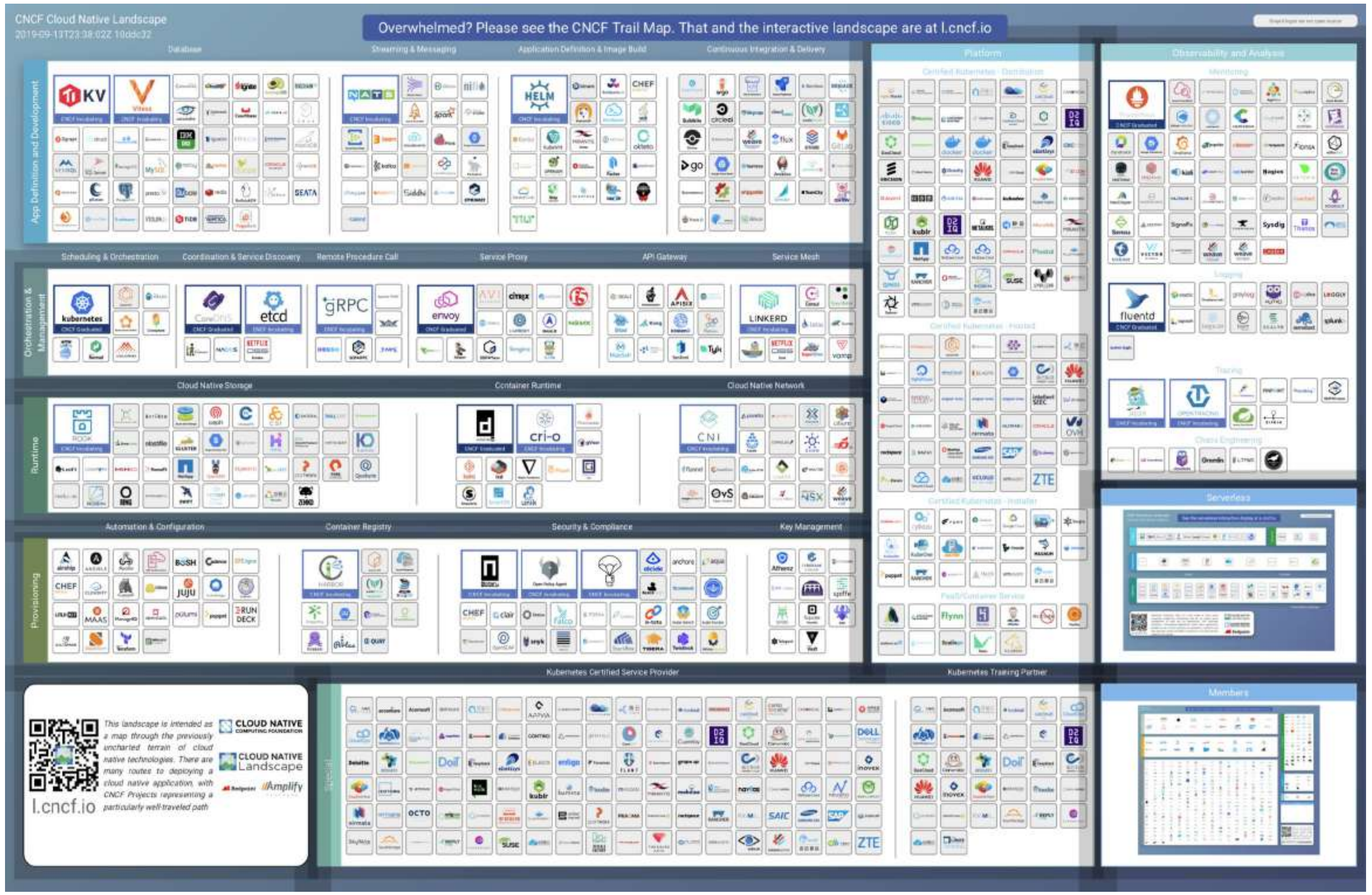
BUILD & DEPLOY

SERVICE MESH

ORCHESTRATION

Red Hat

#RedHatOSD


CNCF Cloud Native Landscape
Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io
Database
Streaming & Messaging
Application Definition & Image Build
Continuous Integration & Delivery
Platform
Observability and Analysis
App Definition and Development
Scheduling & Orchestration
Coordination & Service Discovery
Remote Procedure Call
Service Proxy
API Gateway
Service Mesh
Orchestration & Management
Cloud Native Storage
Container Runtime
Cloud Native Network
Runtime
Automation & Configuration
Container Registry
Security & Compliance
Key Management
Provisioning
Kubernetes Certified Service Provider
Kubernetes Training Partner
Serverless
Members
l.cncf.io
This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path
CLOUD NATIVE COMPUTING FOUNDATION
CLOUD NATIVE Landscape

#RedHatOSD

Application Concerns

Infrastructure Concerns

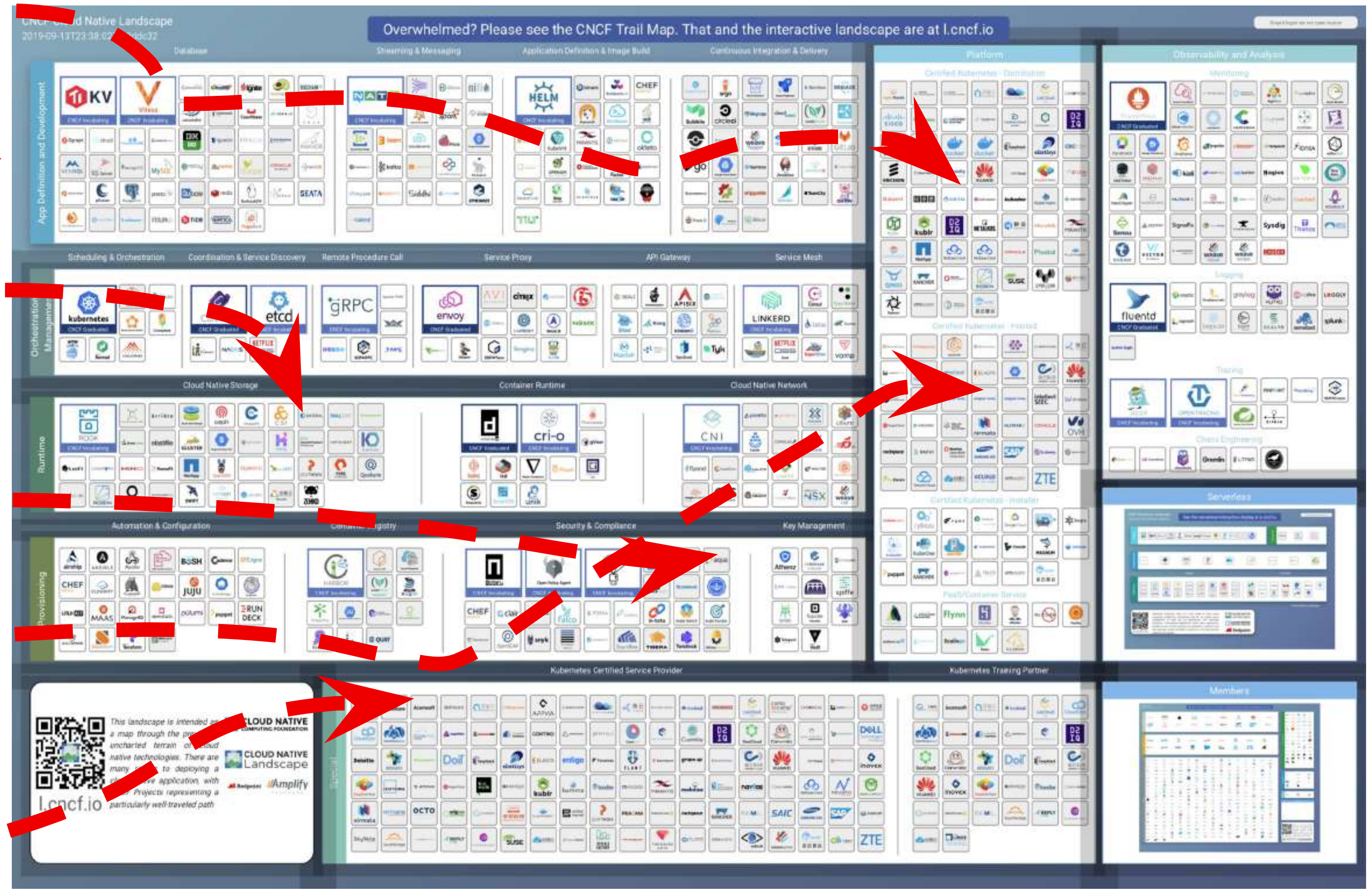
Routing & transformation
Technology Adapters
Error Handling
Development Patterns

