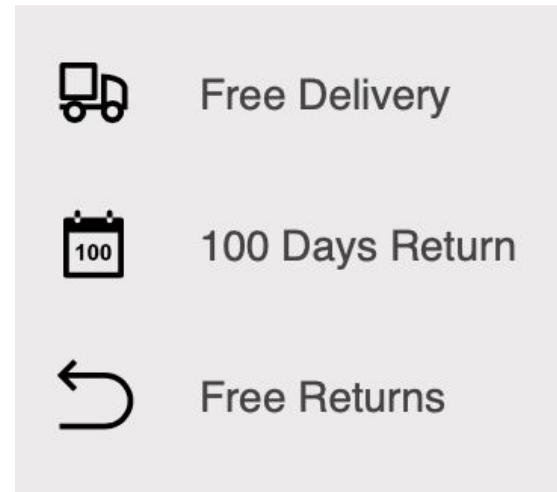


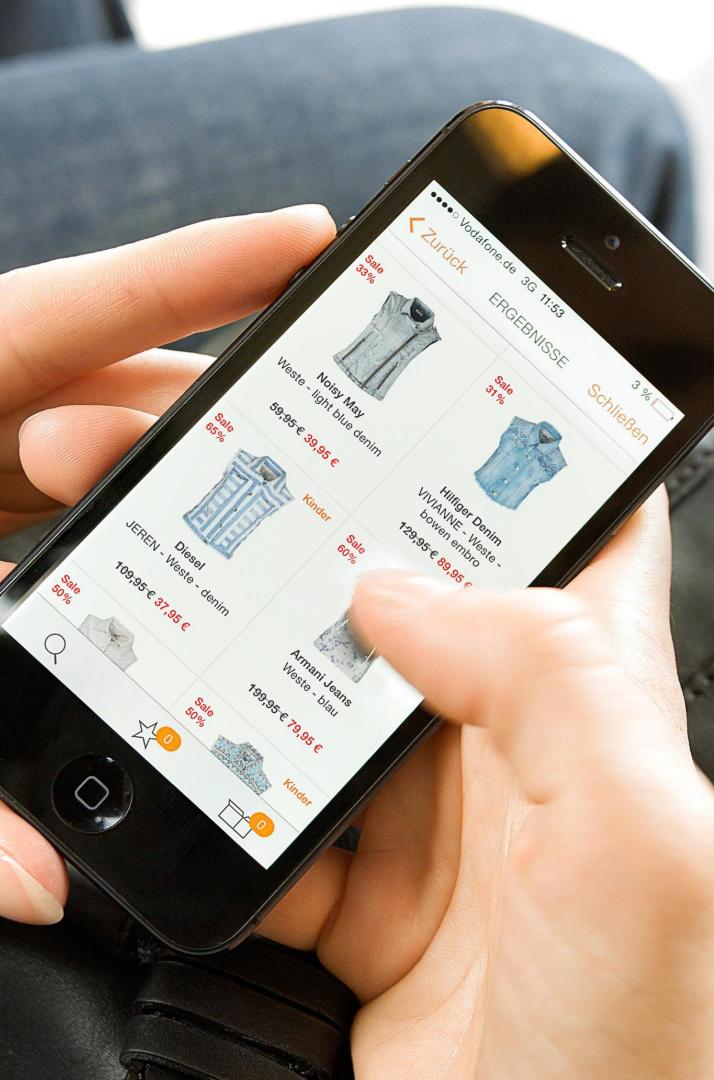
# Building and running applications at scale in Zalando

Online fashion store Checkout case

By Pamela Canchanya

# About Zalando





# About Zalando

~ **5.4** billion EUR

**revenue 2018**

> **15.500**

employees in  
Europe

> **70%**

of visits via  
mobile devices

> **250**  
**million**

visits  
per  
month

> **26**  
**million**  
active customers

> **300.000**

product choices

~ **2.000**

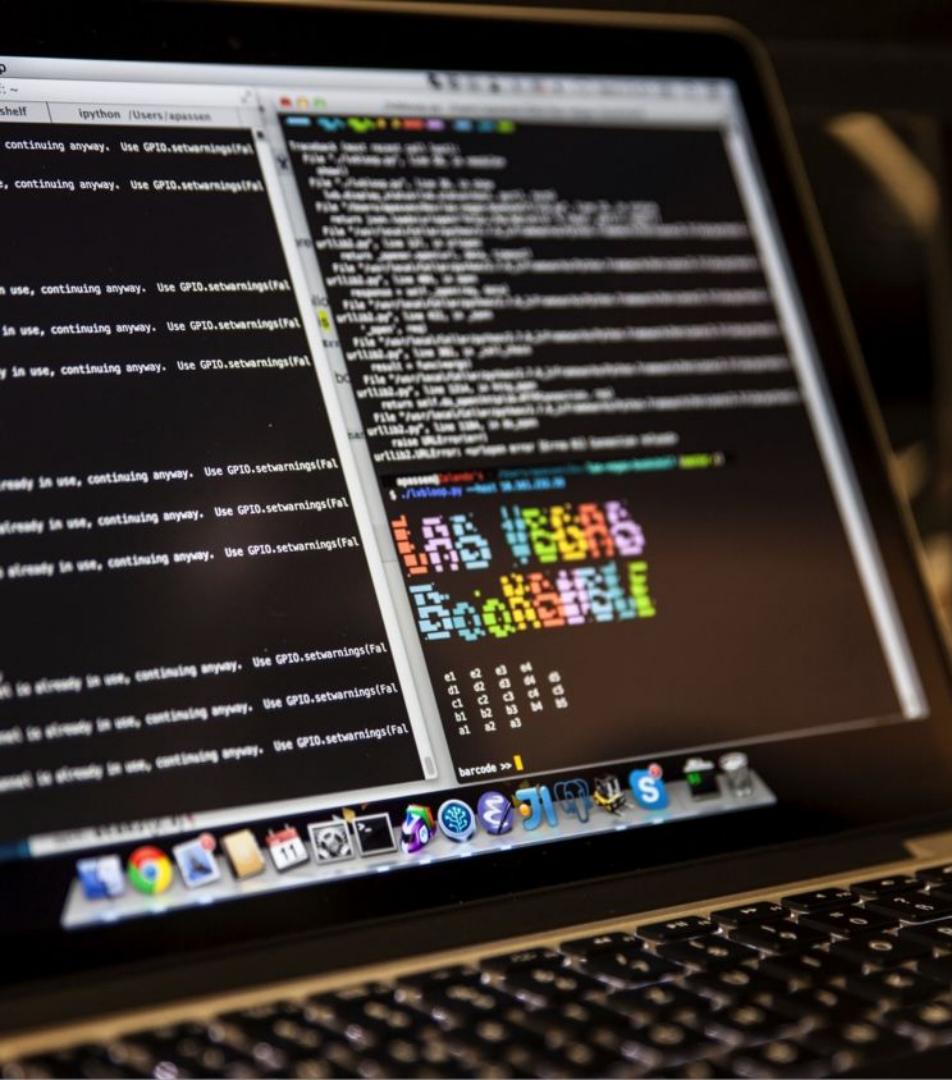
brands

**17**

countries

# Black Friday at a glance

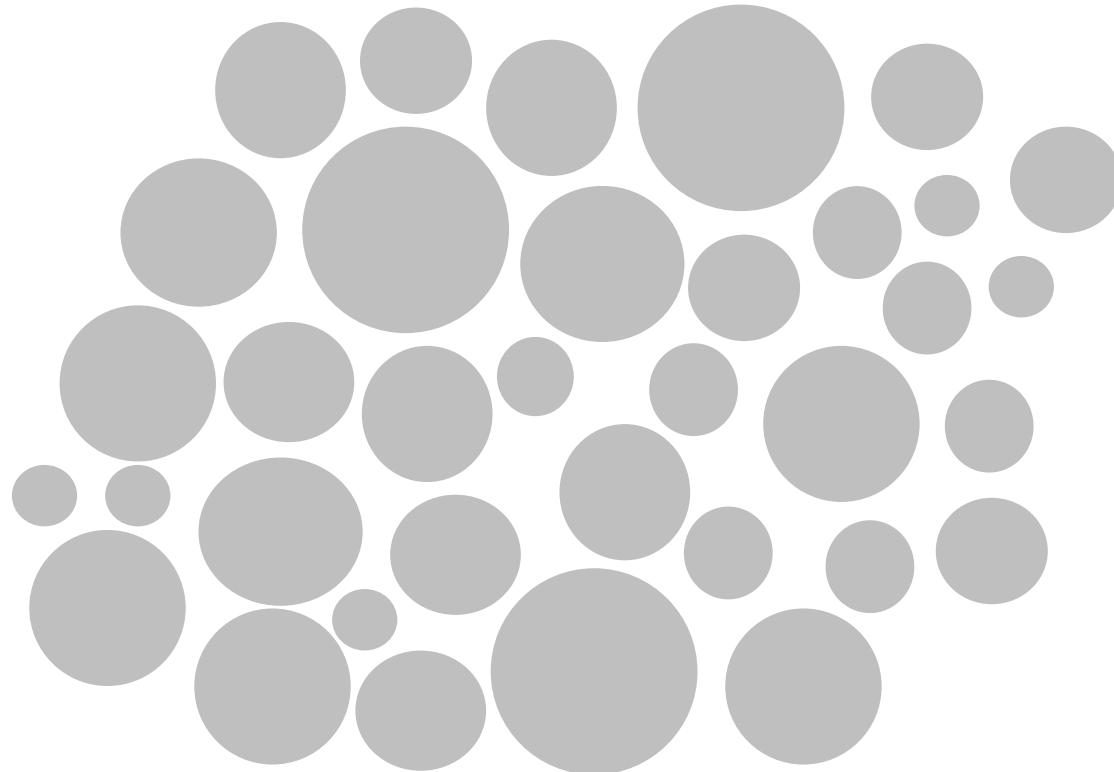




# Zalando Tech

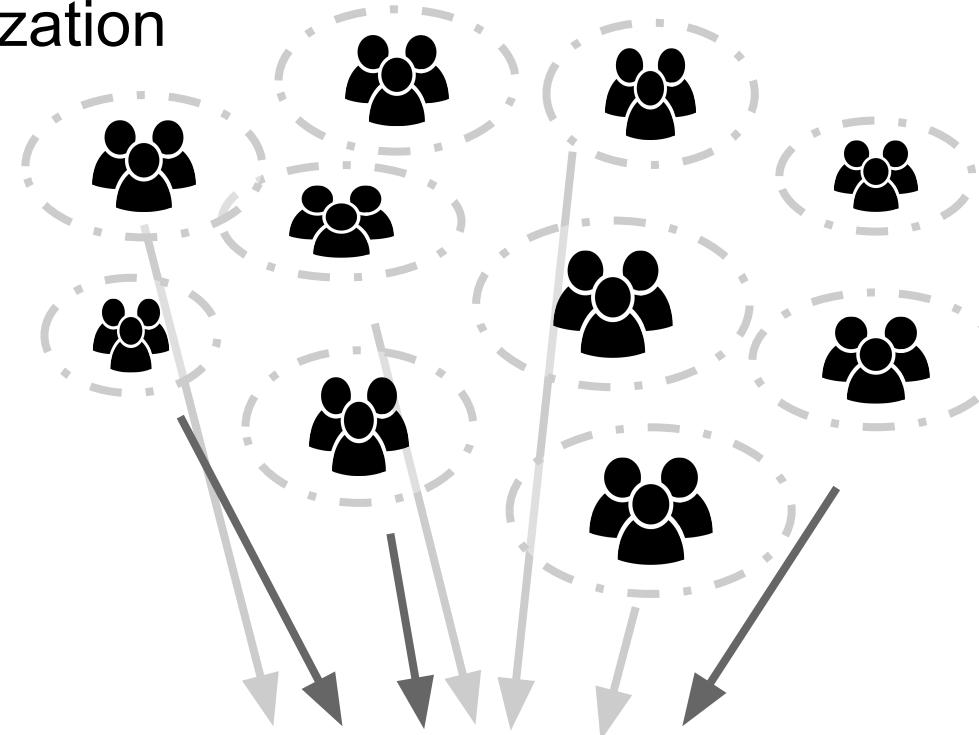
# From monolith to microservice architecture

