



# Power of Serverless

Über das Open Source Fn Project

---

Thomas Robert – Oracle Hamburg

September 2018

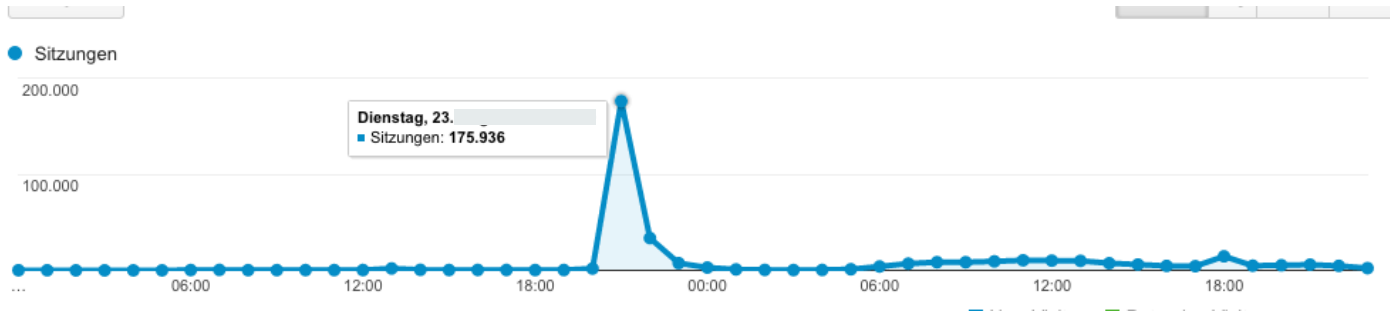
# Warum Serverless?

- Keine Gedanken mehr machen über die Infrastruktur
- Nicht mehr verantwortlich sein für:
  - An- oder Abschalten der Infrastruktur
  - Skalieren der Infrastruktur
  - Überwachen der Infrastruktur
- Einfache Nutzung
- Ergebnis ist:
  - Höhere Produktivität (schnelleres Feedback)
  - Beahlt wird nur für Usage, nicht für Leerlauf-Zeiten

# Functions-as-a-Service

- Serverless Compute
- Verknüpft mit Events  
(z.B. HTTP Requests, oder Message Queues)
- Deployment-Einheit ist eine Funktion  
(z.B. eine Java Methode)
- Jede Funktion skaliert unabhängig von anderen Funktionen

# Bereit für Lastspitzen



# Server gibt es trotzdem noch



# *Irgendwelche* Server



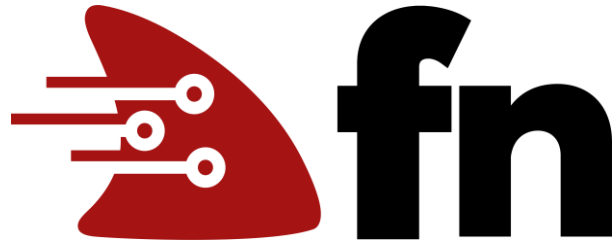
## **Run Fn Project on your Raspberry Pi – Pavel Varchenko – Medium**

I have a cluster of Raspberry Pi for my pet-projects and believe that serverless is very good for home automation. Couple months ago I...

[medium.com](https://medium.com)

# Eine optimale Functions Plattform

- **Open Source** – Kein Hersteller Lock-In
- **Plattformunabhängig** – Laptop, Server, Cloud
- **Benutzerfreundlich** – Einfach für neue Benutzer, detaillierte Kontrolle für anspruchsvolle Anforderungen
- **Docker basierend** – läuft unter Kubernetes, Swarm, Mesos usw



[www.fnproject.io](http://www.fnproject.io)



# Das Projekt Fn

- Open-Source Functions Plattform
- Läuft in beliebigen Cloud Umgebungen u. im eigenen RZ
- Einfach und erweiterbar
- Container basierend
- Aktive Community mit >3200 Commits und >60 Contributoren
- Unabhängig von Oracle. Unterstützt durch die Cloud Native Computing Foundation (CNCF)
- Starke Enterprise Fokussierung (Security, Skalierbarkeit, Administrierbarkeit etc.)

