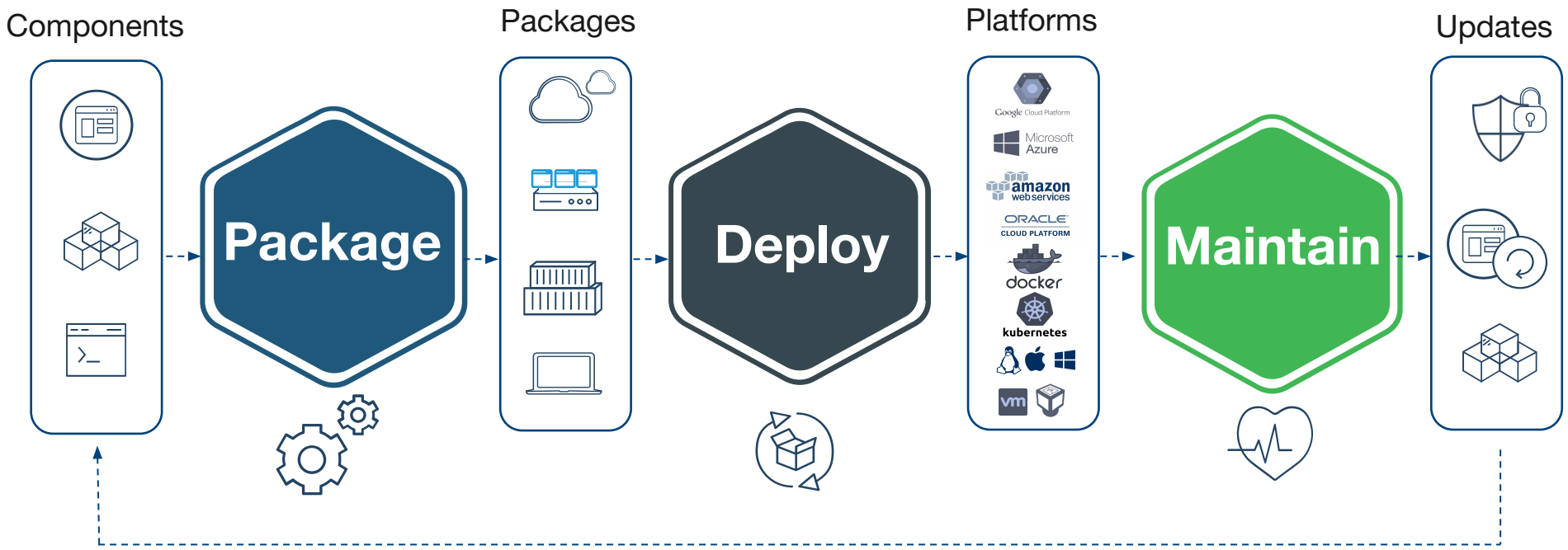
Building Serverless Application Pipelines

- Sebastien Goasguen

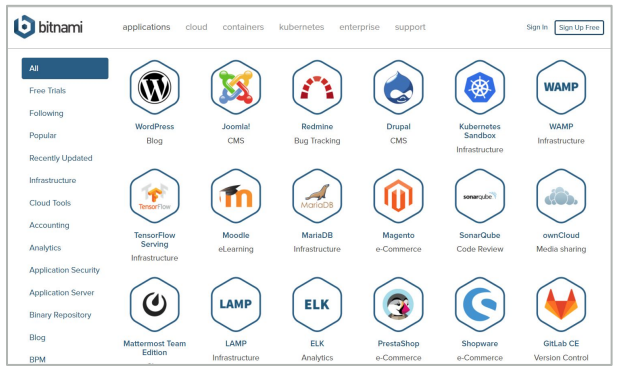
What do we do?