**Reorganization**  
**> 1000**  
**microservices**



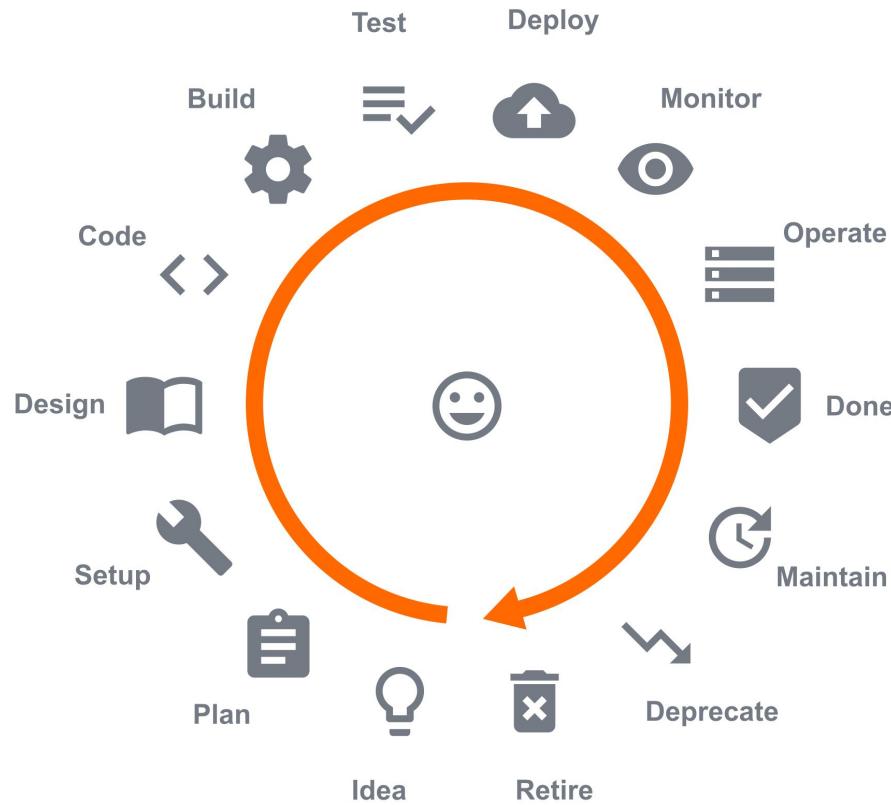
# Tech organization

> 1100  
developers



> 200  
development  
teams

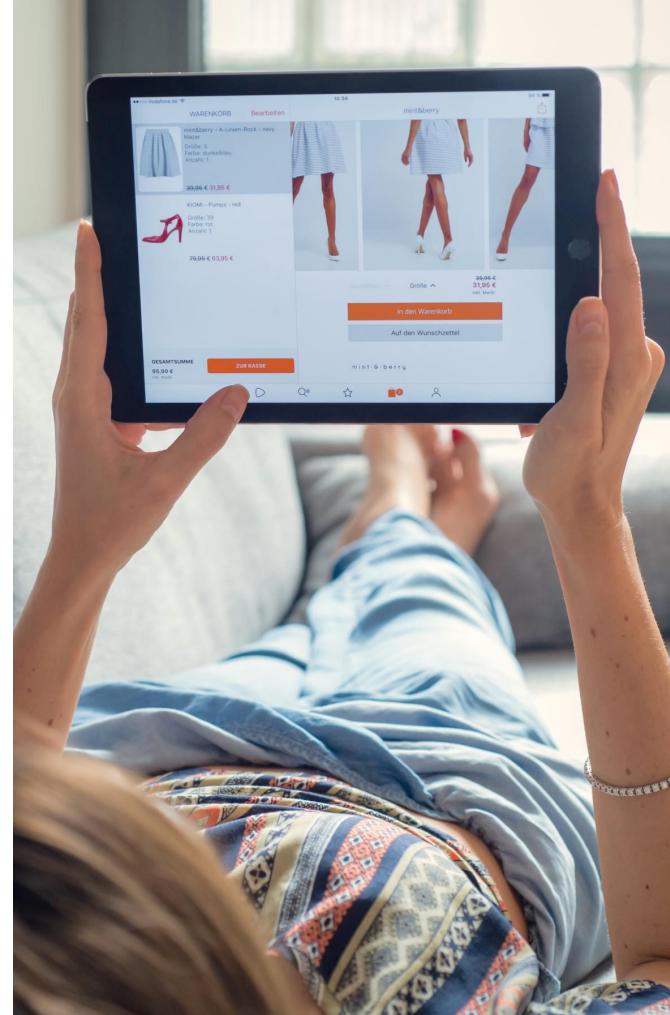
# End to end responsibility



# Checkout

## Goal

“Allow customers to buy seamlessly and conveniently”



# Checkout landscape

**Java  
Scala  
Node JS**

programming languages

**Cassandra**

data storage

**ETCD**

configurations

**REST & messaging**

Communication

**AWS**

&

**Kubernetes**

infrastructure

**React**

client side

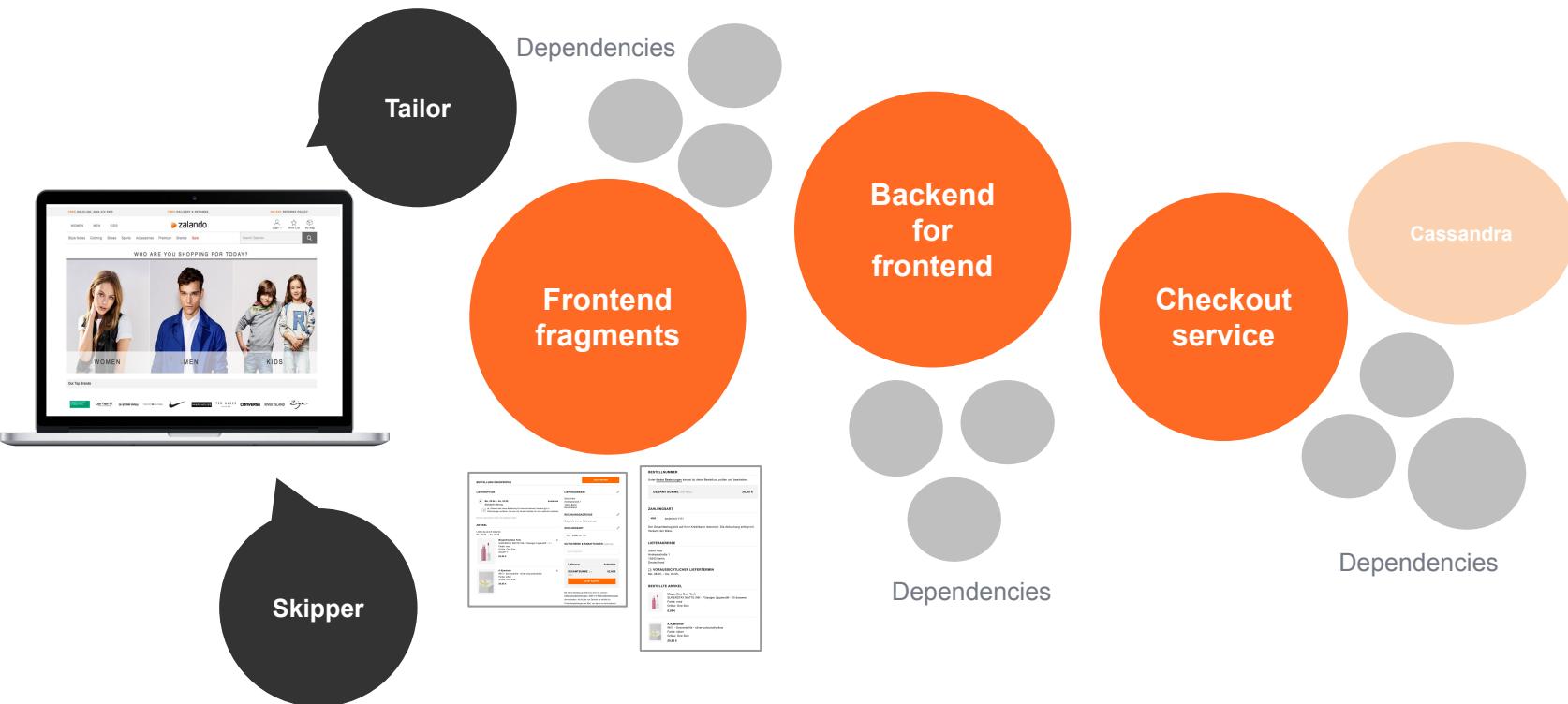
**Docker**

container

**Many**

more

# Checkout architecture



# Checkout is a **critical** component in the shopping journey

- Direct impact in business revenue
- Direct impact in customer experience

# Checkout challenges in a microservice ecosystem

- Increase points of **failures**
- Multiple **dependencies** evolving independently

# **Lessons learnt building Checkout with**

- Reliability patterns
- Scalability
- Monitoring

# Building microservices with reliability patterns

# Checkout confirmation page

Delivery Service

Cart

BESTELLUNG ÜBERPRÜFEN

JETZT KAUFEN

LIEFEROPTION

Mo, 29.04. – Do, 02.05. Standard-Lieferung kostenlos

Ja, Zalando darf meine Bestellung für einen schnelleren Versand ggf. in Teilsendungen aufteilen. Service und Versand bleiben für mich weiterhin kostenlos.

Express-Lieferung kann leider nicht angeboten werden.

ARTIKEL

Lieferung durch Zalando  
Mo, 29.04. – Do, 02.05.

	<b>Maybelline New York</b> SUPERSTAY MATTE INK - Flüssiger Lippenstift - 1... Farbe: rosa Größe: One Size Anzahl: 7 <b>62,65 €</b>
	<b>A.Kjærbede</b> WIVI - Sonnenbrille - silver-coloured/yellow Farbe: silber Größe: One Size <b>29,95 €</b>

LIEFERADRESSE

David Halo  
Andreastraße 1  
10243 Berlin  
Deutschland

RECHNUNGSADRESSE

Entspricht meiner Lieferadresse

ZAHLUNGSART

VISA endet mit 1111

GUTSCHEINE & RABATTCODES (optional)

Code eingeben

Lieferung kostenlos

GESAMTSUMME (inkl. MwSt.) **92,60 €**

JETZT KAUFEN

Delivery Destination

Payments Service

Mit deiner Bestellung erklärst du dich mit unseren [Datenschutzbestimmungen](#), [AGB](#) und [Widerrufsbestimmungen](#) einverstanden. Als Kunde von Zalando.de erhältst du Produktempfehlungen per Mail, von denen du dich jederzeit

# Checkout confirmation page

Delivery Service

BESTELLUNG ÜBERPRÜFEN

JETZT KAUFEN

LIEFEROPTION

Mo, 29.04. – Do, 02.05.  
Standard-Lieferung

kostenlos

Ja, Zalando darf meine Bestellung für einen schnelleren Versand ggf. in Teilsendungen aufteilen. Service und Versand bleiben ~~für mich~~ weiterhin kostenlos.

