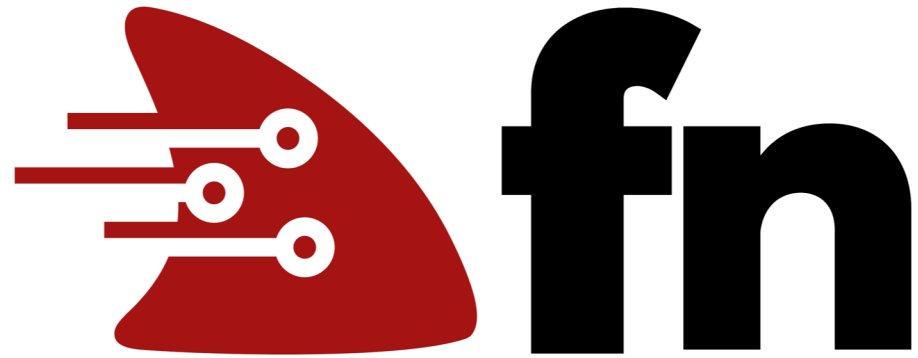# Fn Project

OPEN
SOURCE
DAYS

Zagreb – June 2018

David Delabassee
@delabassee
Oracle

Open Source Container Native FaaS Platform

Function as a Service

In mathematics, a **function** is a relation between a set of inputs and a set of permissible outputs with the property that each input is related to exactly one output. An example is the **function** that relates each real number x to its square x^2.

Function (mathematics) - Wikipedia
https://en.wikipedia.org/wiki/Function_(mathematics)

INPUT x
↓
FUNCTION f:
↓
OUTPUT f(x)
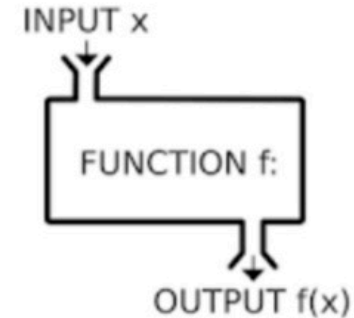
- **Function**   Small bits of code with a well defined job

                    Easy to understand and maintain

- **As a Service**   The system takes care of provisioning, scaling, patching, …

                    Each function can scale independently

fn

Why Serverless?

- **Easier**   Just think about your code, not the infrastructure
- **Powerful**   Scaling, Availability
- **Faster**   Deploy, iterate and innovate faster
- **Cheaper**   Only pay for what you use to the 100ms (never idle)

fn

The Ideal FaaS Platform?

- **Open Source**   No vendor lock-in

- **Approachable**   Easy for new users

                             Low level controls for advanced users

- **Container based**   Leverage Docker and its ecosystem

- **Scheduler independent** Deploy to K8S, Swarm, Mesos, ...

- **Language independent**   Go, Java, Python, …
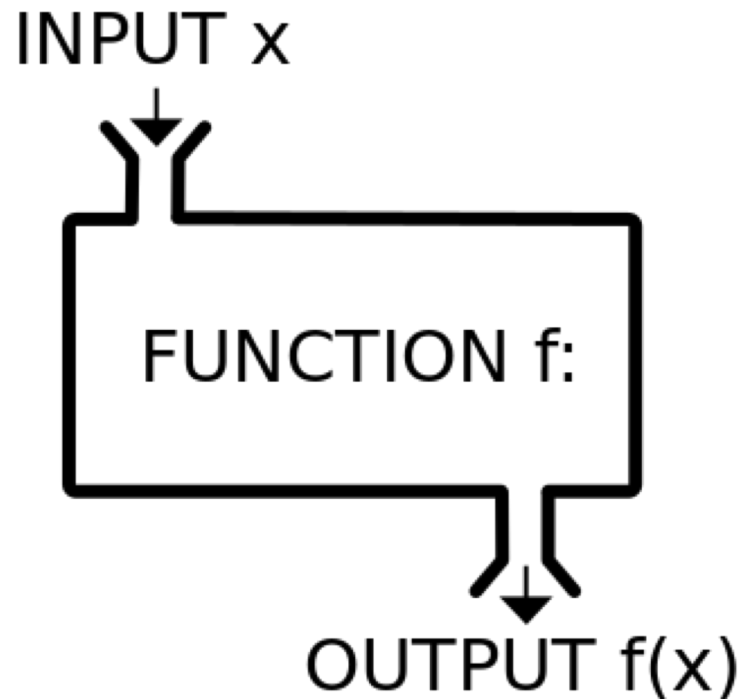
- **Platform independent**   Cloud, On-Perm, laptop