Our Products

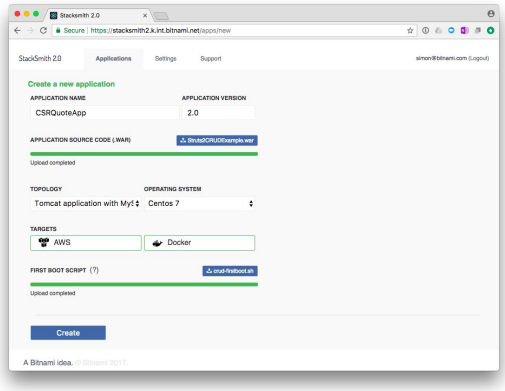
Application Catalog

- 150 applications & dev. runtimes
- Multiple formats: cloud, container, local VMs, native installers…
- Trusted, Maintained, Optimized



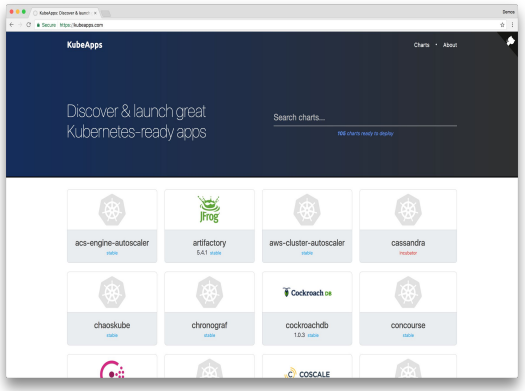
Stacksmith

- Enterprise cloud migration tool
- Productize Bitnami core technology
- Easily Re-platform applications for cloud



Kubernetes

- Defining packaging & deployment tools
- Key projects : Kubeapps, Kubeless, Helm
- Key partners: Microsoft, Deis, Heptio, SAP



Serverless


There is no 'serverless'
it's just someone elses fully managed
execution environment that I only pay a fraction
of a cent for whenever my function is run

Serverless computing refers to the concept of building and running applications that do not require server management. It describes a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform and then executed, scaled and billed on-demand in response to the exact demand needed at the moment.

Serverless Cloud Native Landscape

v0.9.5

App Definition & Development

Libraries

python-λ

Tools

Dashbird

IO|pipe

LambCI

Microcule

Node Lambda

STACKERY

THUNDRA

Frameworks

Chalice

Claudia.js

dawson

DEEP Framework

Gordon

kappa

Lambda SAM Local

serverless

Shep

SPARTA

Spring Cloud Function

APEX Up

Zappa

Platforms

AWS Lambda

Azure Functions

Clay

fn

Google Cloud Functions

HYPER.SH Func

IBM Cloud Functions

IronFunctions

NANO-LAMBDA

Now

OVERCLOCK

PubNub FUNCTIONS

spotinst

stdlib

syncano

twilio

WEBLAB

webtask

Hybrid

OpenWhisk

BINARIS

cloudboost

fx

GALACTIC FOG

nuclio

Kubernetes-native

PLATFORM9

Funktion

Kubeless

OPENFAAS

OpenLambda

Security

PURESEC

snyk

github.com/cncf/landscape

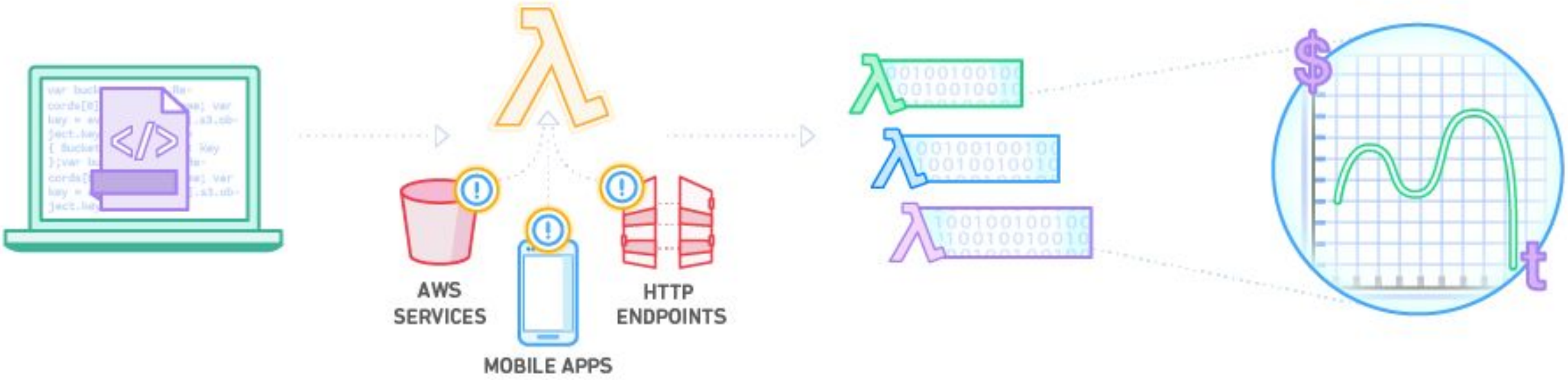
Serverless computing refers to a new model of cloud native computing, enabled by architectures that do not require server management to build and run applications. This landscape illustrates a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform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