Express-Lieferung kann leider nicht angeboten werden.

ARTIKEL

Lieferung durch Zalando  
Mo, 29.04. – Do, 02.05.

**Maybelline New York**  
SUPERSTAY MATTE INK - Flüssiger Lippenstift - 1...  
Farbe: rosa  
Größe: One Size  
Anzahl: 7  
**62,65 €**

**A.Kjærbede**  
WIVI - Sonnenbrille - silver-coloured/yellow  
Farbe: silber  
Größe: One Size  
**29,95 €**

LIEFERADRESSE

David Halo  
Andreastraße 1  
10243 Berlin  
Deutschland

RECHNUNGSADRESSE

Entspricht meiner Lieferadresse

ZAHLUNGSArt

VISA endet mit 1111

GUTSCHEINE & RABATTCODES (optional)

Code eingeben

Lieferung      kostenlos

GESAMTSUMME (inkl. MwSt.) **92,60 €**

JETZT KAUFEN

Mit deiner Bestellung erklärst du dich mit unseren [Datenschutzbestimmungen](#), [AGB](#) und [Widerrufsbestimmungen](#) einverstanden. Als Kunde von Zalando.de erhältst du Produktempfehlungen per Mail, von denen du dich jederzeit

# Unwanted error



## Entschuldige, uns ist etwas verloren gegangen

Es sieht so aus, als wurde diese Seite verschoben oder gelöscht. Wähle aus, für wen du shoppen möchtest und wir bringen dich zurück auf die Spur.

DAMEN

HERREN

KINDER

# Doing retries

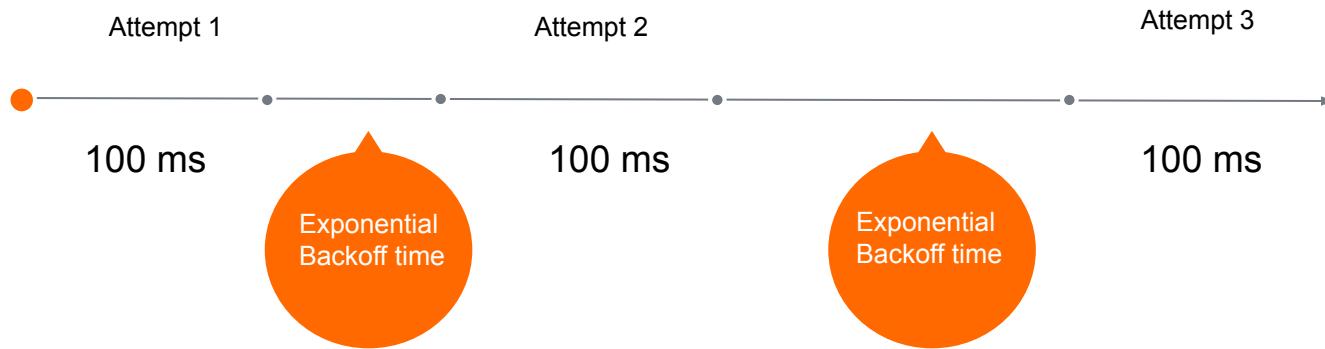
```
for (var i = 1; i < numRetries; i++) {
  try {
    return getDeliveryOptionsForCheckout(cart)
  } catch(error) {
    if (i >= numRetries) {
      throw error;
    }
  }
}
```

**Retry for transient errors  
like a network error  
or service overload**

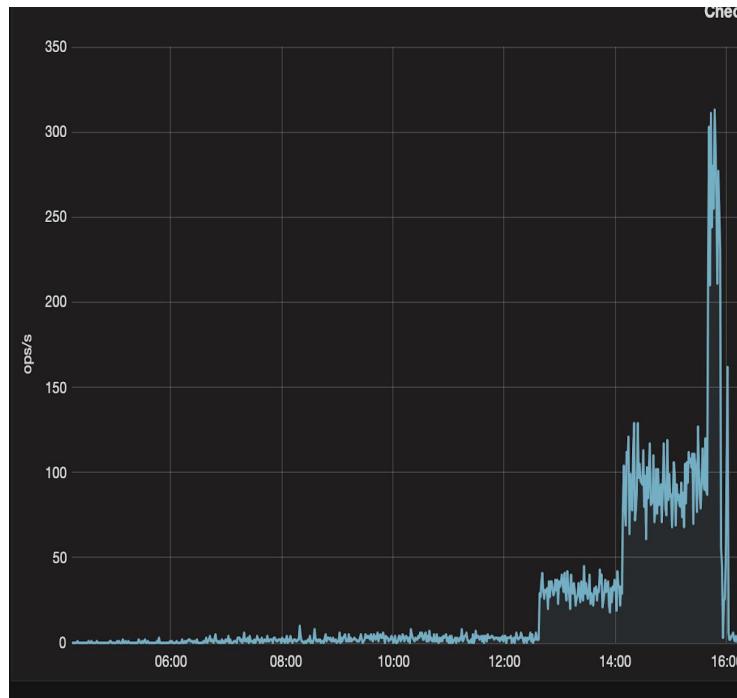
# Retries for some errors

```
try {
    getDeliveryOptionsForCheckout(cart) match {
        case Success() => // return result
        case TransientFailure => // retry operation
        case Error => // throw error
    }
}
catch {
    println("Delivery options exception")
}
```

# Retries with exponential backoff

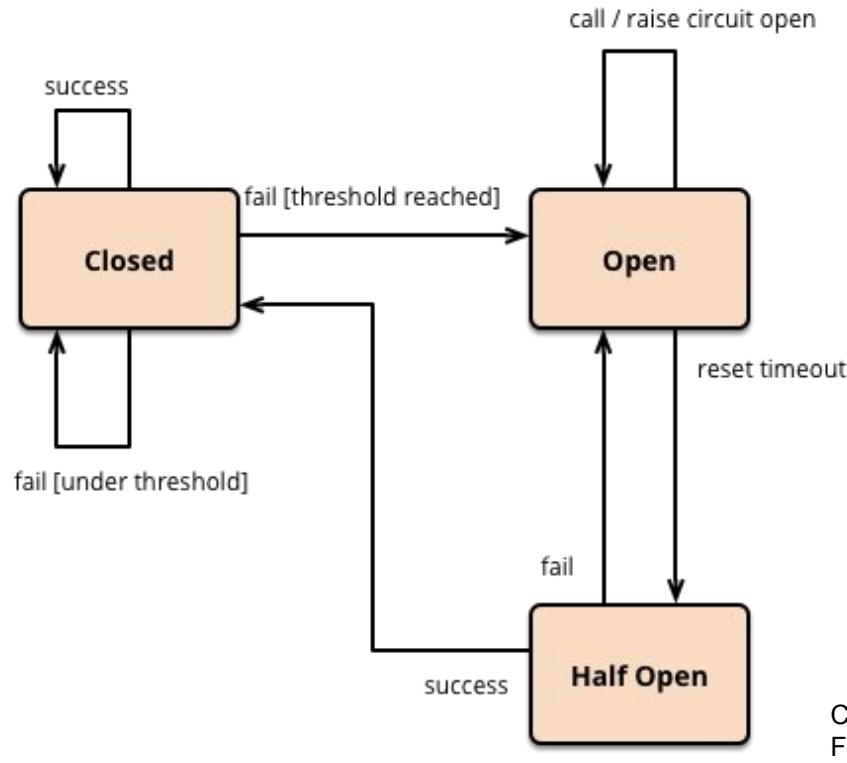


# Exhaustion of retries and failures become permanent



Prevent execution of  
operations that are  
likely to fail

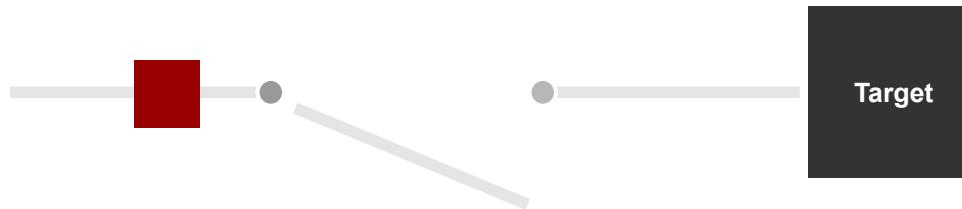
# Circuit breaker pattern



Circuit breaker pattern - Martin Fowler blog post

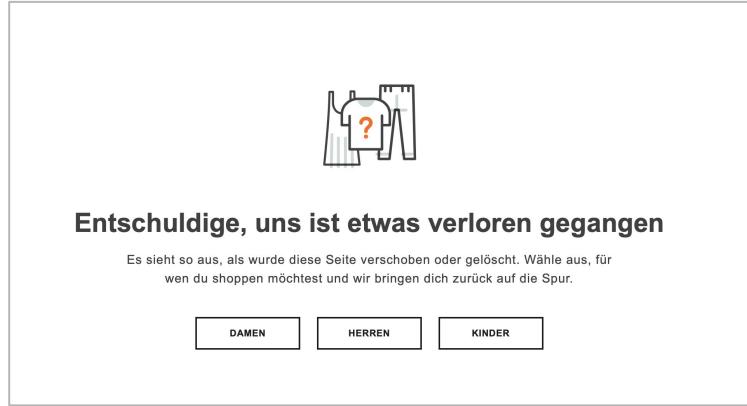
# Open circuit, operations fails immediately

error rate > threshold 50%



getDeliveryOptionsForCheckout = failure

# Fallback as alternative of failure



*Unwanted failure: no Checkout*

The screenshot shows a checkout page with a red circle highlighting the delivery options section. It includes fields for "LIEFEROPTION" (Mo, 29.04. – Do, 02.05. Standard-Lieferung), "LIEFERADRESSE" (David Hahn, Sandrestraße 1, 1243 Berlin, Deutschland), "RECHNUNGSADRESSE" (Same as delivery address), "ZAHLUNGSArt" (VISA ending in 1111), and "GUTSCHEINE & RABATTCODES (optional)". The total "GESAMTSUMME" is 92,60 €. A note at the bottom states: "Mit deiner Bestellung erklärst du dich mit unseren Datenschutzbestimmungen, AGB und Widerrufsbestimmungen einverstanden. Als Kunde von Zalando.de erhältst du Produktempfehlungen per Mail, von denen du dich jederzeit abmelden kannst." At the bottom right is a large orange "JETZT KAUFEN" button.