fn

Introducing Fn Project

Open Source Container Native FaaS Platform

https://github.com/fnproject
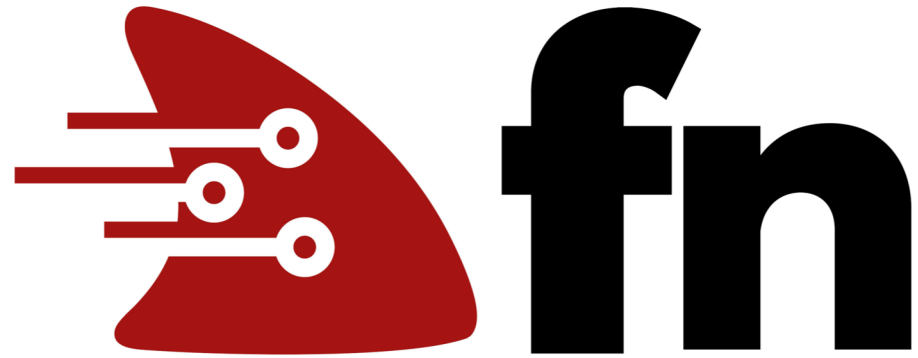
Introducing Fn Function

INPUT x

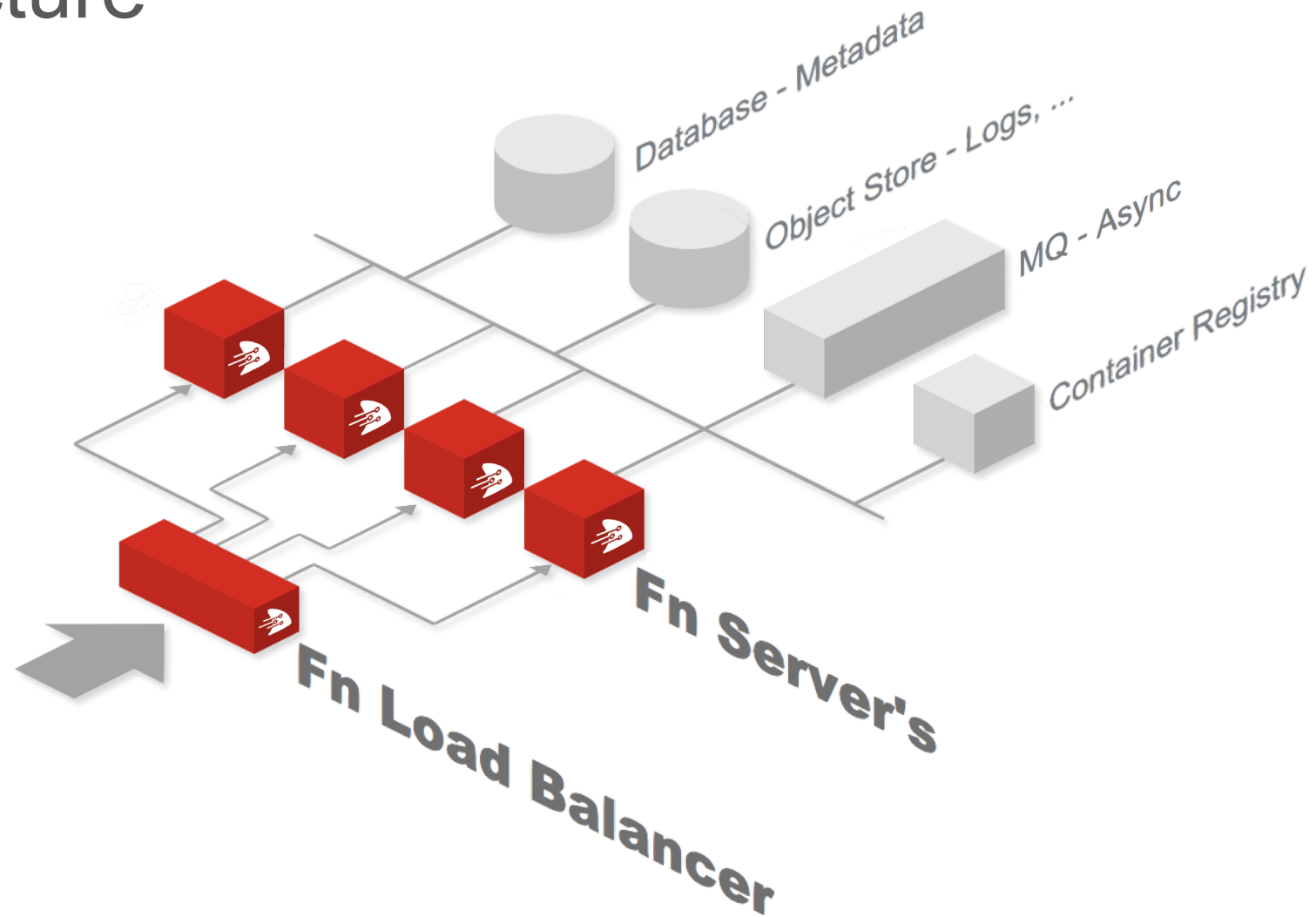FUNCTION f:

OUTPUT f(x)

- Code wrapped in a **Container Image**

- Input from **STDIN** and environment

- Output to **STDOUT**

- Logs to **STDERR**

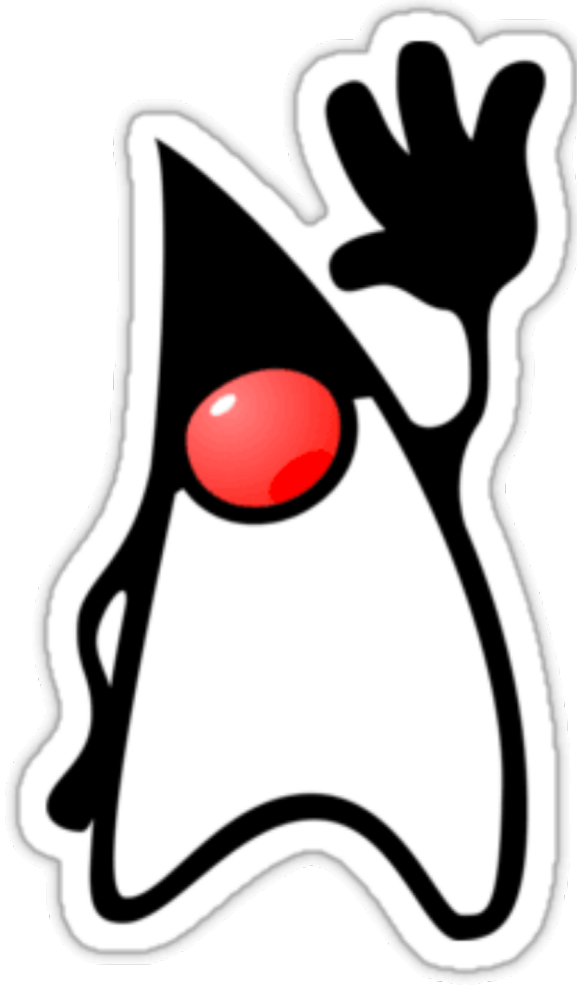The **Fn Server** handles everything else!