Declarative programming
Event orchestration
Activation & scale-to-zero
Service Binding

Continuous Integration
GitOps
Continuous Delivery

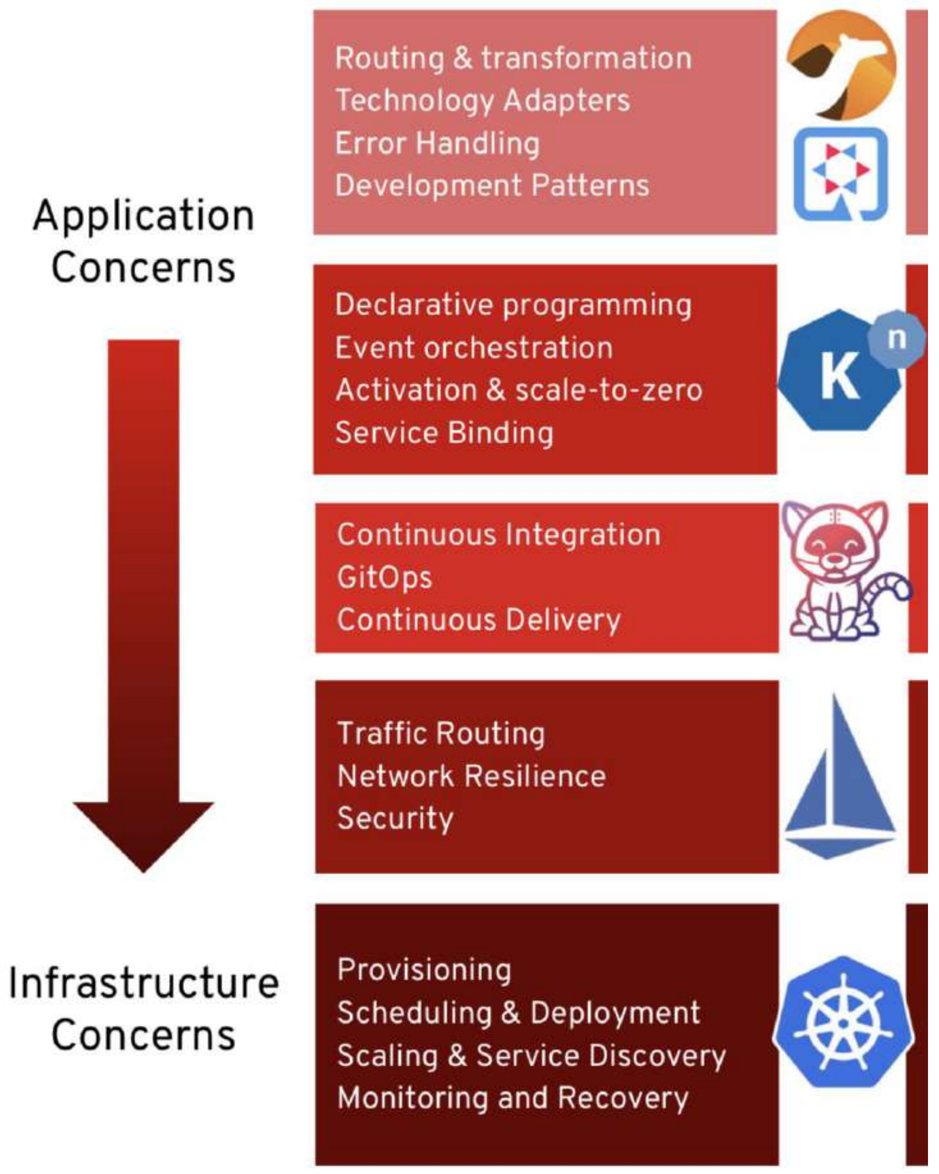
Traffic Routing
Network Resilience
Security

Provisioning
Scheduling & Deployment
Scaling & Service Discovery
Monitoring and Recovery


Cloud Native Landscape
Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io
l.cncf.io

Red Hat

#RedHatOSD



FAAS (*AAS)

SERVERLESS

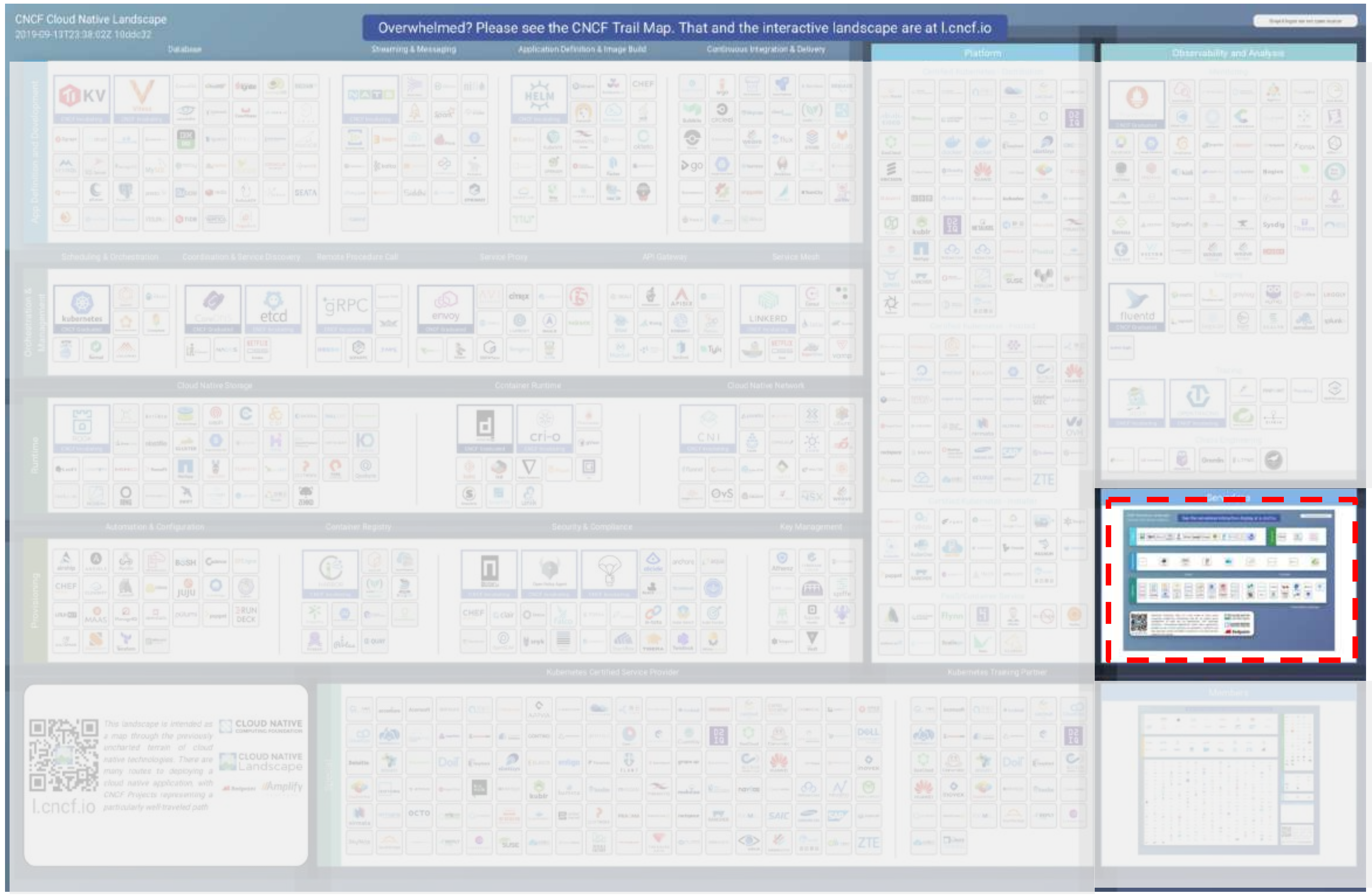
BUILD & DEPLOY

SERVICE MESH

ORCHESTRATION

Red Hat

#RedHatOSD


CNCF Cloud Native Landscape
2019-09-13T23:38:02Z 10ddc32

Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io

This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path.