*Fallback: Only **Standard** delivery service with a default delivery promise*

# Putting all together

Do **retries** of operations with **exponential backoff**

Wrap operations with a **circuit breaker**

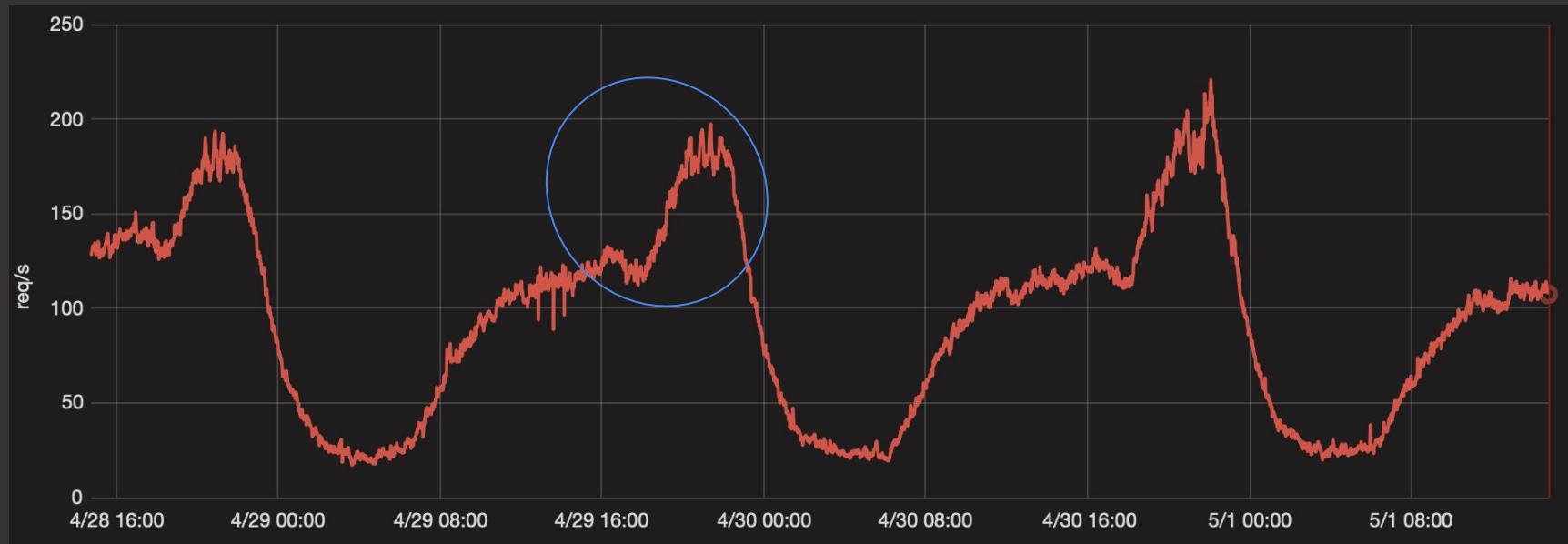
Handle failures with **fallbacks** when possible

Otherwise make sure to **handle the exceptions**

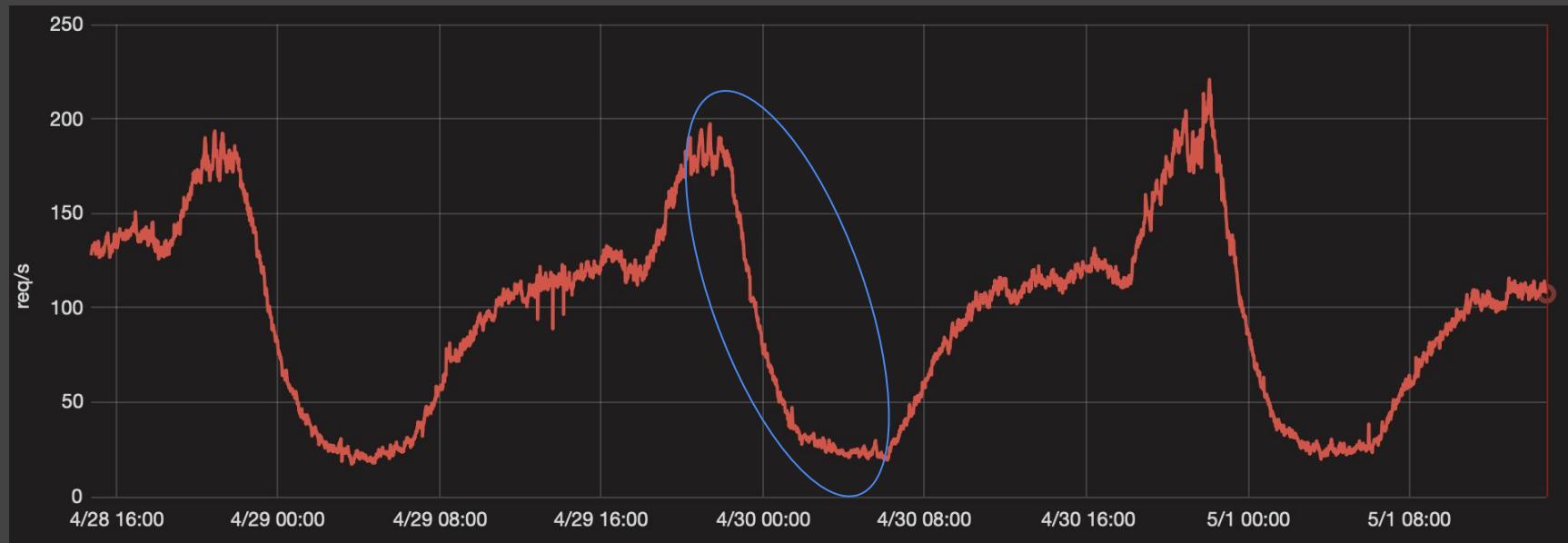
```
circuitCommand(  
    getDeliveryOptionsForCheckout(cart)  
    .retry(2)  
)  
.onSuccess("//do something with result")  
.onError(getDeliveryOptionsForCheckoutFallback)
```