fn

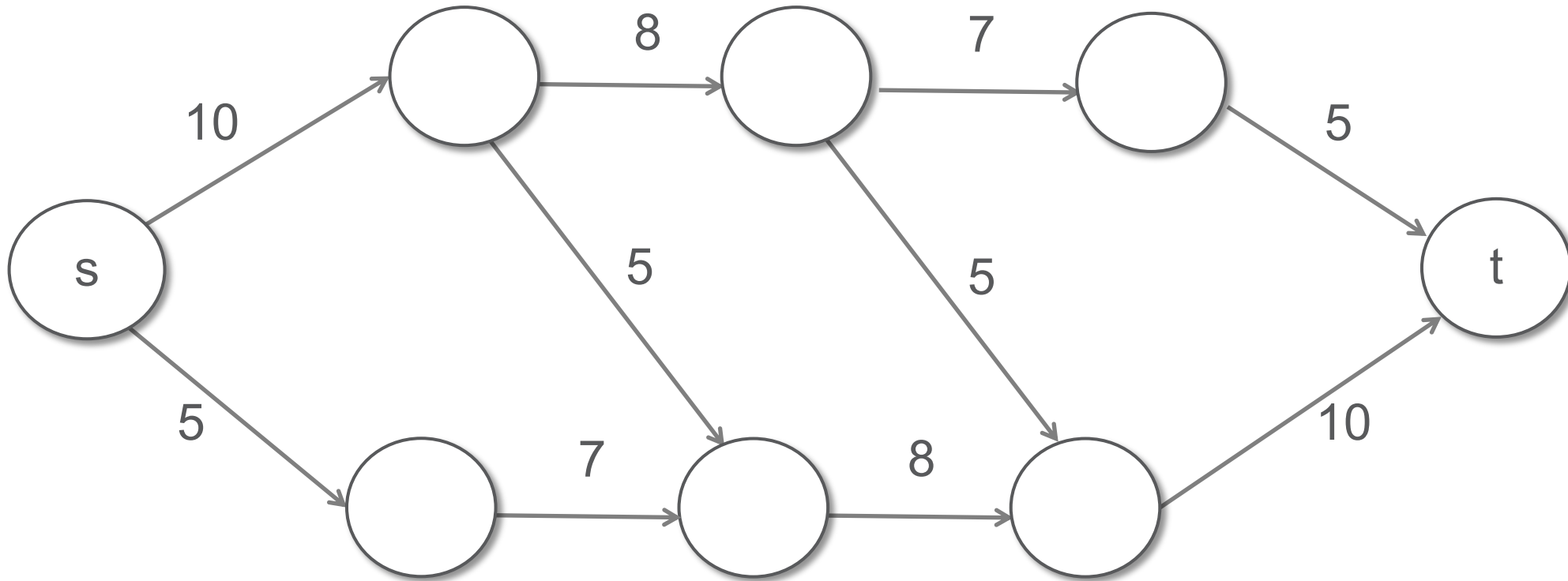# Fn Architecture

- Fn CLI
- Fn FDK's
- Fn Flow

Database - Metadata

Object Store - Logs, ...

MQ - Async

Container Registry

Fn Server's

Fn Load Balancer

Fn
Awesome Java support
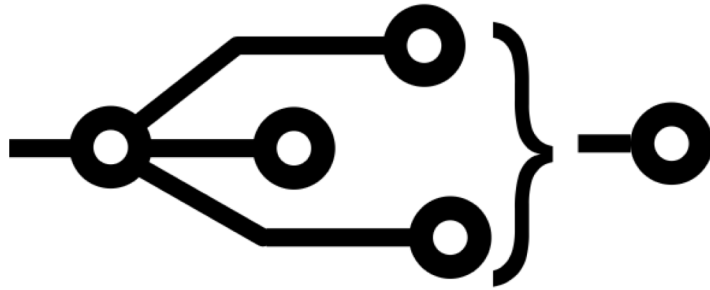
Fn
… not just Java

- Go

- Python

- JS

- Rust

- Ruby

- … bring your own!