CLOUD NATIVE COMPUTING FOUNDATION

Redpoint

Greyed logos are not open source

AWS Lambda



Upload your code to AWS Lambda

Set up your code to trigger from other AWS services, HTTP endpoints, or in-app activity

Lambda runs your code only when triggered, using only the compute resources needed

Pay just for the compute time you use

AWS CLI

```
$ aws lambda create-function \
 --region us-west-2 \
 --function-name CreateThumbnail \
 --zip-file fileb://file-path/CreateThumbnail.zip \
 --role role-arn \
 --handler CreateThumbnail.handler \
 --runtime runtime \
 --profile adminuser \
 --timeout 10 \
 --memory-size 1024
```

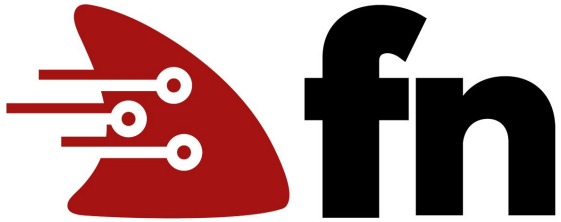
Concepts

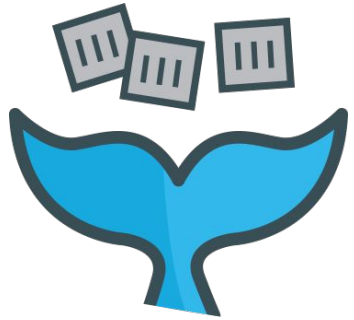
- Function endpoints
- Triggers
- Events

Serverless and FaaS Solutions

CLOUD FUNCTIONS

Create small, single-purpose functions
that respond to events in the cloud





Kubeless

SAP is evaluating and planning to contribute to the open source Kubeless project

BlackRock is using Kubeless to build search indices to enable data discovery for our portfolio managers and researchers. We continue to explore and evaluate the uses of this technology for other applications.