# Scaling microservices

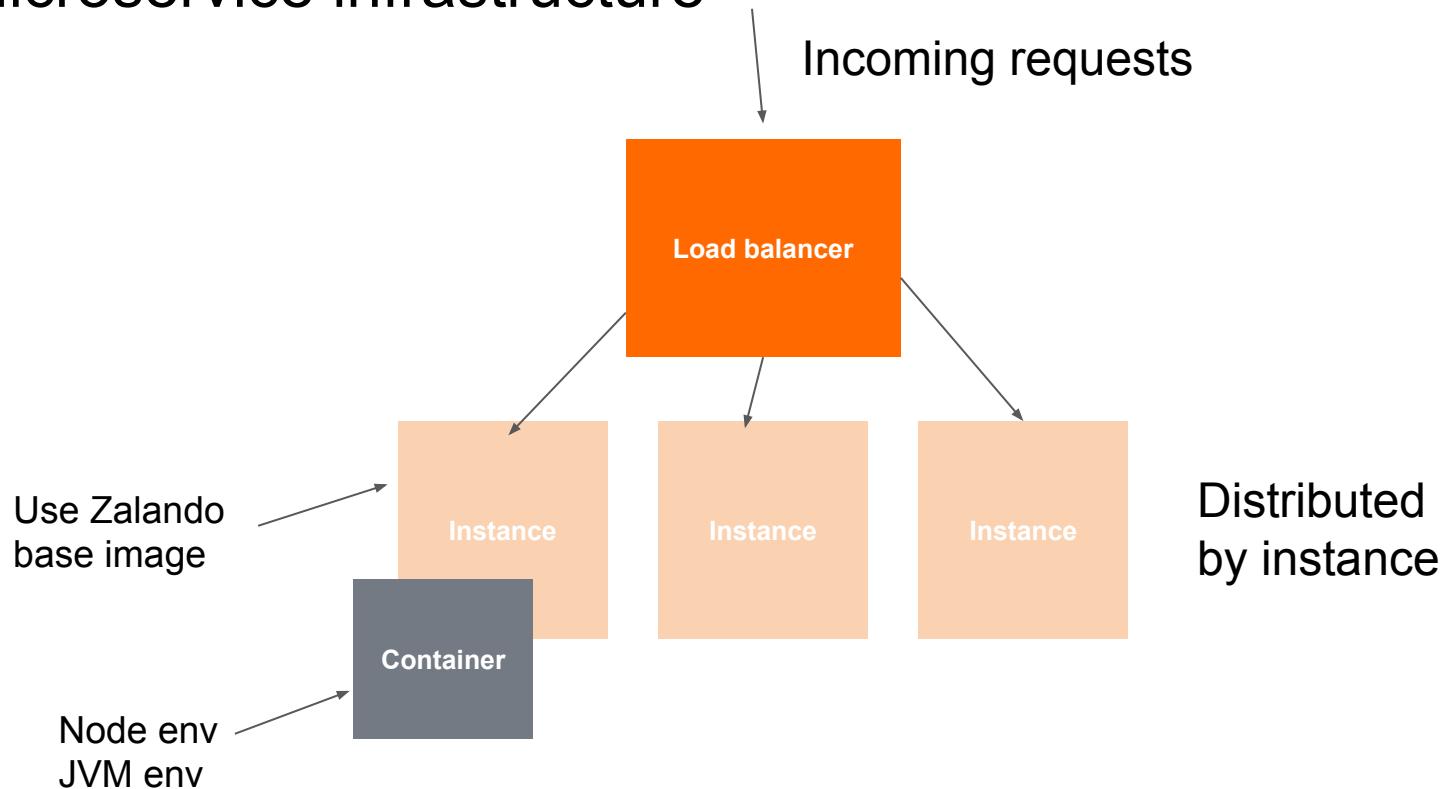
# Traffic pattern



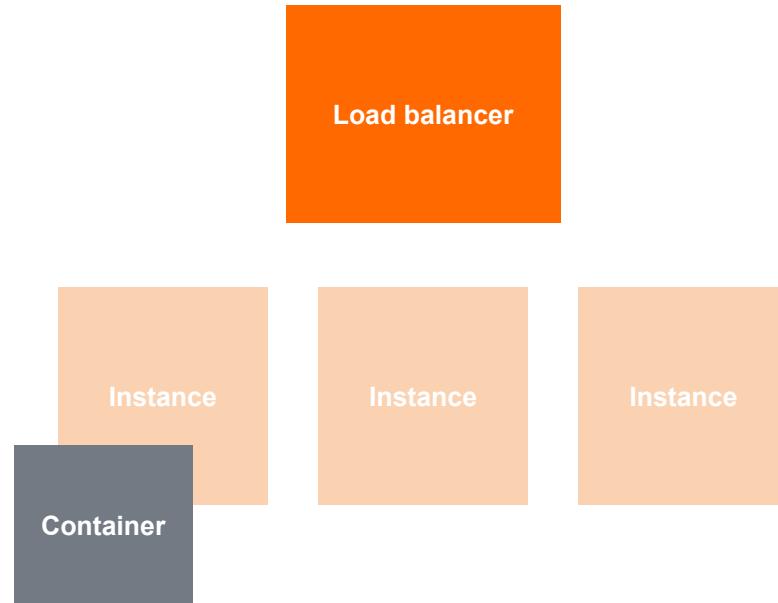
# Traffic pattern



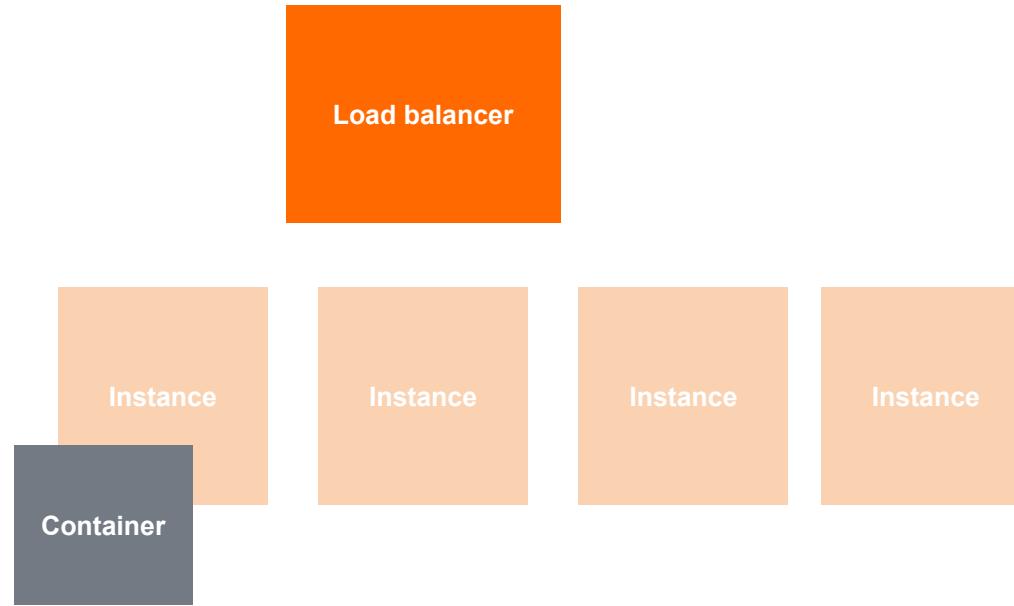
# Microservice infrastructure



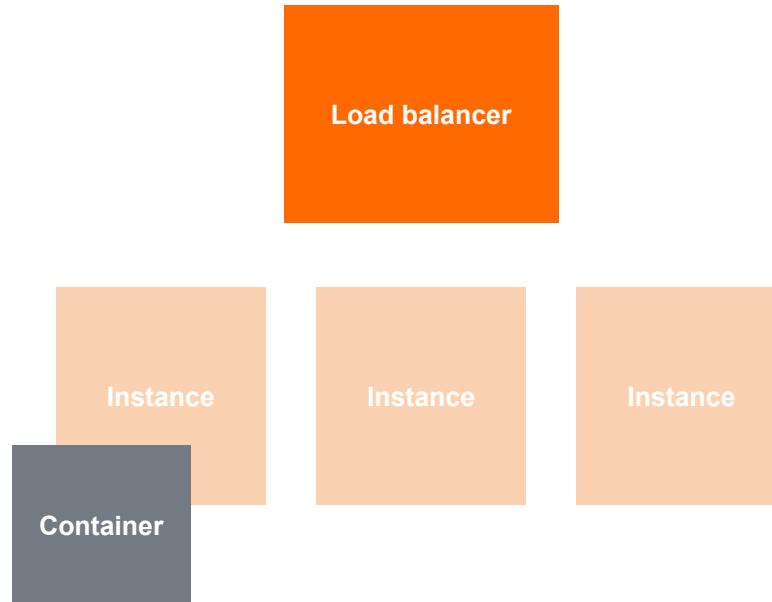
# Scaling horizontally



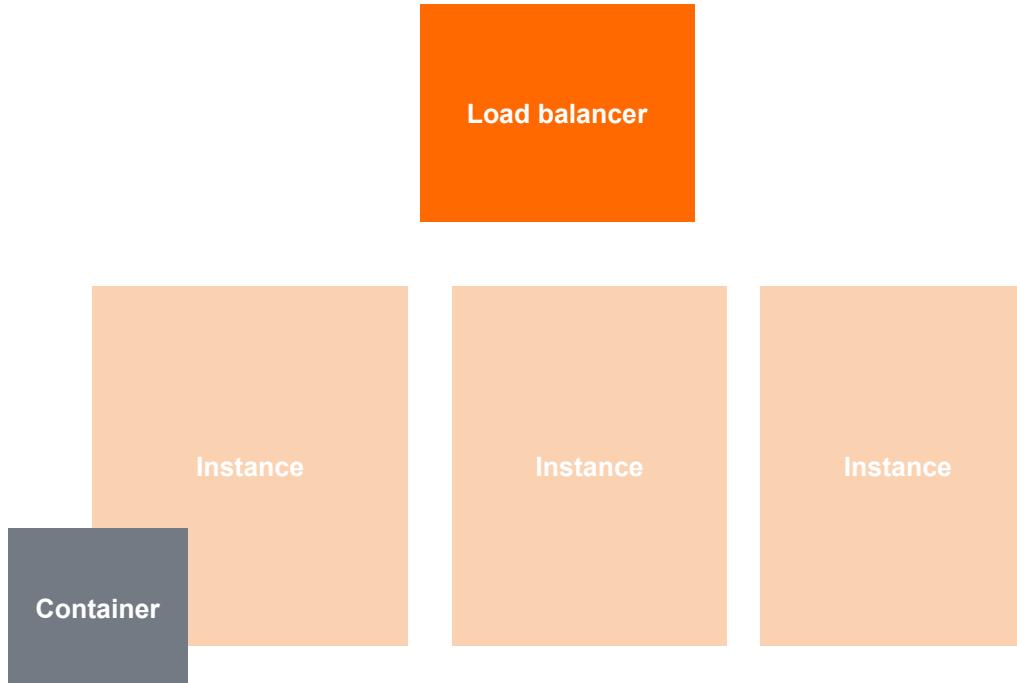
# Scaling horizontally



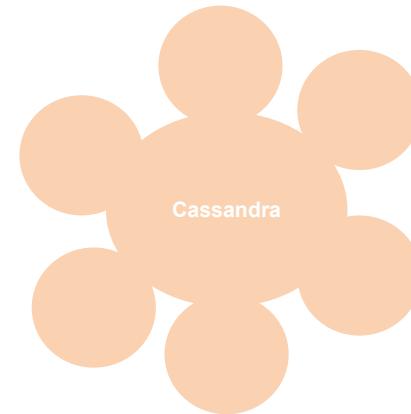
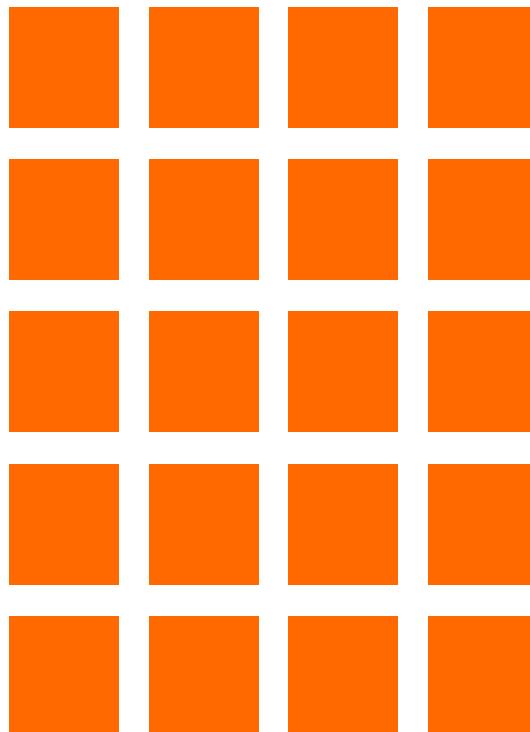
# Scaling vertically



# Scaling vertically



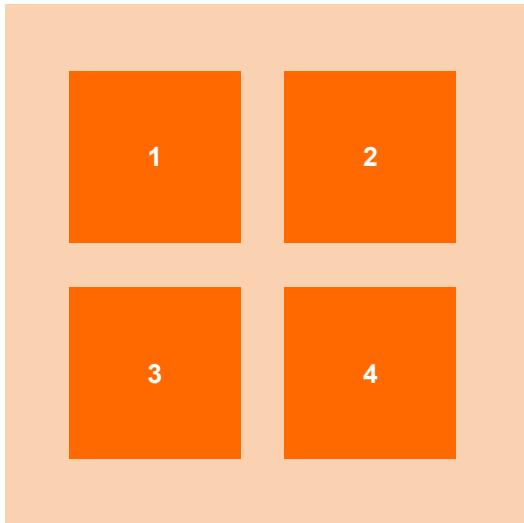
# Scaling consequences



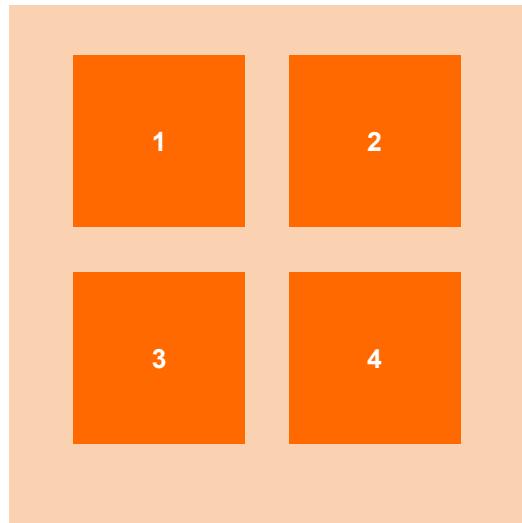
> service connections  
> saturation and risk of  
unhealthy database