fn

Serverless
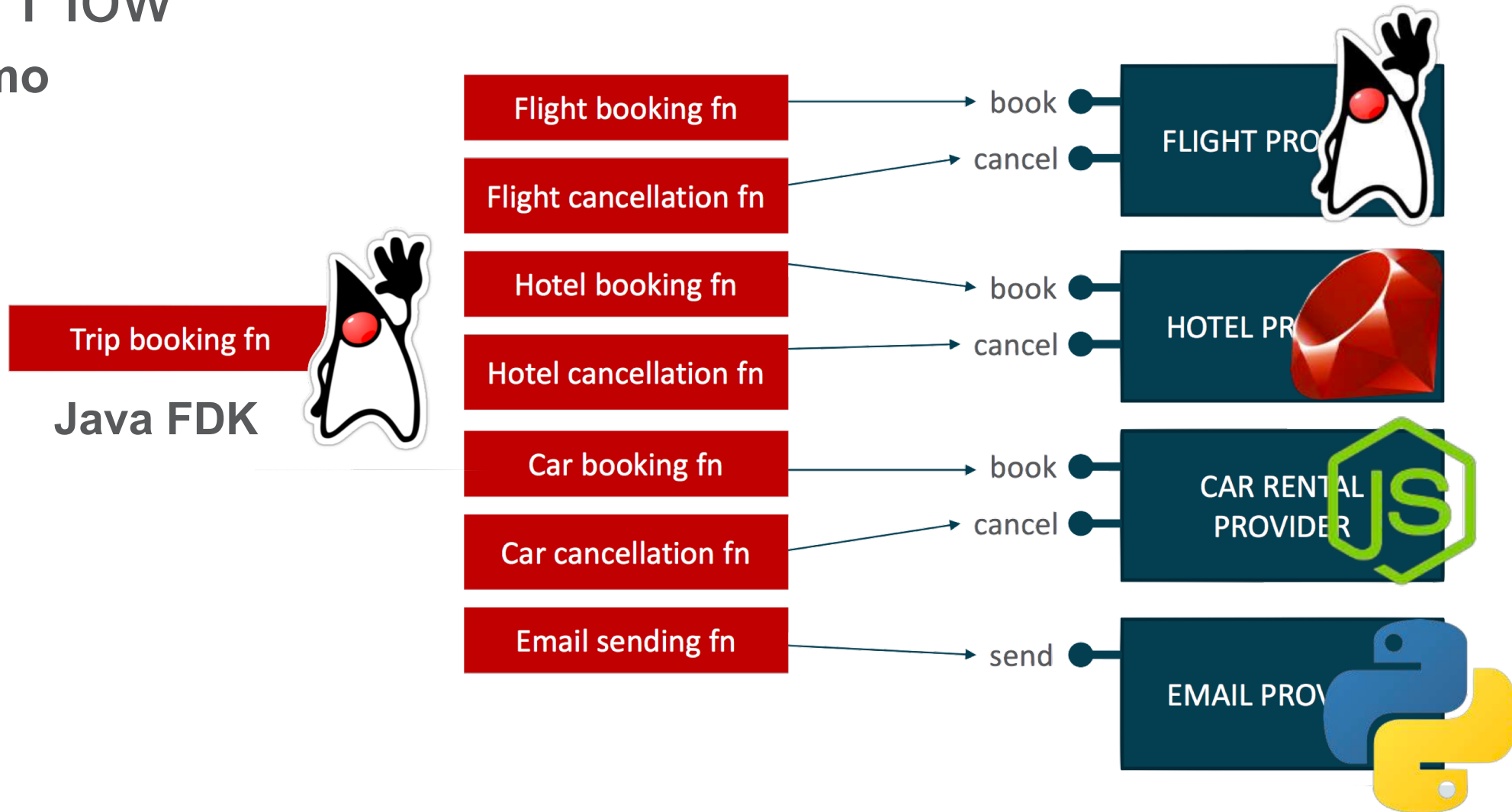## Permanent Storage Lives Elsewhere

fn

Fn Flow

- For long-running, reliable, scalable functions with primitives for fork-join, chaining, delays and error handling

- Java support based on Java 8 **CompletableFuture** API
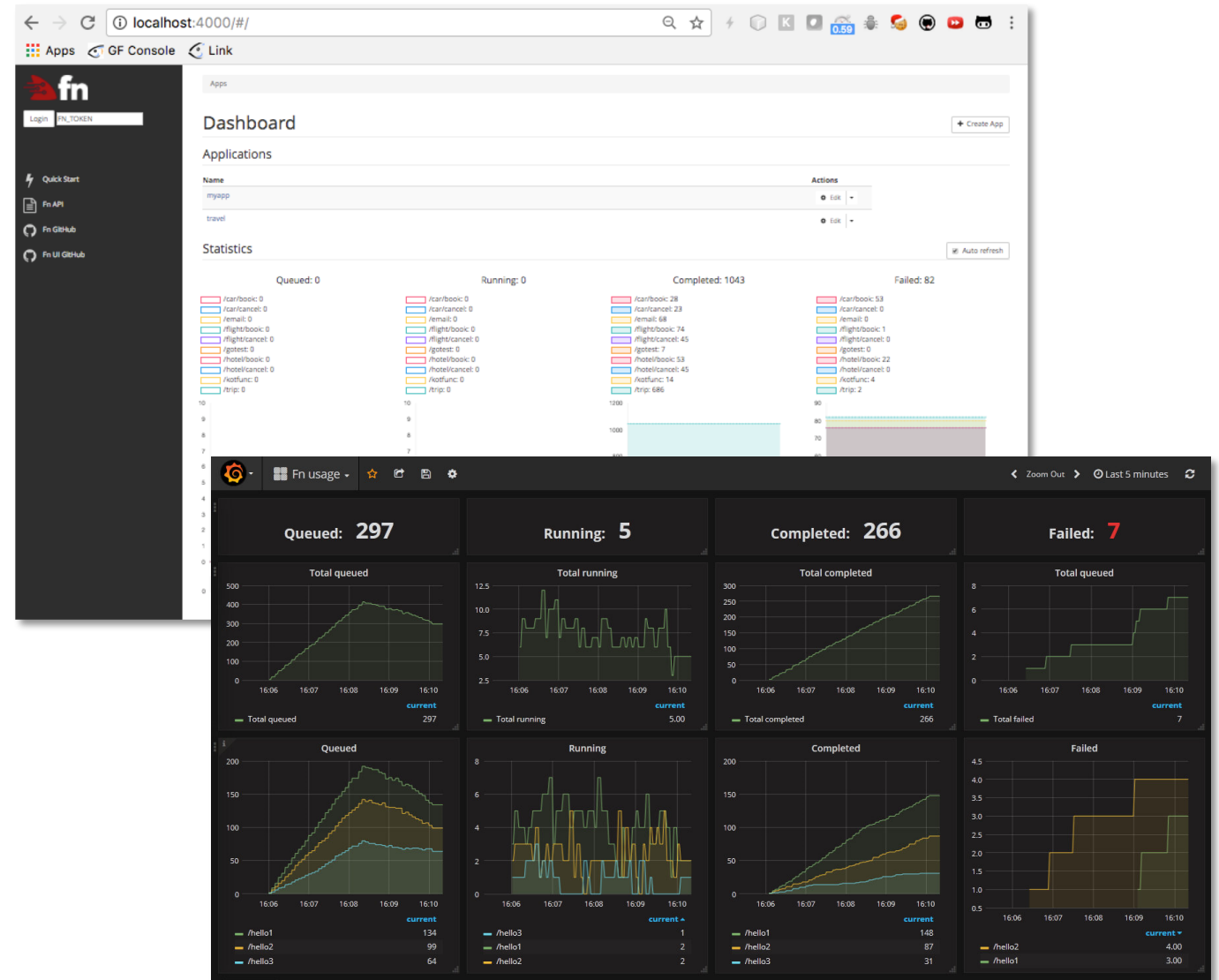
- Go, Node and Python support on the way!