Kubernetes Native

Extend Kubernetes

Use the Kubernetes API server

Use Kubernetes API Objects

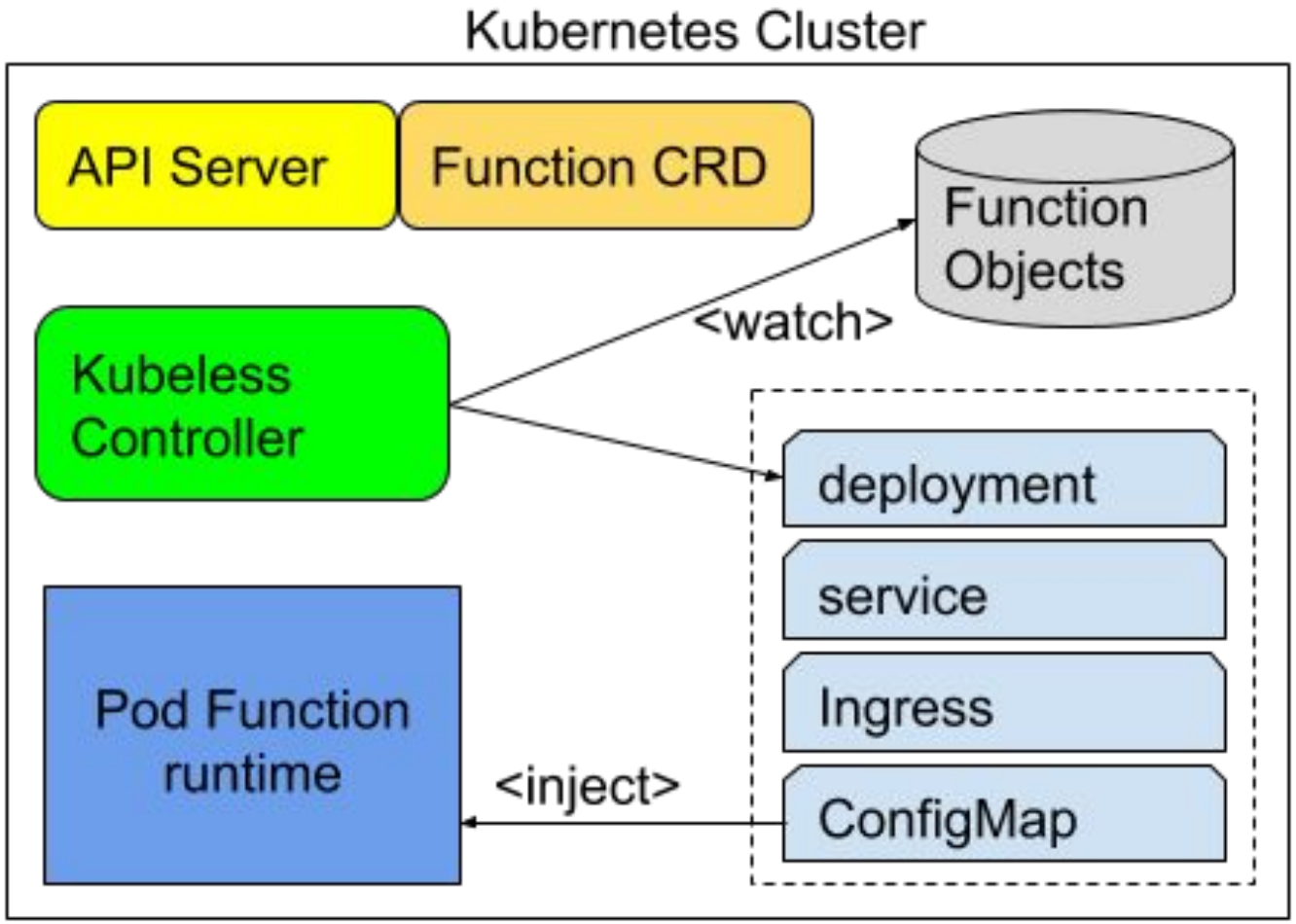
- Deployments/Services
- ConfigMaps
- Ingress

Horizontal Pod AutoScaler

Use CNCF monitoring - Prometheus

Use Istio/Envoy for traffic encryption, distributed tracing and more

Architecture



Use Custom Resource Definitions

```
$ kubectl get customresourcedefinition
NAME                                    AGE
functions.kubeless.io                   11h
$ kubectl get functions
NAME       AGE
hello   5h

$ kubectl get functions hello -o yaml
apiVersion: kubeless.io/v1beta1
kind: Function
metadata:
```

Controller Pattern

https://github.com/GoogleCloudPlatform/kube-metacontroller

```
$ kubectl get pods -n kubeless
NAME                                   READY  STATUS    RESTARTS   AGE
kubeless-controller-586c9498f9-pstmv   1/1    Running   0          11h
```

Monitoring

https://github.com/kubeless/kubeless/blob/master/docker/runtime/python-2.7/http-trigger/kubeless.py

```
...
import prometheus_client as prom
...
func_hist = prom.Histogram('function_duration_seconds',
                           'Duration of user function in seconds',
                           ['method'])
```