Red Hat

#RedHatOSD

Tools

- Cloud Native Landscape
- Cloudzero
- dashbird
- epsagon
- HASURA GraphQL Engine
- IO|pipe
- Iron.io
- lumigo
- Node Lambda
- SCAR
- SIGMA
- STACKERY
- THUNDRA

Security

- Protego
- threat stack

Framework

- APEX
- Architect
- AWS SAM
- CHALICE
- Claudia.js
- dapr
- FLOGO
- serverless
- SPARTA
- Spring Cloud Function

Platform

Hosted

- ALGORITHMIA
- netlify Functions
- Alibaba Cloud Function Compute
- nimbella
- AWS Lambda
- Nuweba
- Azure Functions
- PubNub Functions
- BINARIS
- spotinst
- CLOUDFLARE Workers
- Standard Library
- Google Cloud Functions
- Tencent Cloud
- HUAWEI FunctionStage
- twilio Functions
- IBM Cloud Functions
- ZEIT

Installable

- Apache Camel
- Kyma
- Apache OpenWhisk
- nuclio
- AppScale
- OPENFAAS
- fission
- PipelineAI
- fn
- riff
- Knative
- Virtual Kubelet
- Kubeless

Cloud Native Landscape

s.cncf.io

Serverless computing refers to a new model of cloud native computing, enabled by architectures that do not require server management to build and run applications. This landscape illustrates a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform and then executed, scaled, and billed in response to the exact demand needed at the moment

CLOUD NATIVE Landscape

CLOUD NATIVE COMPUTING FOUNDATION

Redpoint

Tools

Cloud Native Landscape, Cloudzero, dashbird, epsagon, HASURA GraphQL Engine, IO|pipe, Iron.io, lumigo, Node Lambda, SCAR, SIGMA, STACKERY, THUNDRA

Security

Protego, threat stack

Framework

APEX, Architect, AWS SAM, CHALICE, Claudia.js, dapr, FLOGO, serverless, SPARTA, Spring Cloud Function

Hosted

Installable

Platform

ALGORITHMIA, Alibaba Cloud Function Compute, AWS Lambda, Azure Functions, BINARIS, CLOUDFLARE Workers, Google Cloud Functions, HUAWEI FunctionStage, IBM Cloud Functions, netlify Functions, nimbella, Nuweba, PubNub Functions, spotinst, Standard Library, Tencent Cloud, twilio Functions, ZEIT

Camel, Apache OpenWhisk, AppScale, fission, fn, Knative, Kubeless, Kyma, nuclio, OPENFAAS, PipelineAI, riff, Virtual Kubelet

Cloud Native Landscape

s.cncf.io

Serverless computing refers to a new model of cloud native computing, enabled by architectures that do not require server management to build and run applications. This landscape illustrates a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform and then executed, scaled, and billed in response to the exact demand needed at the moment

CLOUD NATIVE Landscape

CLOUD NATIVE COMPUTING FOUNDATION

Redpoint

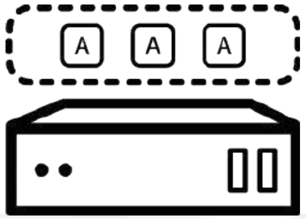
What is Knative ?

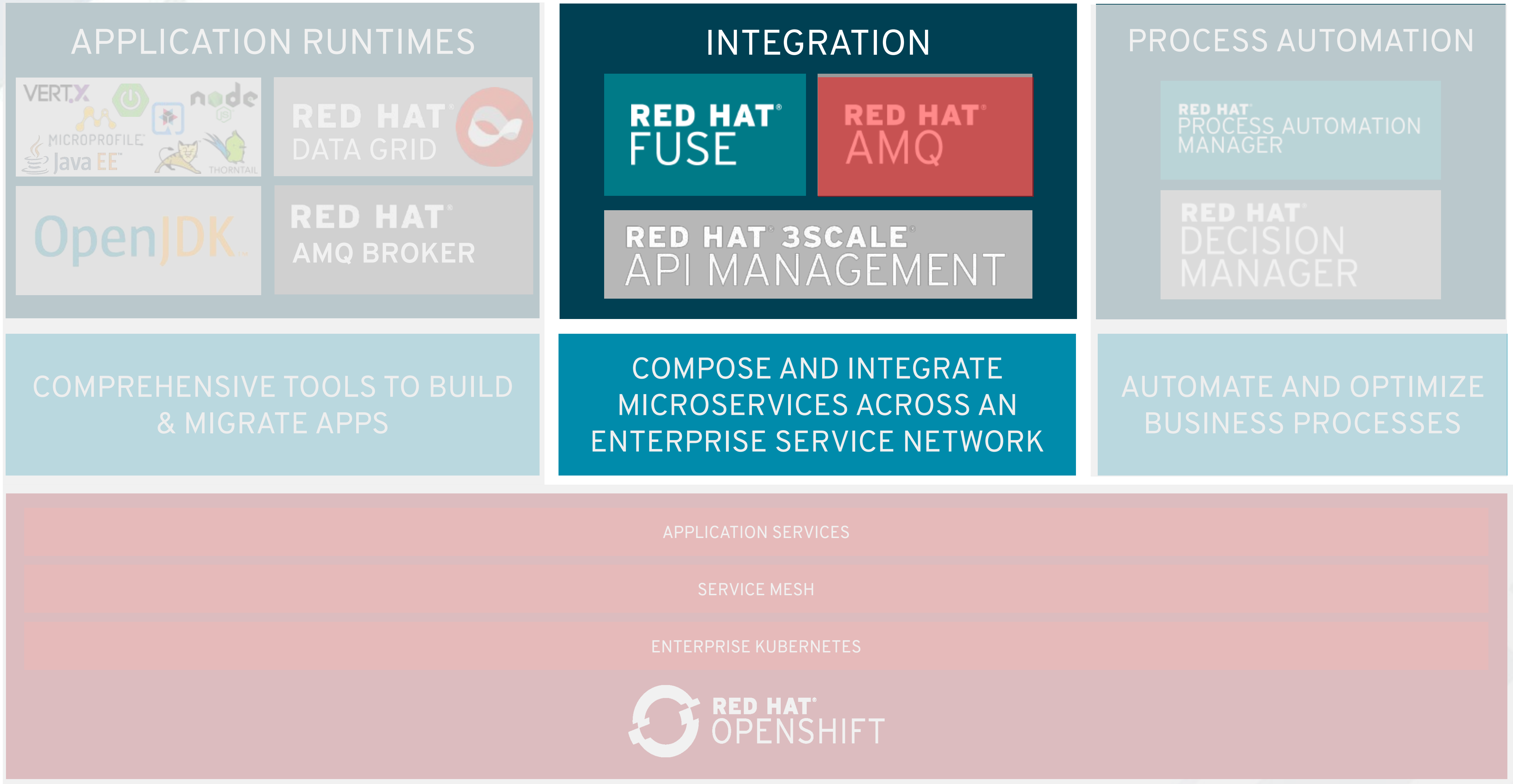
SERVING

An event-driven model that serves the container with your application and can "scale to zero".

EVENTING

Common infrastructure for consuming and producing events that will stimulate applications.




APPLICATION RUNTIMES
VERT.X
node
MICROPROFILE
Java EE
THORNTAIL
RED HAT DATA GRID
OpenJDK
RED HAT AMQ BROKER
INTEGRATION
RED HAT FUSE
RED HAT AMQ
RED HAT 3SCALE API MANAGEMENT
PROCESS AUTOMATION
RED HAT PROCESS AUTOMATION MANAGER
RED HAT DECISION MANAGER
COMPREHENSIVE TOOLS TO BUILD & MIGRATE APPS
COMPOSE AND INTEGRATE MICROSERVICES ACROSS AN ENTERPRISE SERVICE NETWORK
AUTOMATE AND OPTIMIZE BUSINESS PROCESSES
APPLICATION SERVICES
SERVICE MESH
ENTERPRISE KUBERNETES
RED HAT OPENSHIFT

Broker

- Store & Forward
- Volatile & Durable
- Full JMS 2.0 Support
- Best-in-class performances

Interconnect

- High-performance direct messaging
- Distributed messaging backbone

Streams

- Streaming platform
- Durable pub/sub
- Replayable streams
- Based on Apache Kafka project

AMQ Online

- Scalable, easy-to-manage messaging utility based on OpenShift container platform
- Enable developers to provision messaging infrastructure through a web console.