**Microservices cannot be  
scalable if downstream  
microservices cannot scale**

# Low traffic rollouts

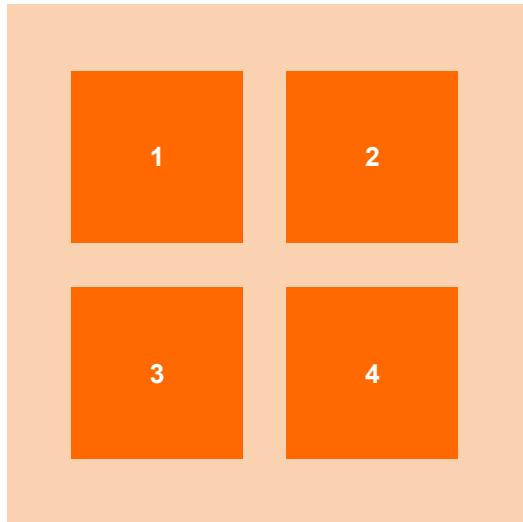


*Service v2  
Traffic 0%*

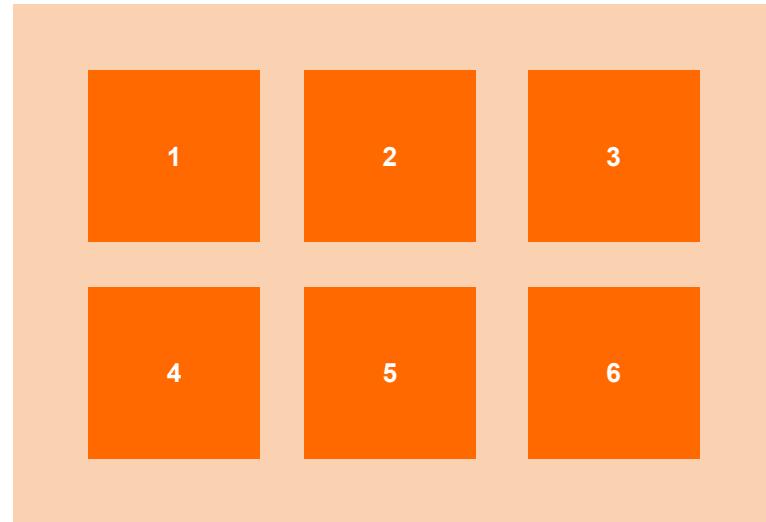


*Service v1  
Traffic 100%*

# High traffic rollouts

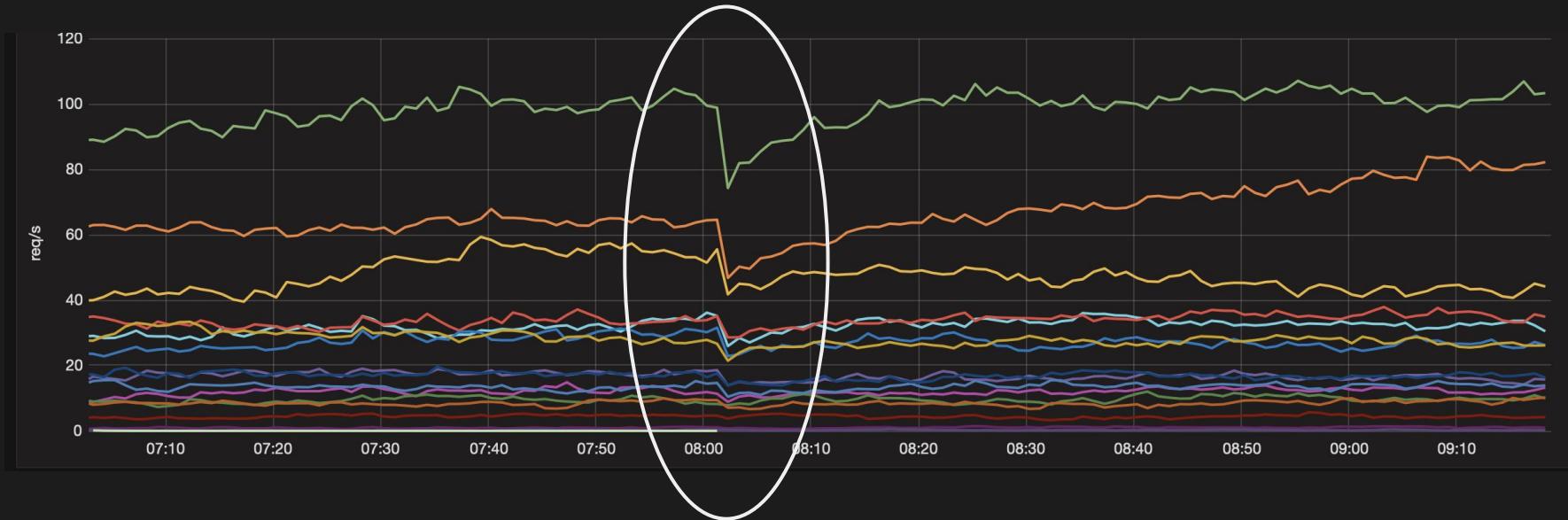


*Service v2*  
*Traffic 0%*



*Service v1*  
*Traffic 100%*

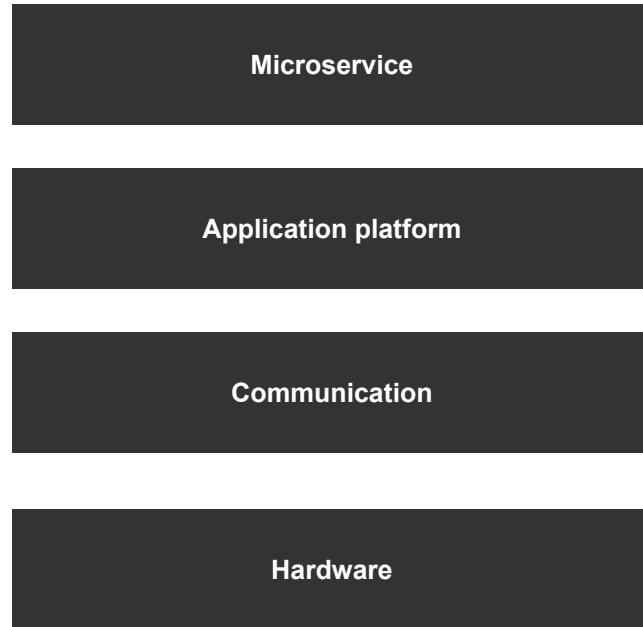
# Rollout with not enough capacity



**Rollouts should consider  
allocate same capacity like  
version with 100% traffic**

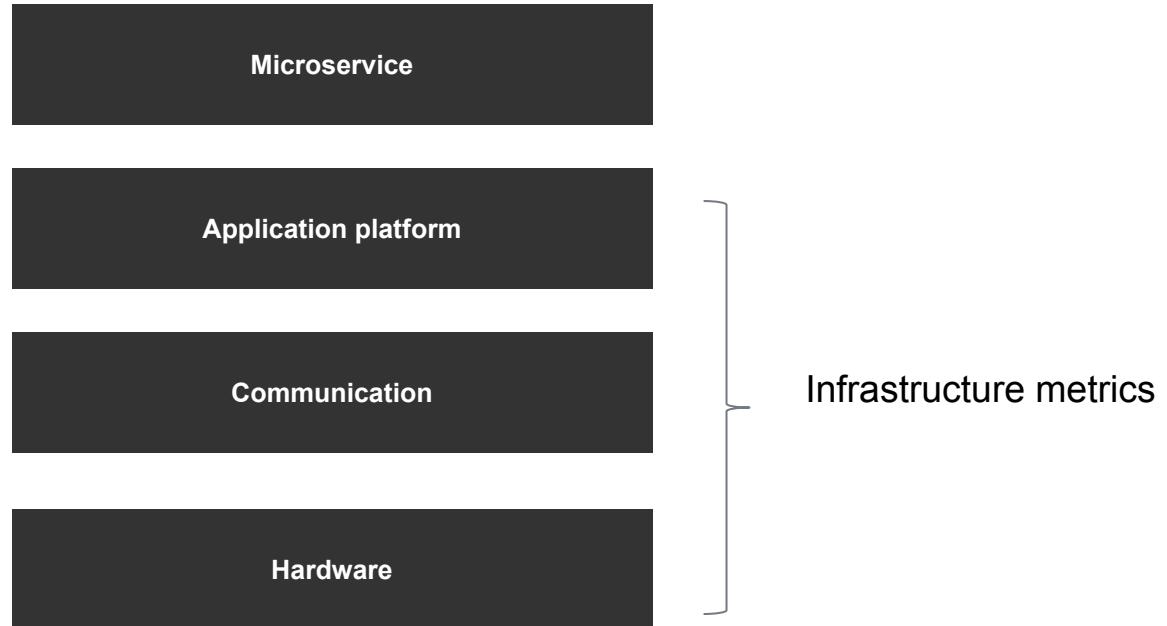
# Monitor microservices

# Monitoring microservice ecosystem



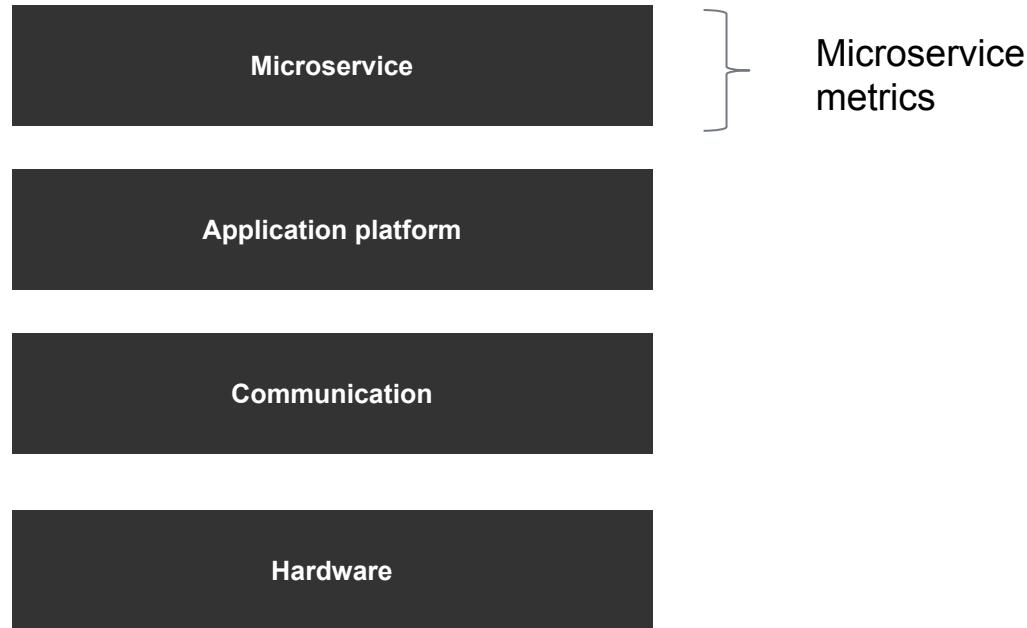
*Four layer model of microservice ecosystem*