Dashboard


Kubeless
Zoom Out
Nov 23, 2017 10:59:50 to Nov 23, 2017 12:21:57
Function call rate
60
50
40
30
20
10
0
11:00
11:10
11:20
11:30
11:40
11:50
12:00
12:10
12:20
function=get-python
Function failure rate
1.0
0.5
0
-0.5
-1.0
11:00
11:10
11:20
11:30
11:40
11:50
12:00
12:10
12:20
function=get-python
Execution duration
0.0100
0.0075
0.0050
0.0025
0
11:00
11:05
11:10
11:15
11:20
11:25
11:30
11:35
11:40
11:45
11:50
11:55
12:00
12:05
12:10
12:15
12:20
{function="get-python"}

Autoscaling with custom metrics

https://github.com/kubeless/kubeless/tree/master/manifests/autoscaling

```
$ kubectl create -f custom-metrics.yaml

$ kubectl get po -n custom-metrics
NAME                                        READY     STATUS    RESTARTS   AGE
custom-metrics-apiserver-2956926076-wcgmw   1/1       Running   0          1h
```

Serverless Plugin

https://serverless.com/framework/docs/providers/kubeless/

CloudEvents

cloudevents

A specification for describing event data in a common way

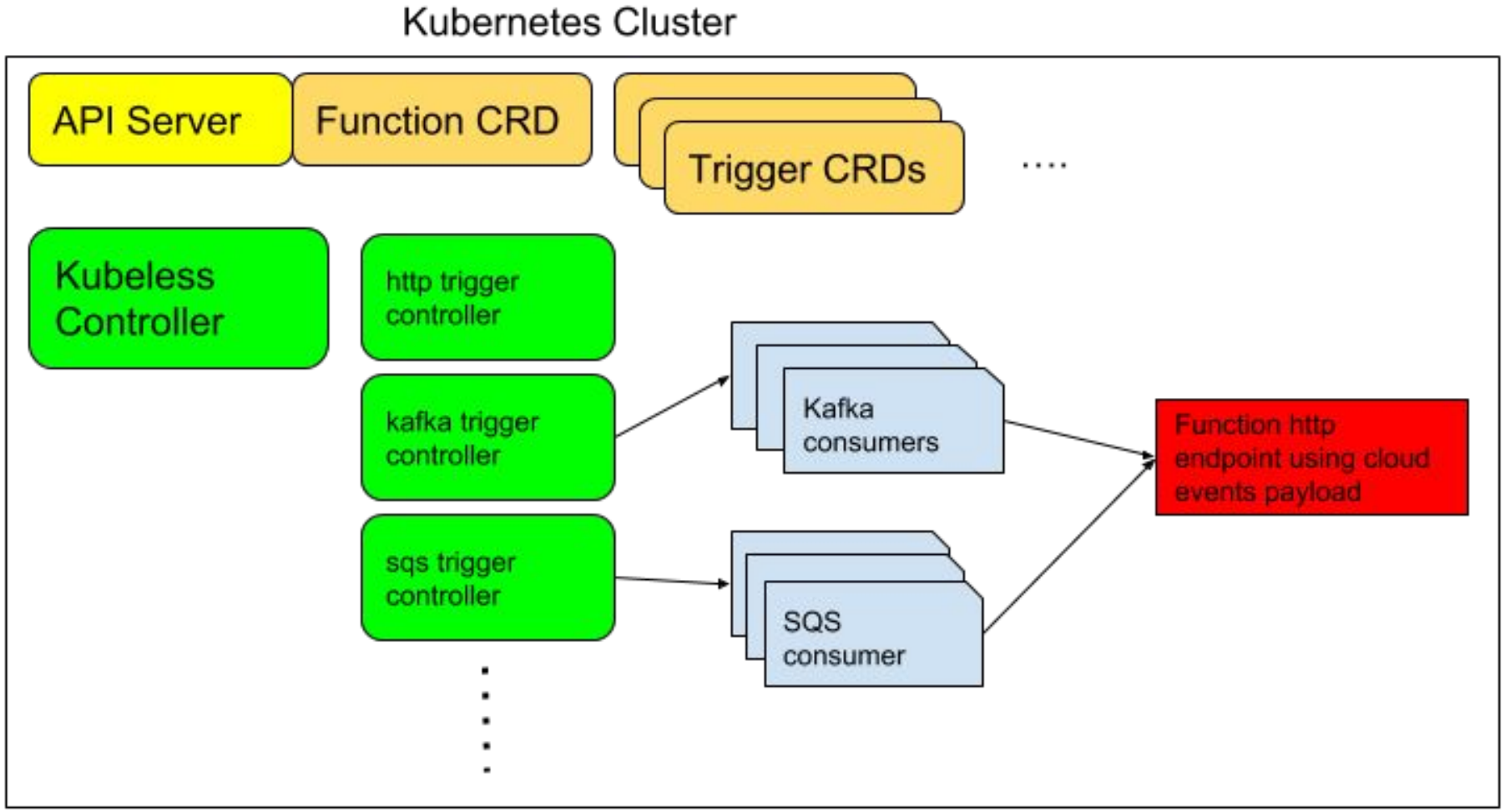
Events are everywhere. However, event publishers tend to describe events differently.