Broker

Apache ACTIVEMQ

Interconnect

Qpid

Streams

APACHE kafka

AMQ Online

enmasse

AMQ Streams: What's new

AMQ Streams 1.3.0 (Released Oct 2019)

- Kafka 2.3.0 support
- Kafka Bridge move to GA
- Kafka Exporter
- Change Data Capture (Tech Preview)

https://www.flickr.com/photos/orreip/

APPLICATION RUNTIMES

VERT.X · node · MICROPROFILE · Java EE · THORNTAIL

RED HAT DATA GRID

OpenJDK

RED HAT AMQ BROKER

COMPREHENSIVE TOOLS TO BUILD & MIGRATE APPS

INTEGRATION

RED HAT FUSE

RED HAT AMQ

RED HAT 3SCALE API MANAGEMENT

COMPOSE AND INTEGRATE MICROSERVICES ACROSS AN ENTERPRISE SERVICE NETWORK

PROCESS AUTOMATION

RED HAT PROCESS AUTOMATION MANAGER

RED HAT DECISION MANAGER

AUTOMATE AND OPTIMIZE BUSINESS PROCESSES

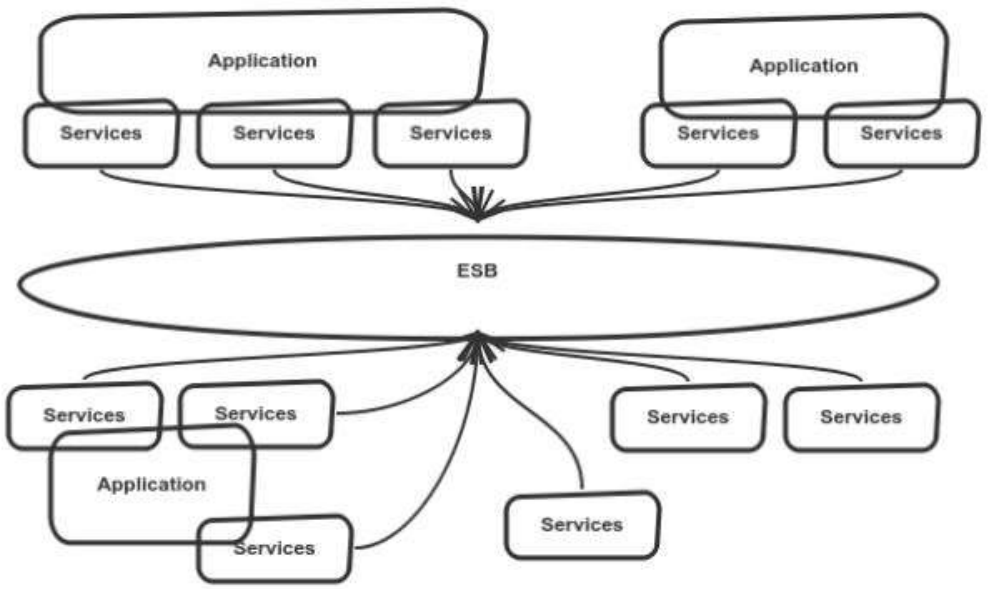
APPLICATION SERVICES

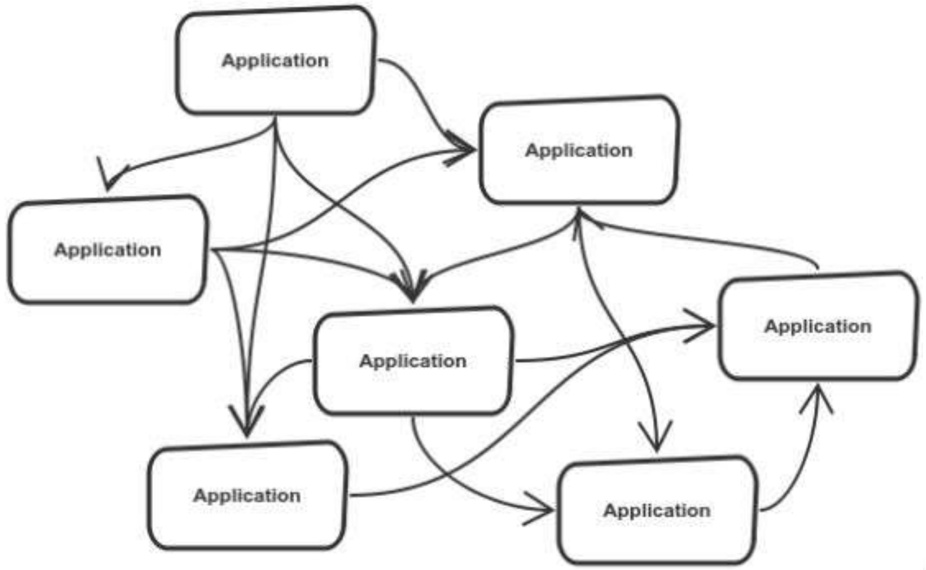
SERVICE MESH

ENTERPRISE KUBERNETES

RED HAT OPENSHIFT

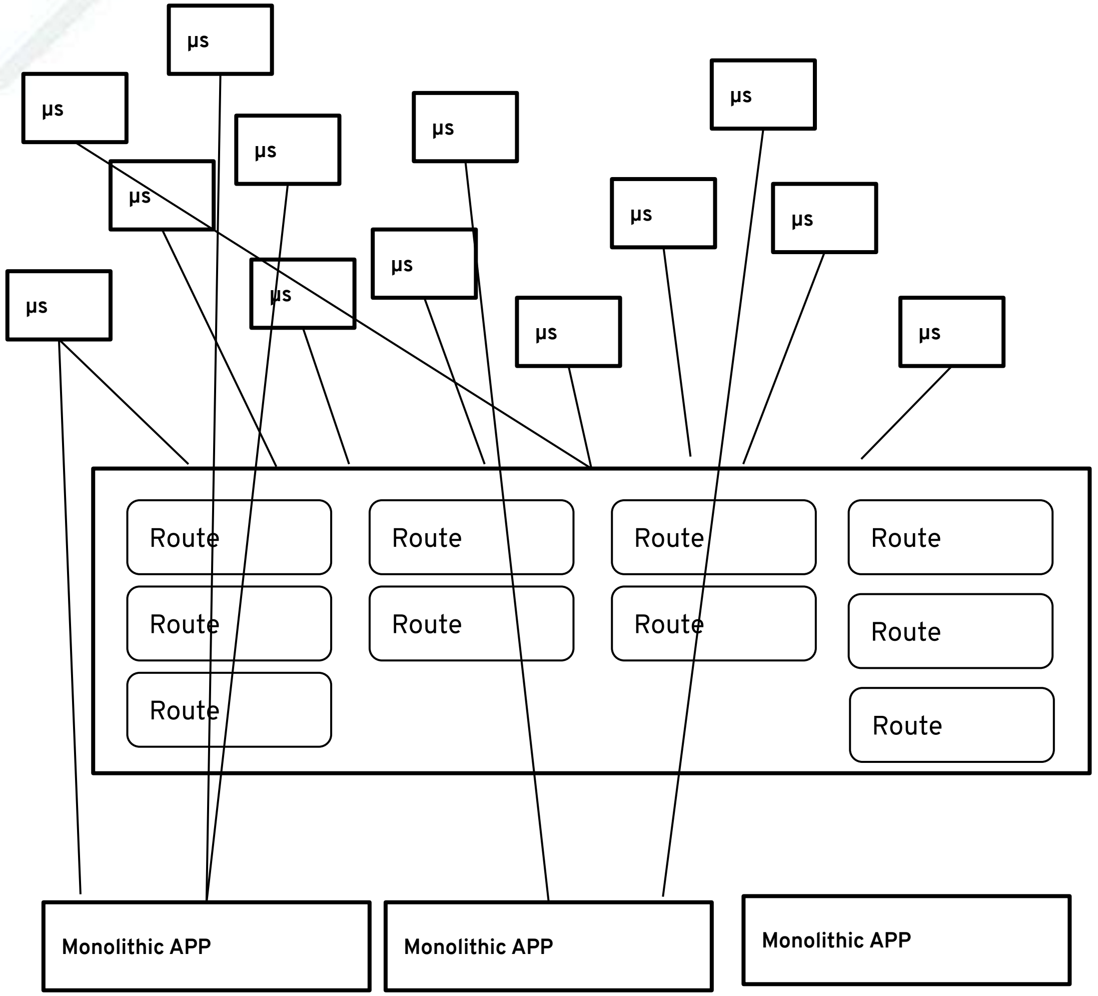
Is Integration still relevant, in a μSVCs world?