# Eine Fn Funktion

- Kleines Stück Code, mit allen Abhängigkeiten in ein Container Image gepackt
- Fn Server übergibt Input und Environment an die Funktion
- Fn Server gibt Ergebnis an den Aufrufer zurück
- STDERR automatisch an Logs gesandt

Der Fn Server übernimmt alle Aufgaben wie API Gateway, Weiterleitungen, Speichern der Logs etc.



# Functions == Container

Functions werden als Container bereitgestellt – somit kann jeder Container als Funktion deployt werden

=> Wiederverwendung von Open Source Bibliotheken als Functions!



# Fn Command Line Interface (CLI)

- `fn use context prod-env`
- `fn init --runtime java --trigger http myfunc`
- `fn deploy --app myapp`
- `fn invoke myapp myfunc`

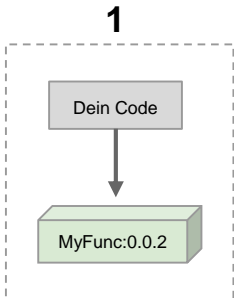
trigger → `http://localhost:8080/t/myapp/mytrigger`



**Demo**

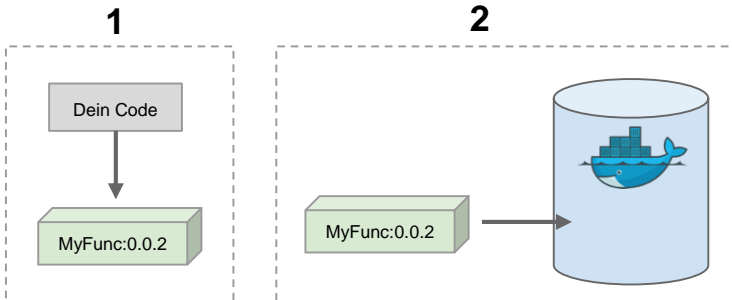
# Was passiert beim fn deploy

- Bauen des Containers (multi-stage) + Version erhöhen



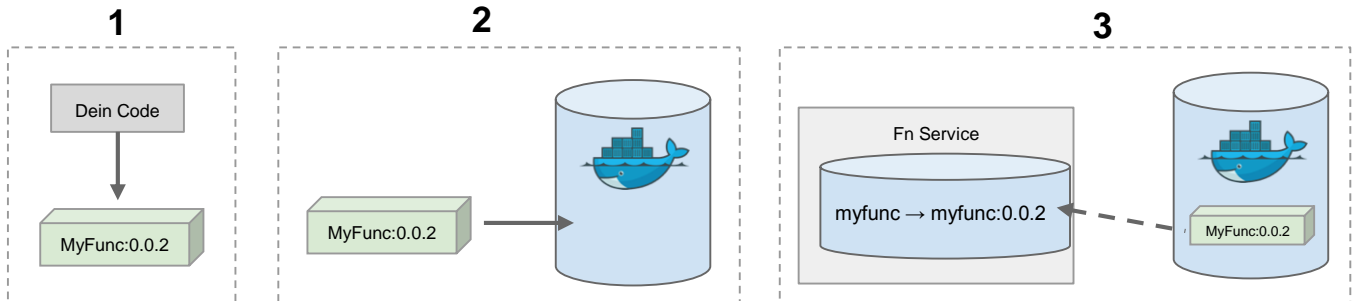
# Was passiert beim fn deploy

- Bauen des Containers (multi-stage) + Version erhöhen
- Push des Containers in die Registry