# Monitoring microservice ecosystem



*For layer model of microservice ecosystem*

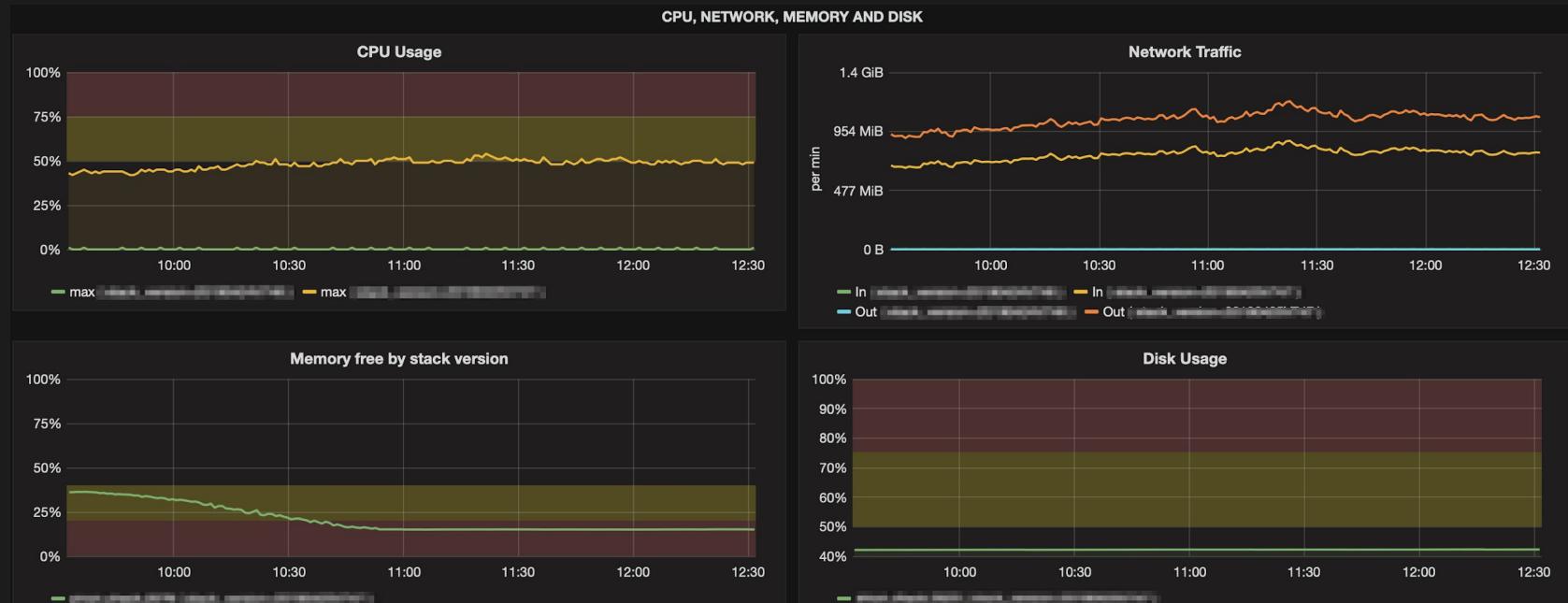
# Monitoring microservice ecosystem



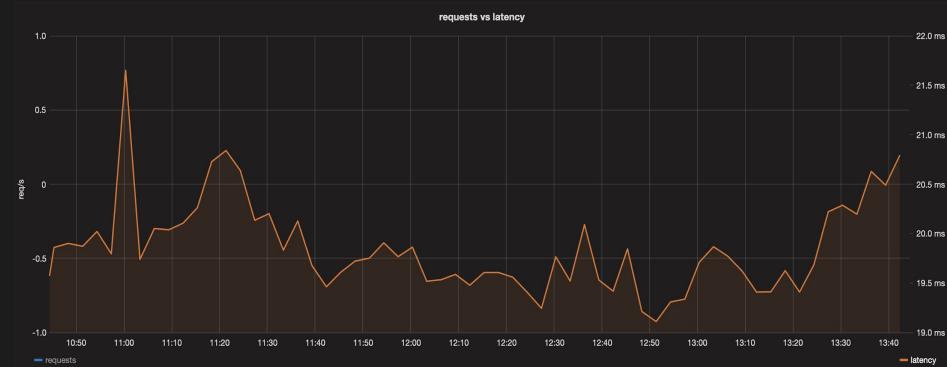
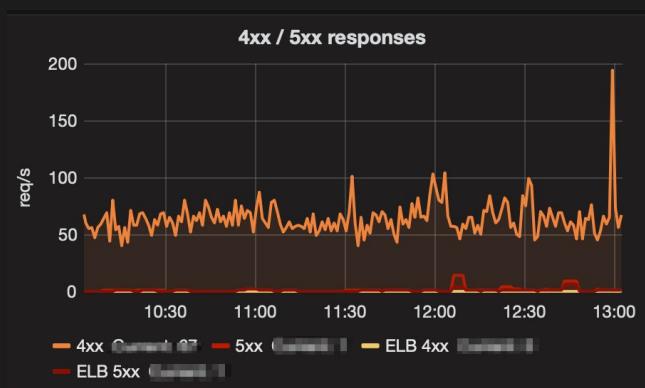
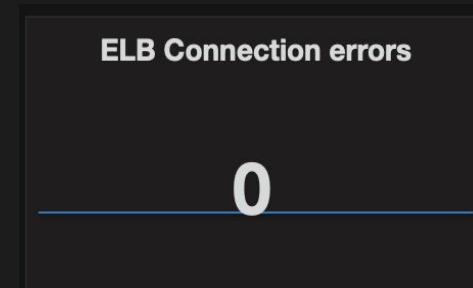
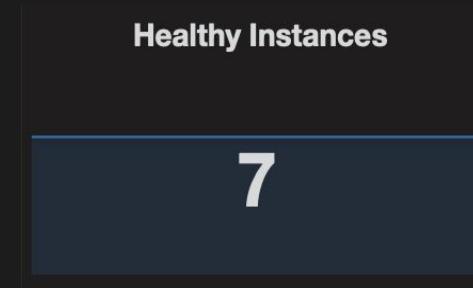
*For layer model of microservice ecosystem*