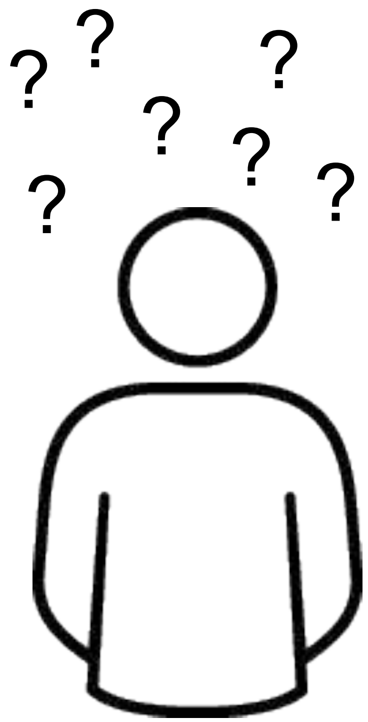


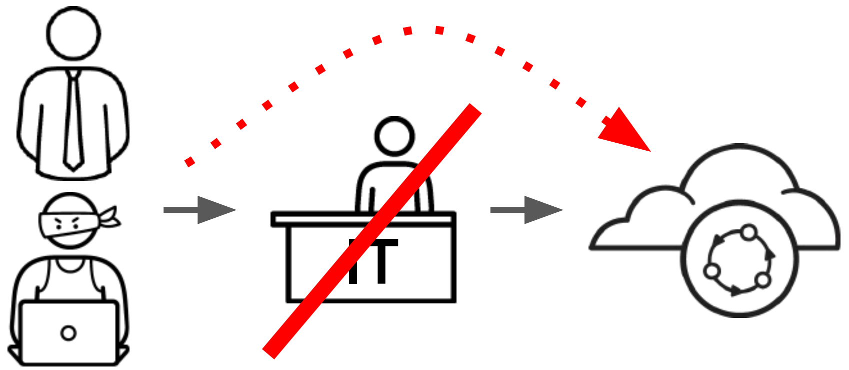


Don’t let integration get buried in your architecture!

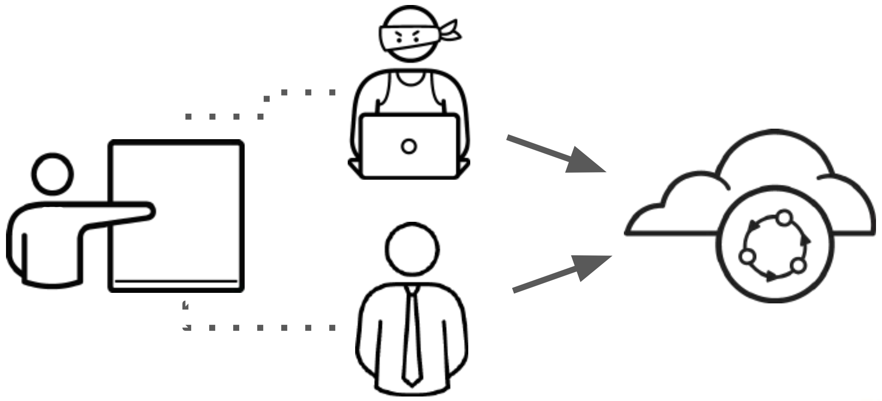
https://www.flickr.com/photos/hodgers/


µs
µs
µs
µs
µs
µs
µs
µs
µs
µs
µs
µs
µs
µs
µs
Route
Route
Route
Route
Route
Route
Route
Route
Route
Route
Monolithic APP
Monolithic APP
Monolithic APP


?
?
?
?
?
?
?



Central IT as Gatekeeper



Central IT as Advisor

His royal majesty, Apache Camel

http://camel.apache.org

The swiss knife of integration

>10 years of development - still one of the most active Apache projects

Based on Enterprise Integration Patterns (EIP)

Uses a powerful Domain Specific Language (DSL)

Can integrate anything

Supports 300+ components

What is Camel K?

A lightweight integration platform, born on Kubernetes, with serverless superpowers

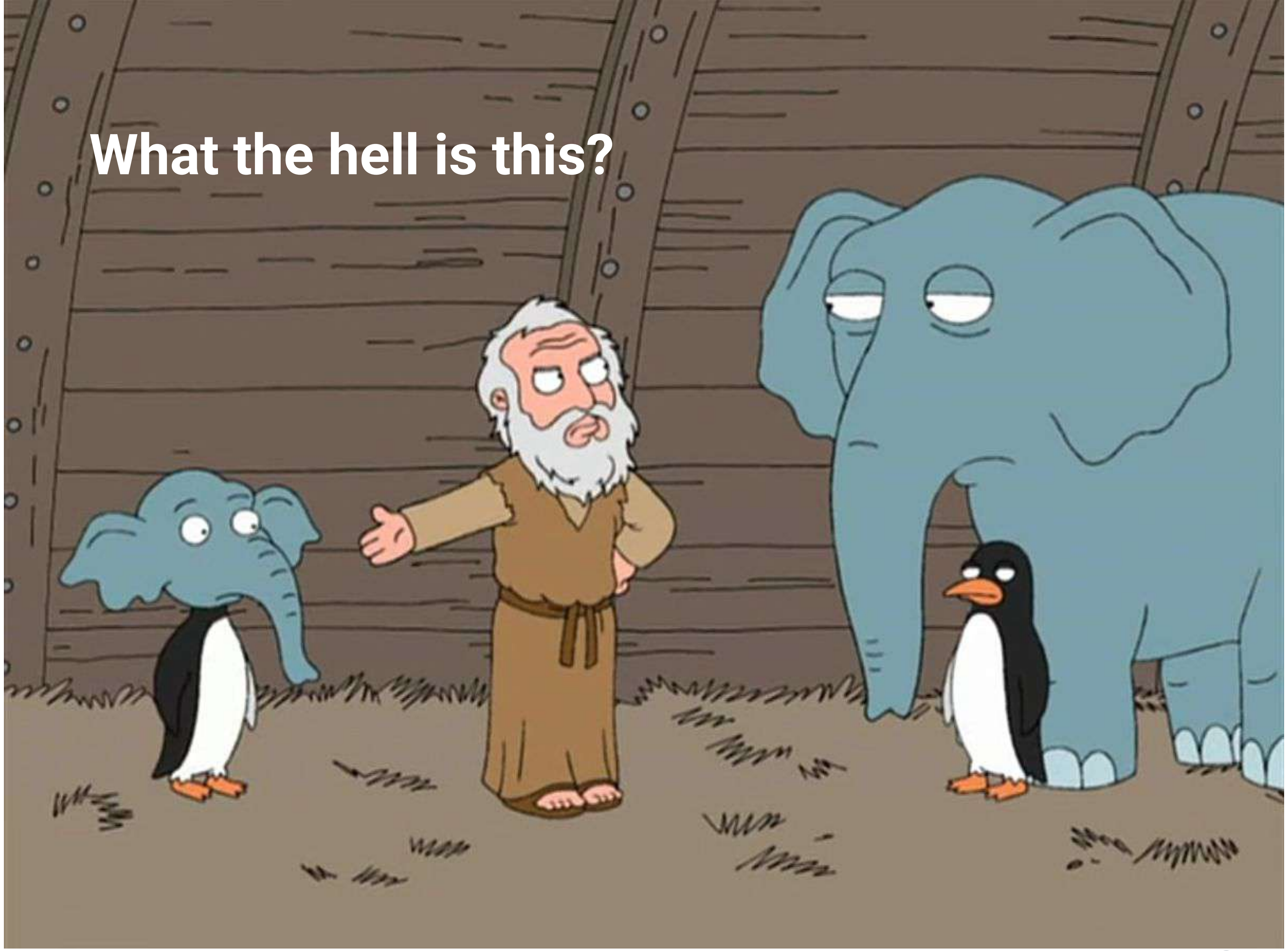
1.