Fn Flow
Demo

Fn
Monitoring & Management

- OpenTracing support
- Pluggable backend
  - Prometheus Metrics Sets
    - Function counts
    - Operation durations
    - Docker metrics
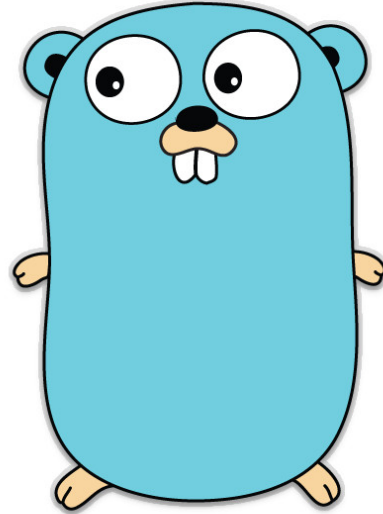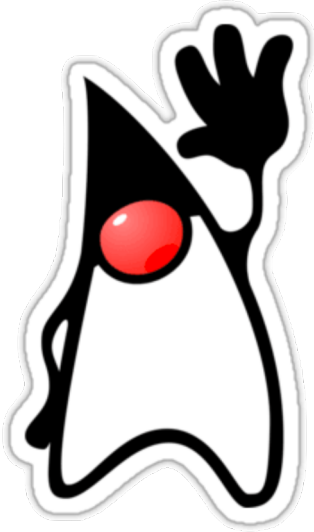- Logging via syslog
- Fn CLI
- Fn Dashboard

Fn

And more…

- Hot – Frozen – Cold Functions
- Async Functions
- Helm Chart for K8S
- JAX-RS support
- Spring Cloud Function support
- CNCF CloudEvents support
- Serverless Framework support, …

Fn
Recap

- Container Native, Cloud Agnostic, Polyglot, Open Source FaaS Platform
- `fn init | test | run | deploy | call`

fn

Fn - An Ideal Functions Platform?

- **Platform Independent** - laptop, server, cloud
- **Scheduler Independent** - deploy to Kubernetes, Swarm, etc.
- **Docker Based** - leverage Docker ecosystem
- **Open Source** - no vendor lock-in
- **Approachable** - easy for new users, low level controls for advanced users

fn

Fn
Resources

- https://github.com/fnproject/
- https://fnproject.slack.com/messages
- http://fnproject.io
- https://medium.com/fnproject/
- https://twitter.com/fnproj

fn

Hvala puno!

Fn
Resources

- https://github.com/fnproject/
- https://fnproject.slack.com/messages
- http://fnproject.io
- https://medium.com/fnproject/
- https://twitter.com/fnproj

Safe Harbor Statement

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.