# Was passiert beim fn deploy

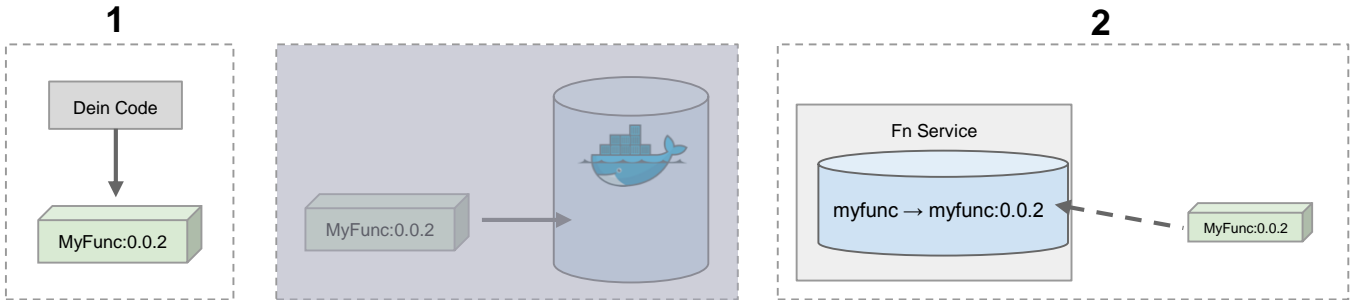
- Bauen des Containers (multi-stage) + Version erhöhen
- Push des Containers in die Registry
- Anlegen der Funktion & Trigger (Image wird erst beim Aufruf geladen)





# fn deploy --local

- Bauen des Containers (multi-stage) + Version erhöhen
- ~~Push des Containers in die Registry~~
- Anlegen der Funktion & Trigger (Image wird erst beim Aufruf geladen)



# Function Development Kit (FDK)

- Unterstützt Parsen des Input und Schreiben des Output
- Benötigt nur eine “handler” Funktion. Das FDK Interface übergibt STDIN und sorgt für korrektes Output Format
- Einfaches Erstellen einer Funktion