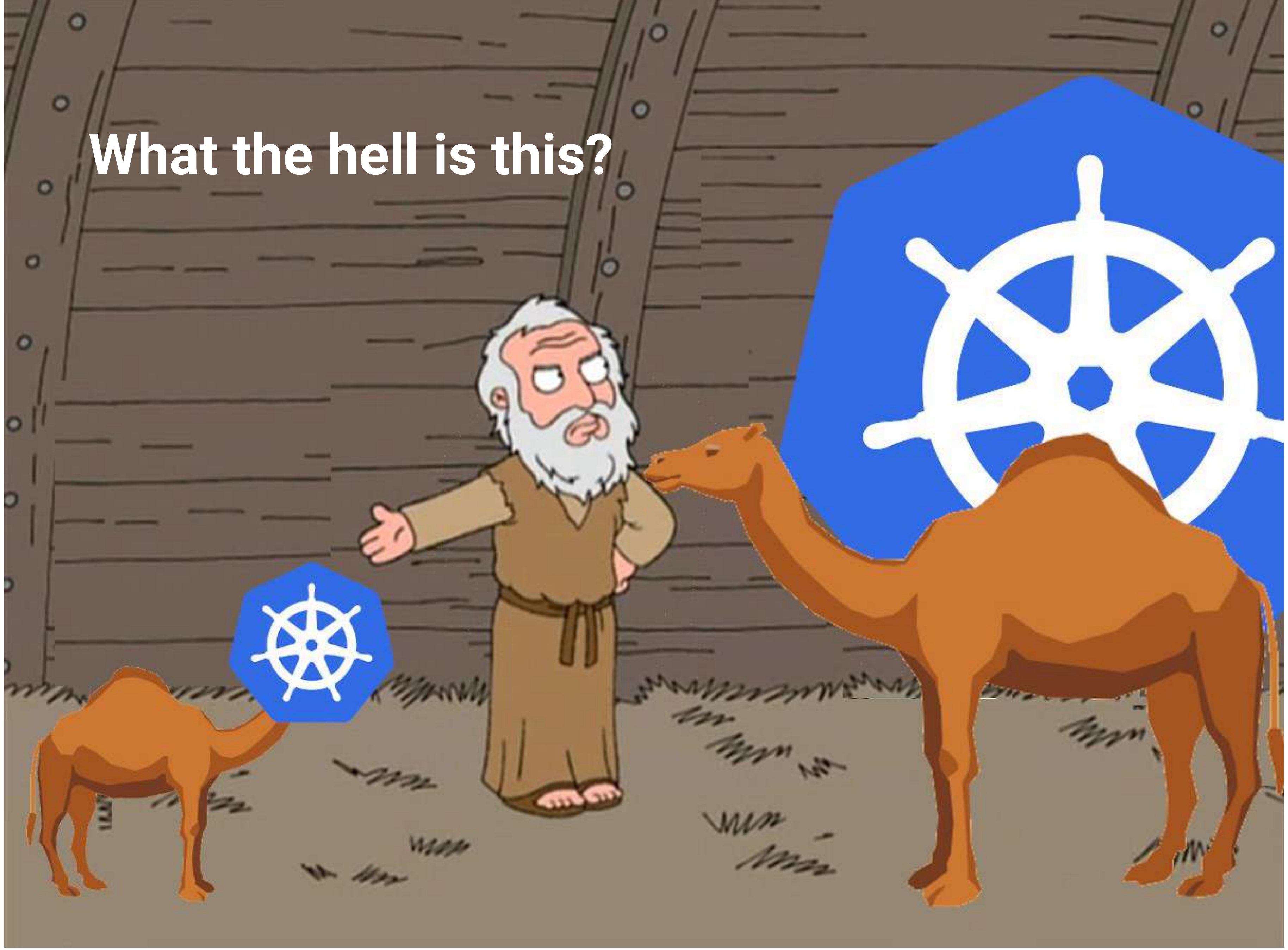
2.

3.

Runs on "vanilla" Kubernetes (1), Openshift (2)
and gives its best on a Knative-powered cluster (3)!

https://github.com/apache/camel-k


What the hell is this?


What the hell is this?

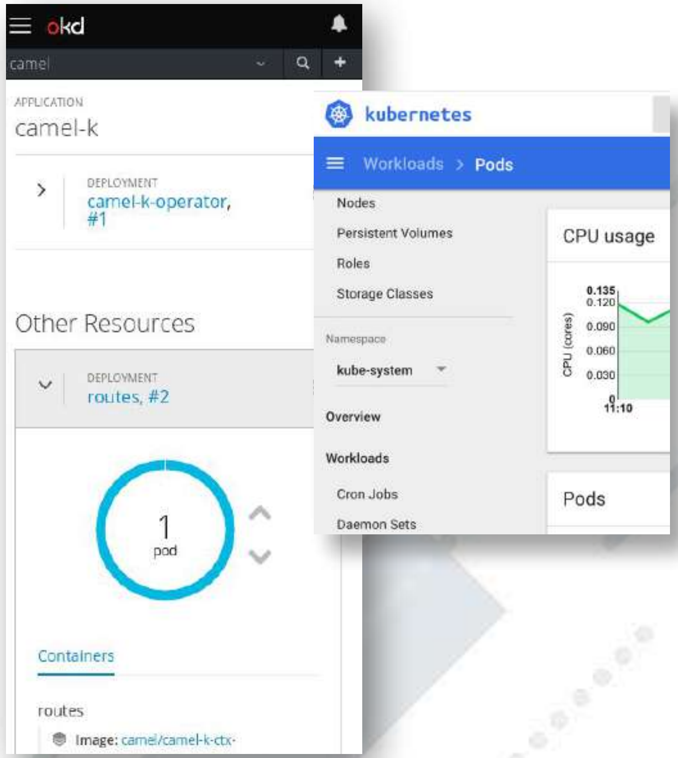
What?

Camel DSL, based on EIPs...

1. Create an integration file (Java, Groovy, Kotlin, JS, XML, ...)

```
from('paho:sensors?brokerUrl=tcp://mqtt-broker:1883')
  .log('${body}')
  .to('knative:channel/measures')
```

3. It runs on Kubernetes / OpenShift

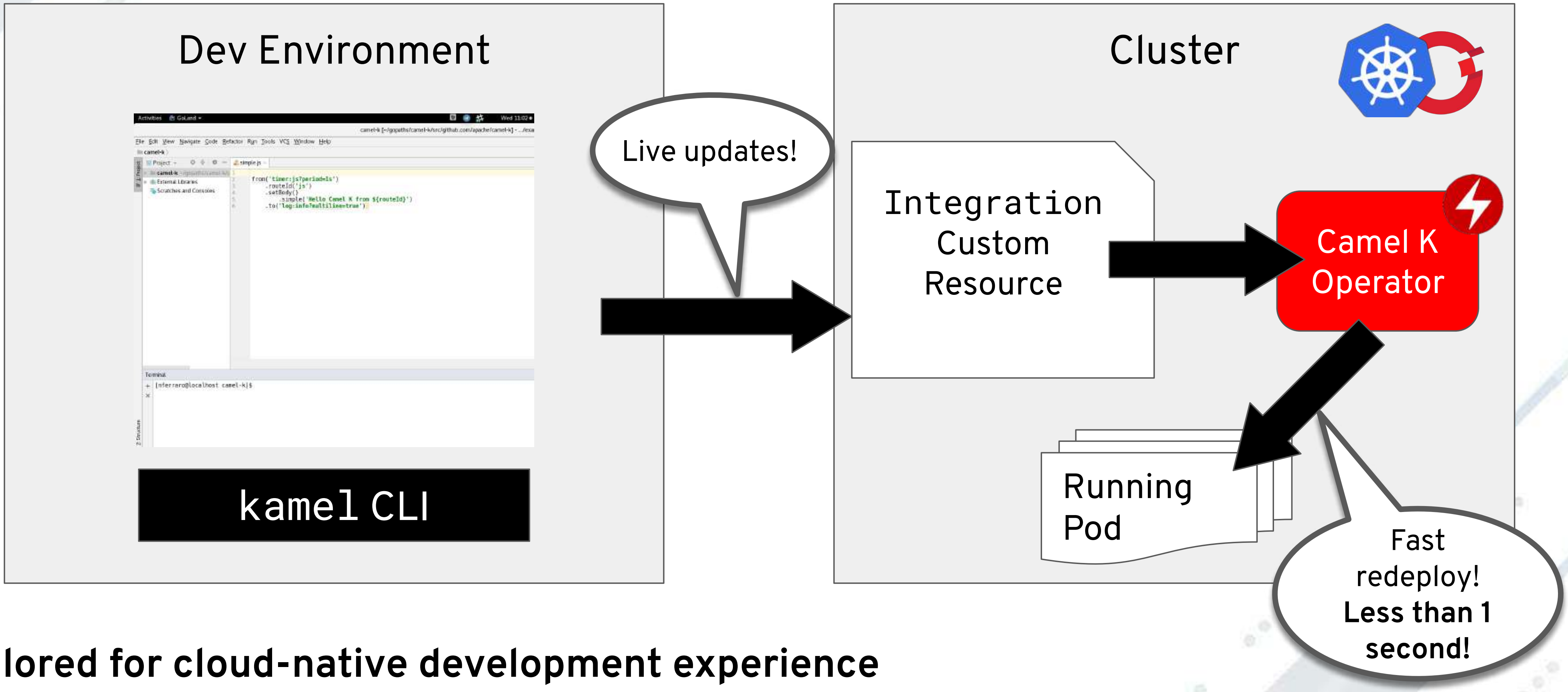


2. Run it

```
$ kamel run integration.groovy
```

Camel K CLI

How?