The lack of a common way of describing events means developers must constantly re-learn how to receive events. This also limits the potential for libraries, tooling and infrastructure to aide the delivery of event data across environments, like SDKs, event routers or tracing systems. The portability and productivity we can achieve from event data is hindered overall.

Kubeless function interface

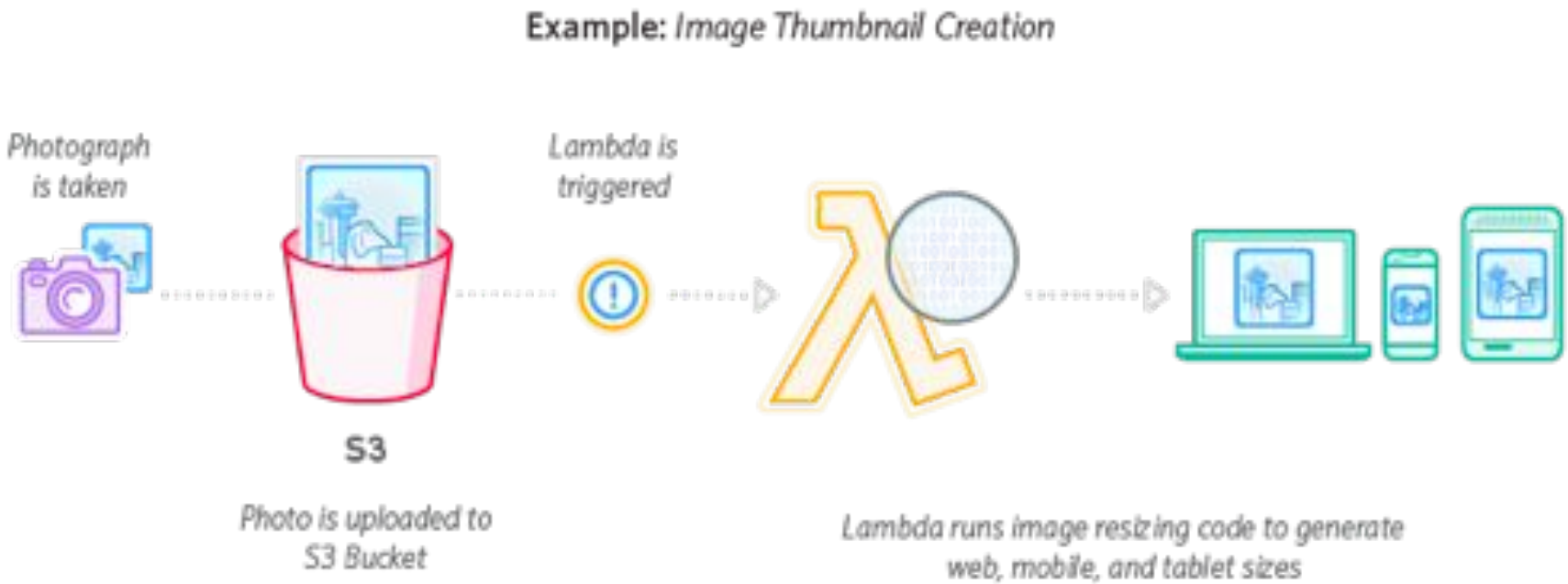
```
event:                                  # Event data
  data:
    foo: bar                              # Parameter when calling the function with a JSON object
  event-id: 123123
  event-type: application/json
  event-time: Tue Feb 20 2018 18:15:21 GMT+0000 (UTC)
  event-namespace: kafka.kubeless.io
  extensions:                           # Optional parameters, used to expose HTTP request properties
    request: ...
context:
    function-name: pubsub-nodejs
    timeout: 180
    runtime: nodejs6
    memory-limit: 128M
```