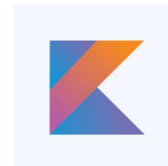
# Auch für beliebige andere Sprachen



elixir



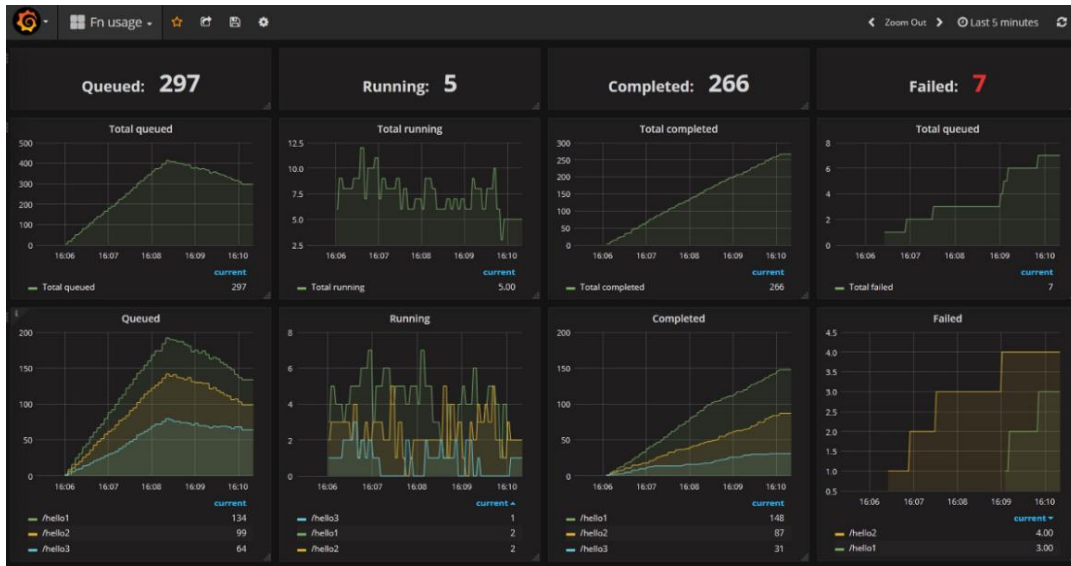
Clojure



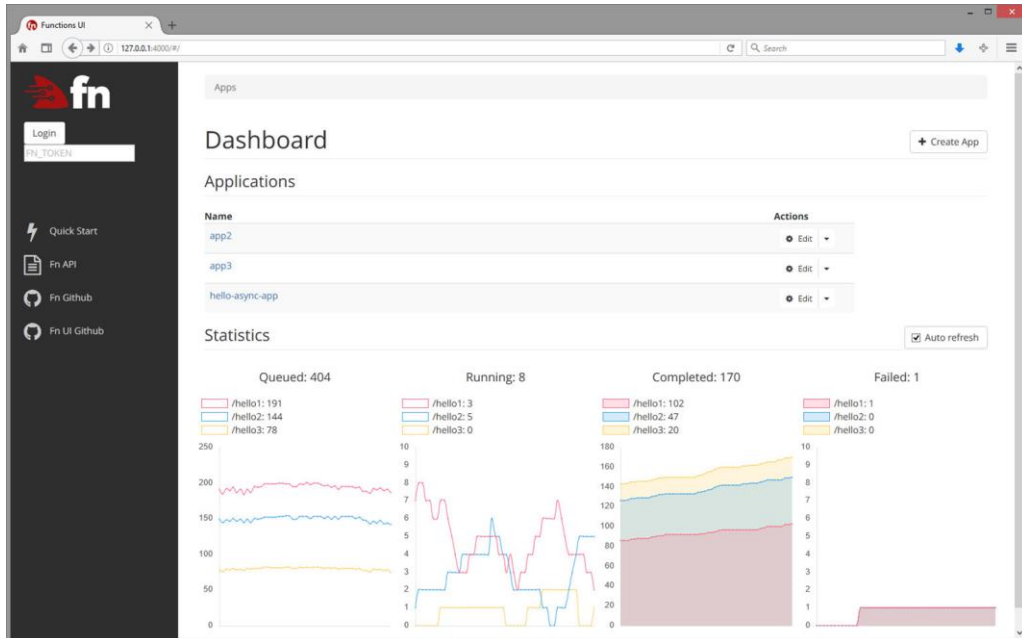
# Optimaler Java Support (natürlich)



# Open Tracing und Prometheus

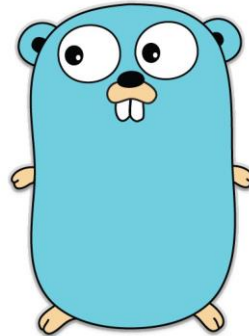
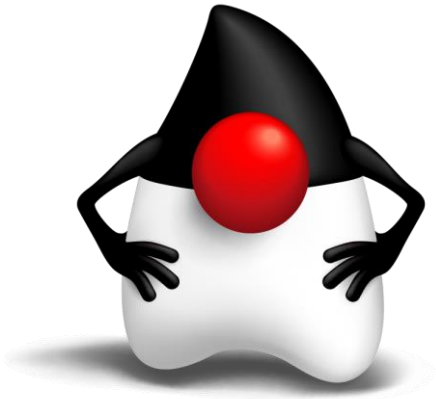


# Fn UI



# **Fn Kafka Demo**

Es ist (einfach) nur Java oder Go  
oder Node.js or ...





# Fn – Eine ideale Functions Plattform?

- ✓ **Open Source** – Kein Hersteller Lock-In
- ✓ **Plattformunabhängig** – Laptop, Server, Cloud
- ✓ **Benutzerfreundlich** – Einfach für neue Benutzer, detaillierte Kontrolle für anspruchsvolle Anforderungen
- ✓ **Docker basierend** – läuft unter Kubernetes, Swarm, Mesos usw

<http://fnproject.io>



# Vielen Dank!

---

## Mach mit

1. Erste Schritte: [github.com/fnproject/fn](https://github.com/fnproject/fn)
2. Meinungen Austauschen: [slack.fnproject.io](https://slack.fnproject.io)
3. Mehr Infos: [fnproject.io](https://fnproject.io)
4. Kontakte: [katrin.tietz@oracle.com](mailto:katrin.tietz@oracle.com)  
[thomas.leopold@oracle.com](mailto:thomas.leopold@oracle.com)