Dev Environment
Activities
GoLand
Wed 11:02
camel-k [~/gopaths/camel-k/src/github.com/apache/camel-k] - .../exa
File Edit View Navigate Code Refactor Run Tools VCS Window Help
camel-k
Project
simple.js
camel-k
External Libraries
Scratches and Consoles
Terminal
kamel CLI
Live updates!
Cluster
Integration Custom Resource
Camel K Operator
Running Pod
Fast redeploy! Less than 1 second!

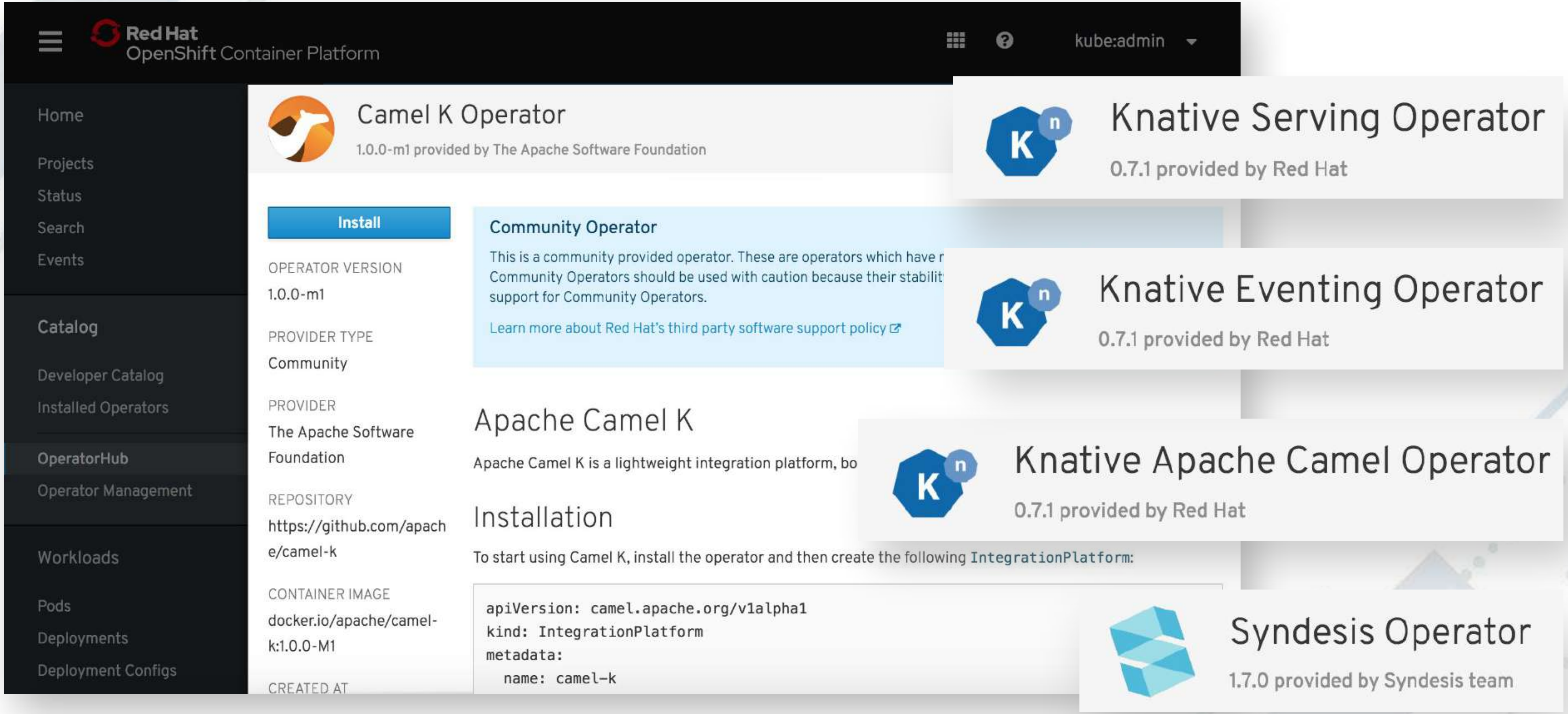
```
from('timer:js?period=1s')
    .routeId('js')
    .setBody()
        .simple('Hello Camel K from ${routeId}')
    .to('log:info?multiline=true')
```

```
[nferraro@localhost camel-k]$
```

Tailored for cloud-native development experience

https://github.com/operator-framework/operator-sdk

Operator Hub



Camel K Operator

1.0.0-m1 provided by The Apache Software Foundation

OPERATOR VERSION
1.0.0-m1

PROVIDER TYPE
Community

PROVIDER
The Apache Software Foundation

REPOSITORY
https://github.com/apache/camel-k

CONTAINER IMAGE
docker.io/apache/camel-k:1.0.0-M1

CREATED AT

Community Operator

This is a community provided operator. These are operators which have r
Community Operators should be used with caution because their stabilit
support for Community Operators.

Learn more about Red Hat's third party software support policy

Apache Camel K

Apache Camel K is a lightweight integration platform, bo

Installation

To start using Camel K, install the operator and then create the following `IntegrationPlatform`:

```
apiVersion: camel.apache.org/v1alpha1
kind: IntegrationPlatform
metadata:
  name: camel-k
```

Knative Serving Operator

0.7.1 provided by Red Hat

Knative Eventing Operator

0.7.1 provided by Red Hat

Knative Apache Camel Operator

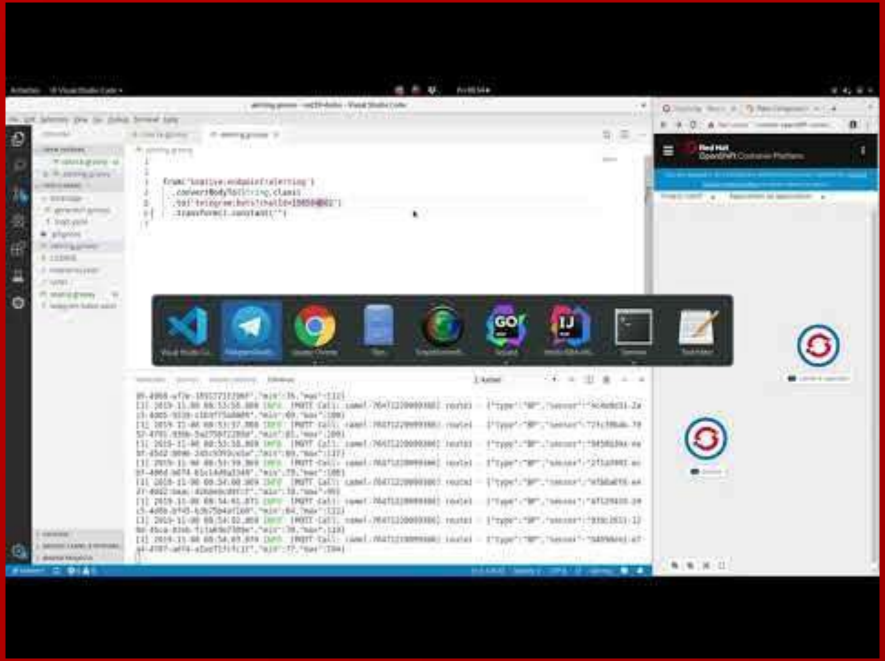
0.7.1 provided by Red Hat

Syndesis Operator

1.7.0 provided by Syndesis team

Red Hat https://operatorhub.io #RedHatOSD

DEMO: Camel K



```
from('knative:endpoint/alerting')
  .convertBodyTo(String.class)
  .to('telegram:bots?chatId=[illegible]')
  .transform().constant("")
```