Scaling Event Sources

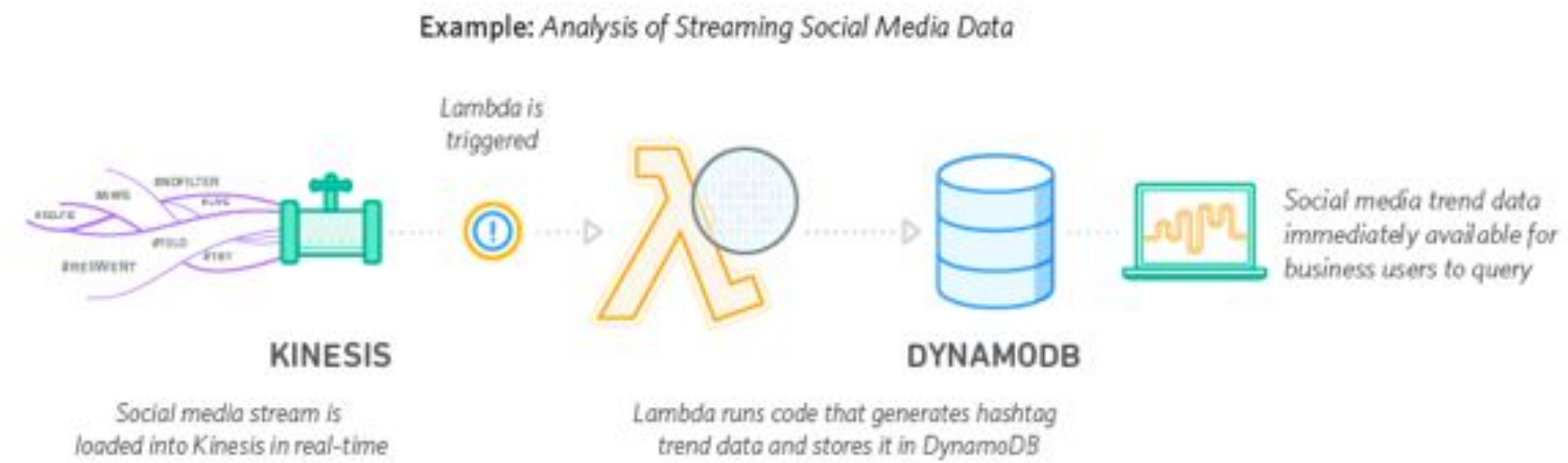


What is a serverless application ?

What type of Apps ?



Data Streams and Processing



How ?

Combine Charts + Service Broker + Functions

Local service using a Chart (deployed on prem or in managed Kubernetes)

Remote Cloud service, instantiated via the service catalog with bindings loaded as k8s secrets

Business logic deployed as functions and triggered via events.

Function Store at:
https://github.com/kubeless/functions

Demo on Katacoda

https://katacoda.com/kubeless/scenarios/getting-started

Thank You!
@sebgoa
https://github.com/kubeless/kubeless
http://kubeapps.com