Fuse Standalone

“Classic” integration

Fuse on OpenShift

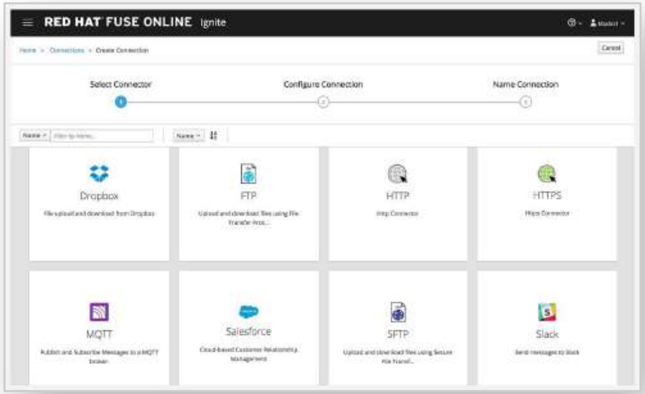
“Cloud native” integration

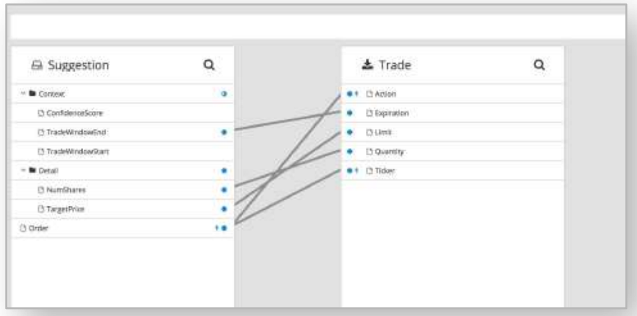
Fuse Online (iPaaS)

Low/no-code UX

SYNDESIS

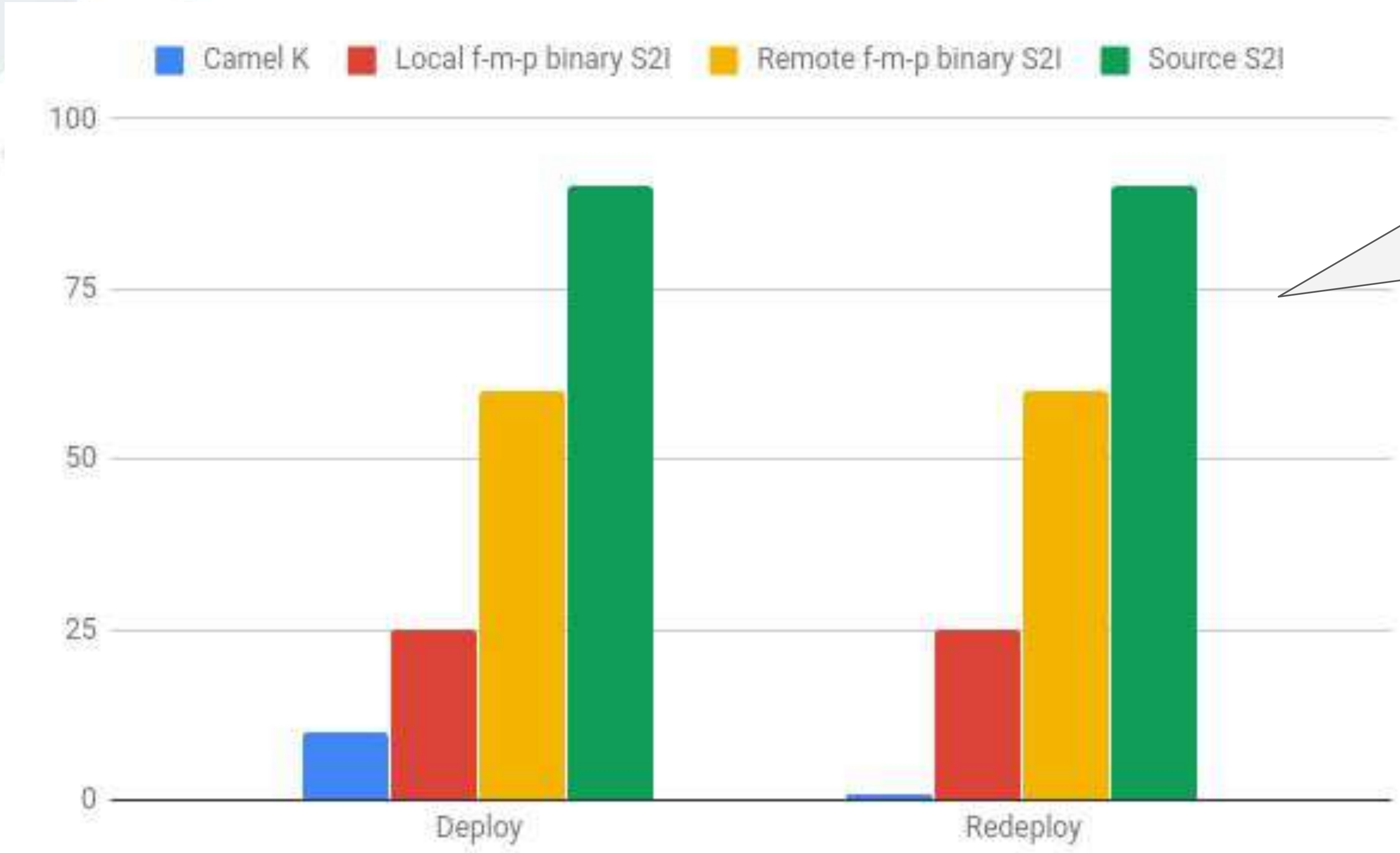
3scale by Red Hat





- Self-service
- API Design & Implementation
- Data Mapping
- Connectivity
- Intelligent Routing
- Extensibility

Time to run an integration in Syndesis



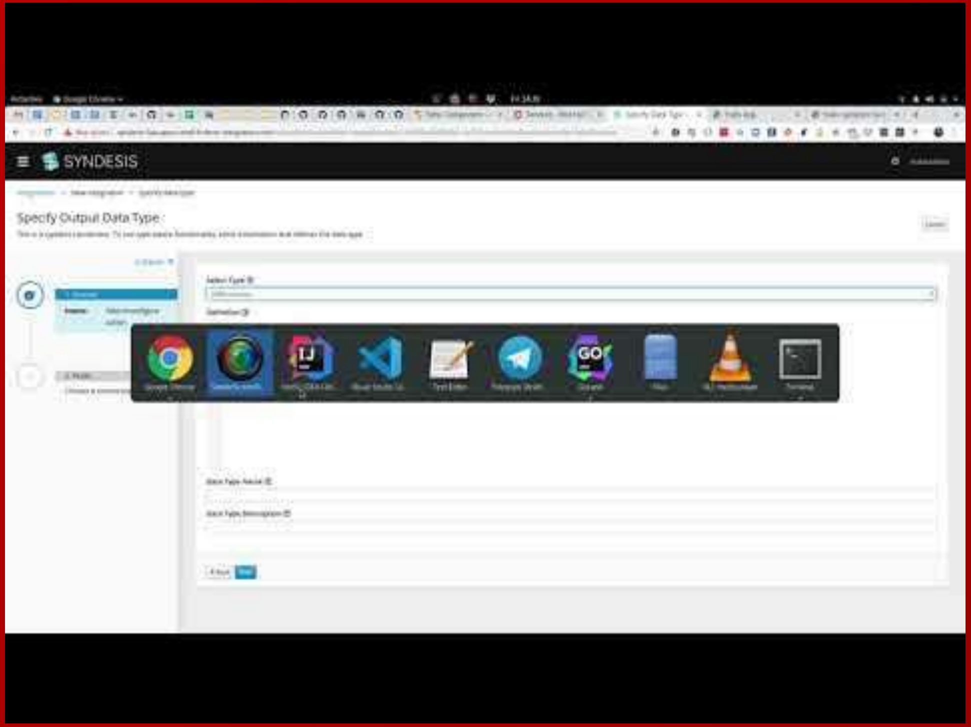
Lower is better :)

Red Hat

#RedHatOSD

#RedHatOSD

DEMO: Syndesis



Red Hat