# **First example**

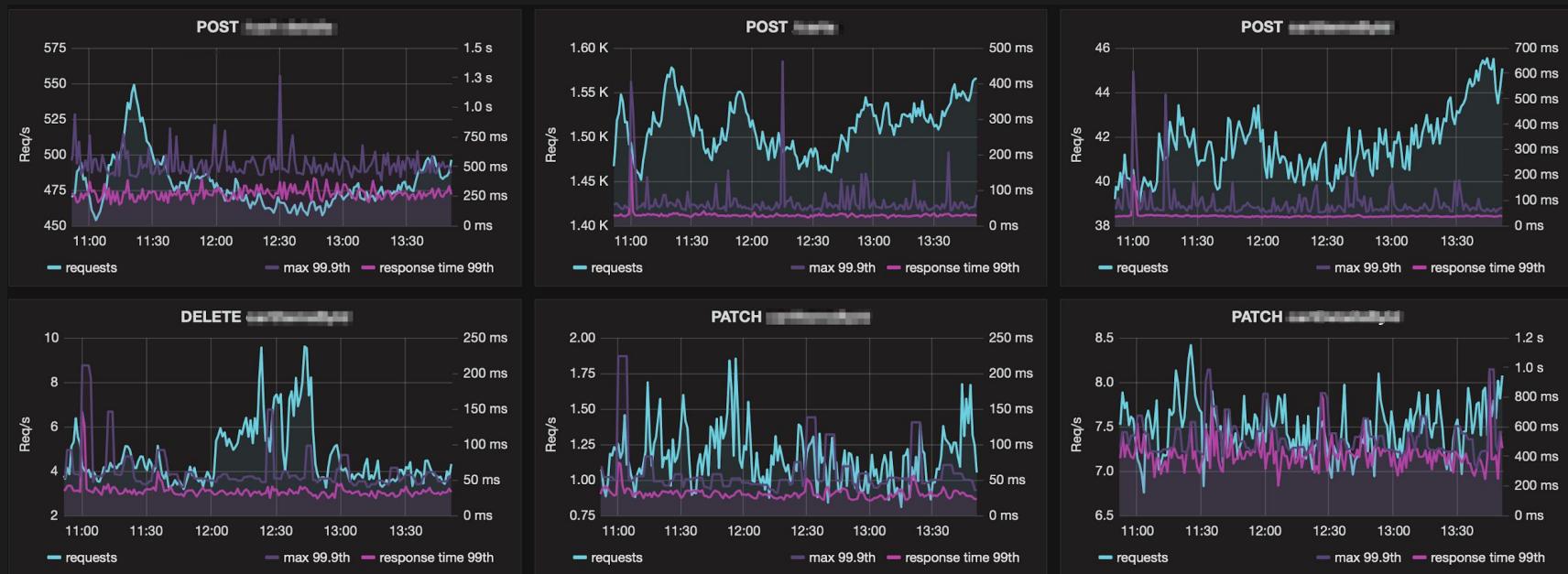
# Hardware metrics



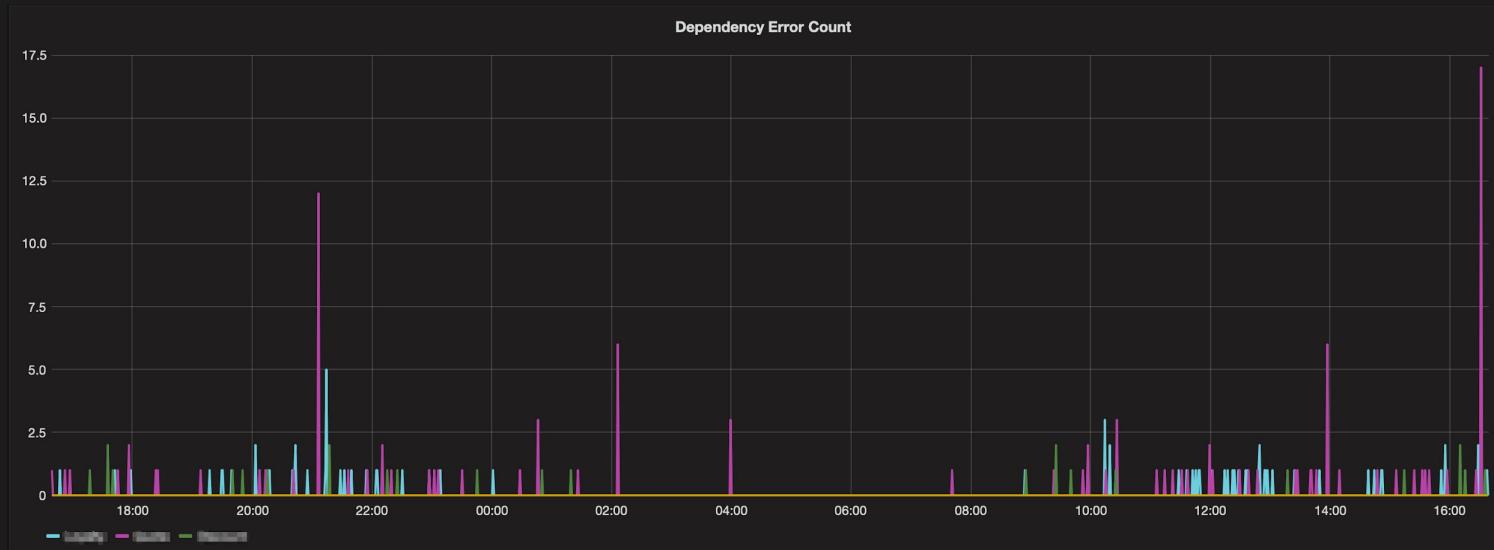
# Communication metrics



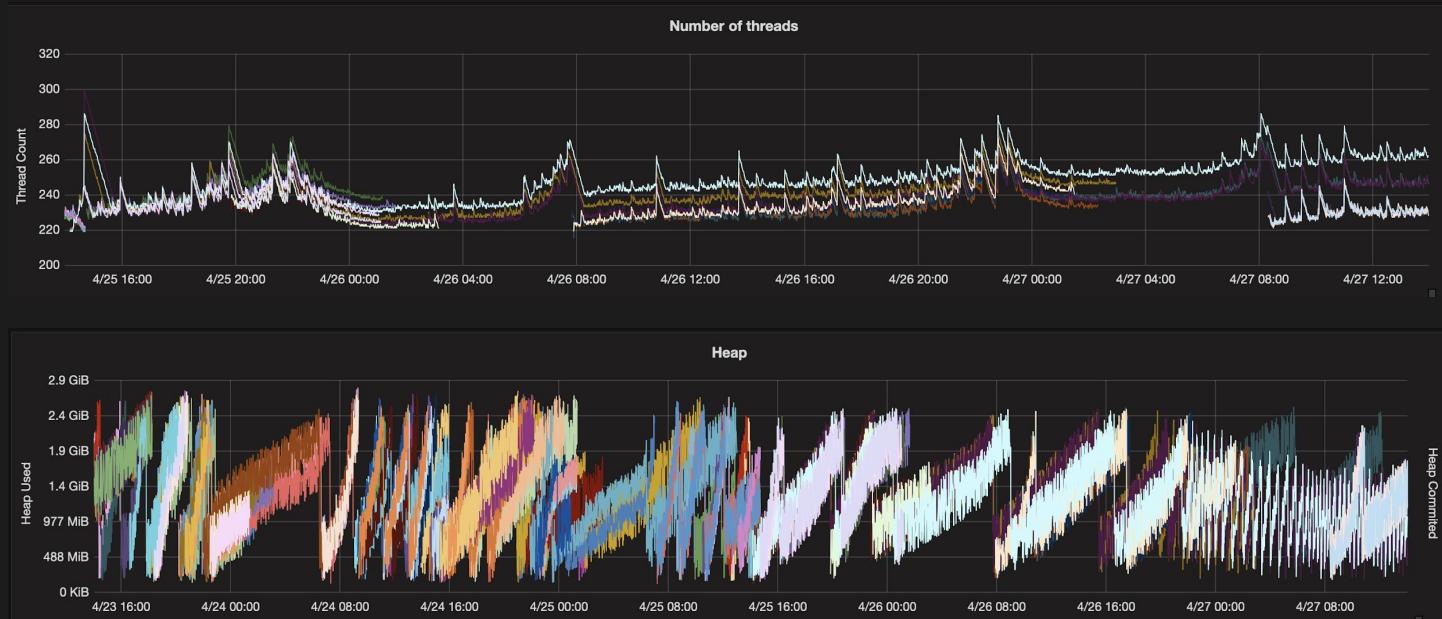
# Rate and responses of API endpoints



# Dependencies metrics

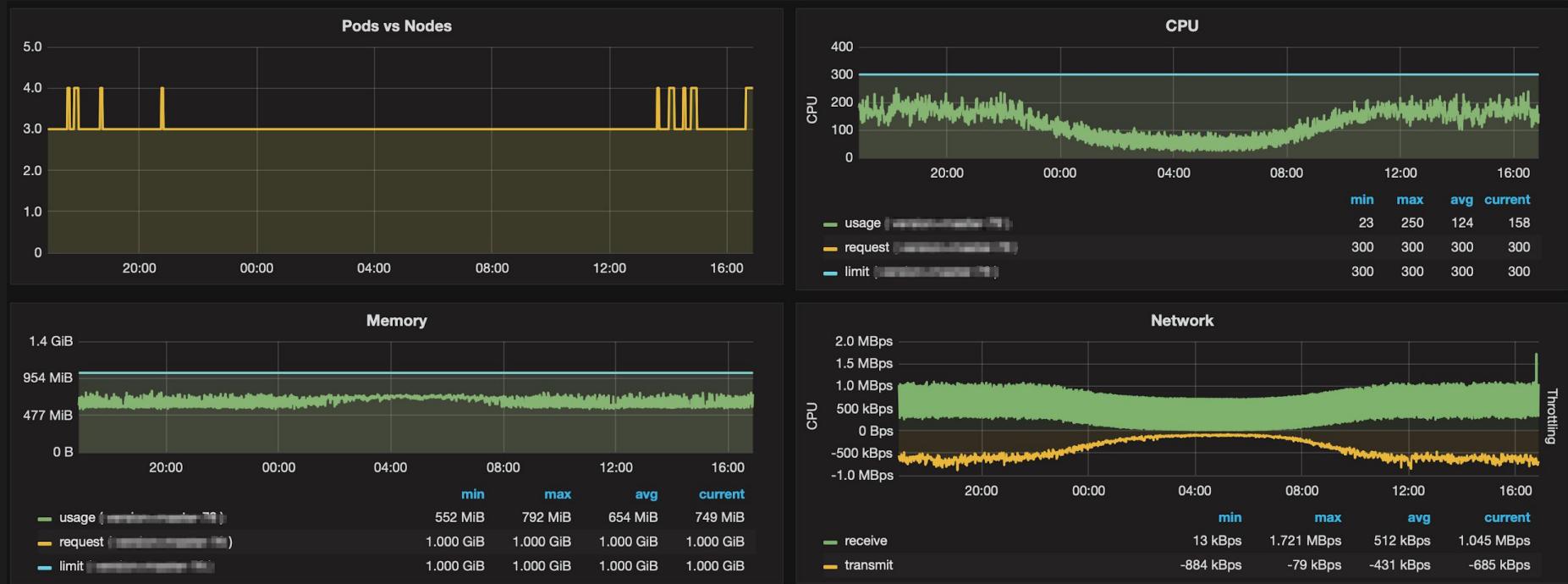


# Language specific metrics

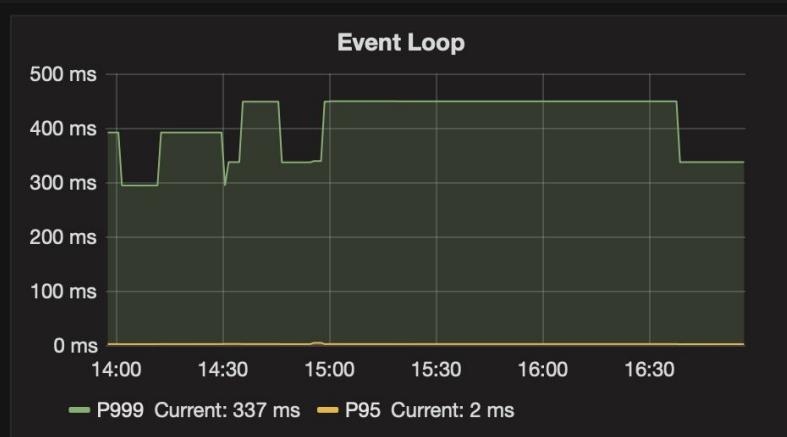
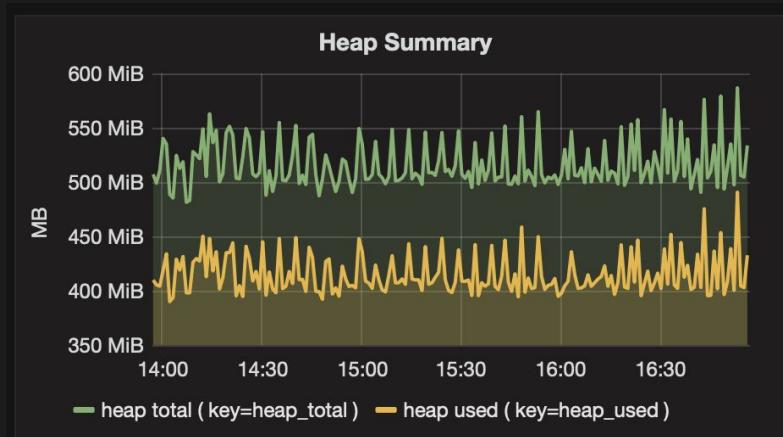
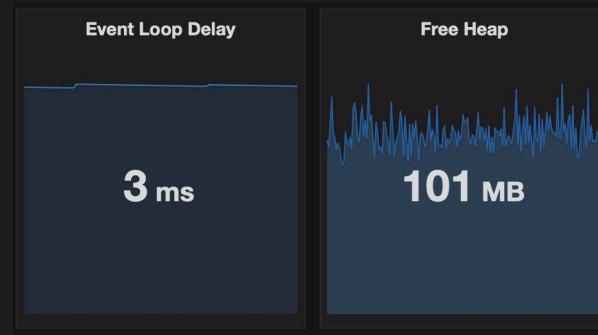


# **Second Example**

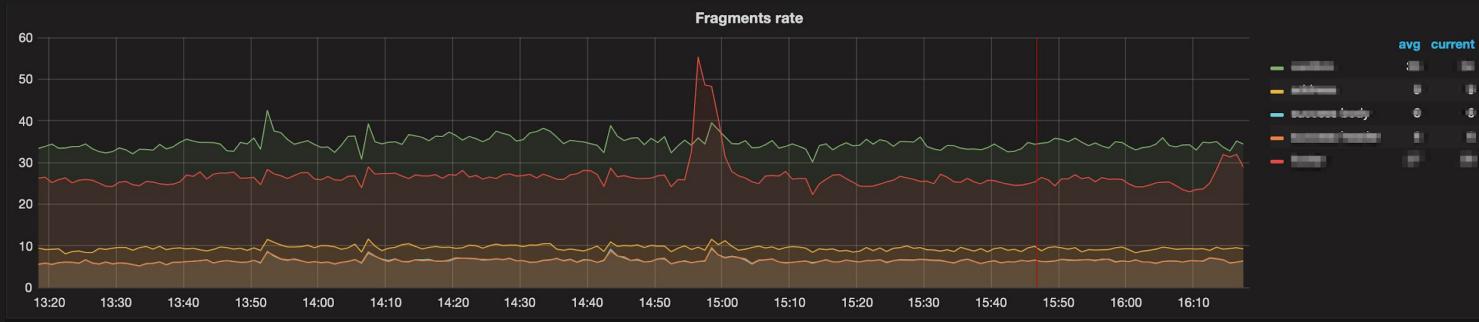
# Infrastructure metrics



# Node JS metrics

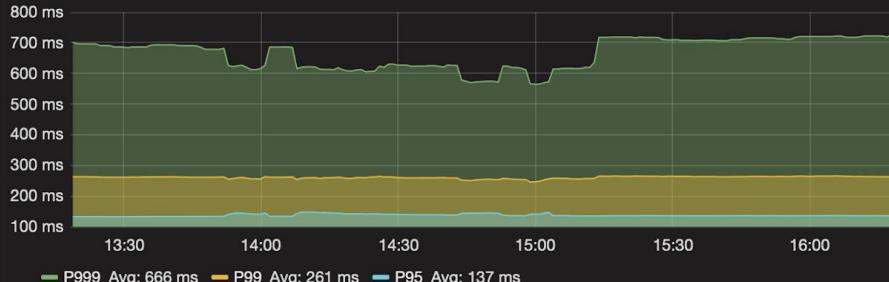


# Frontend microservice metrics

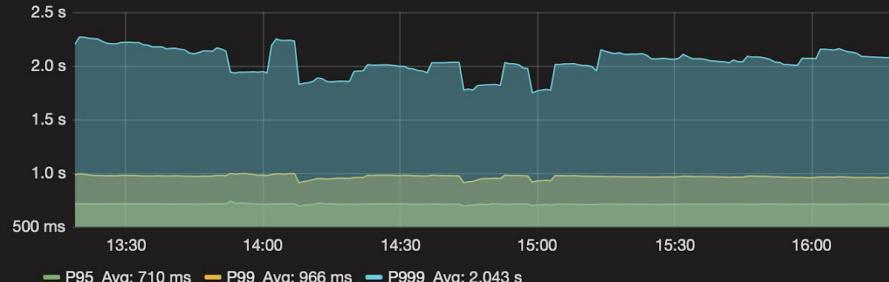


## FRAGMENTS TIMINGS

Render ██████████ - Percentile



Render ██████████ - Percentile



# Anti pattern: Dashboard usage for outage detection

# Alerting

“Something is broken, and somebody needs to fix it right now! Or, something might break soon, so somebody should look soon.”

Practical Alerting - Monitoring distributed systems  
Google SRE Book

Alert

## **Unhealthy instances 1 of 5**

No more memory, JVM is misconfigured

## Alert

**Service checkout is returning 4XXs  
responses above threshold 25%**

Recent change broke contract of API for  
unconsidered business rule

Alert

## No orders in last 5 minutes

Downstream dependency is  
experimenting connectivity issues

## Alert

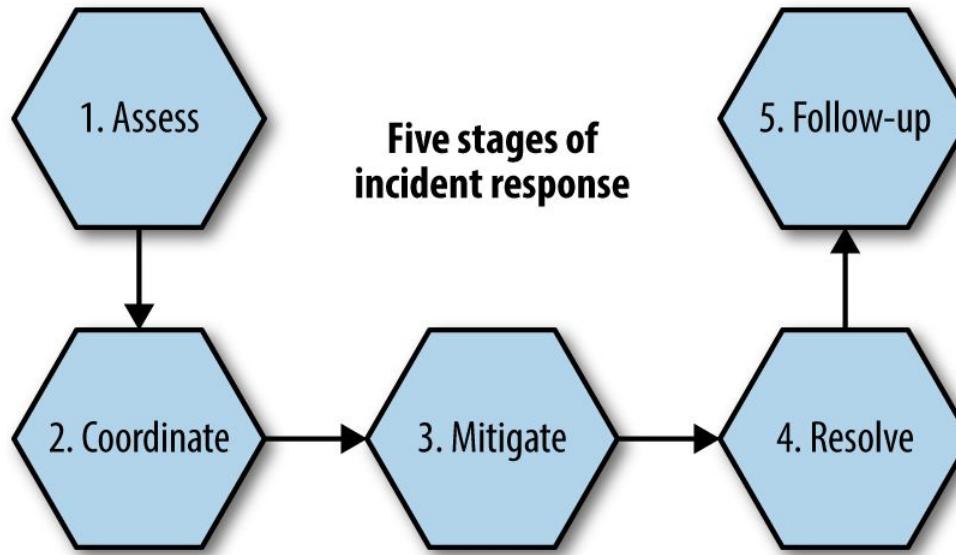
**Checkout database disk utilization is  
80%**

Saturation of data storage by an increase  
in traffic

Alerts notify about  
symptoms

Alerts should be  
actionable

# Incident response



*Figure Five stages of incident response.  
Microservices ready to production*

# Example of postmortem

## Summary of incident

No orders in last 5 minutes 13.05.2019 between 16:00 and 16:45

## Impact of customers

2K customers could not complete checkout

## Impact of business

50K euros loss of order that could be completed

## Analysis of root cause

Why there was no orders?

## Action items

...

**Every incident should  
have postmortem**



# Preparing for Black Friday

- Business forecast
- Load testing of real customer journey
- Capacity planning

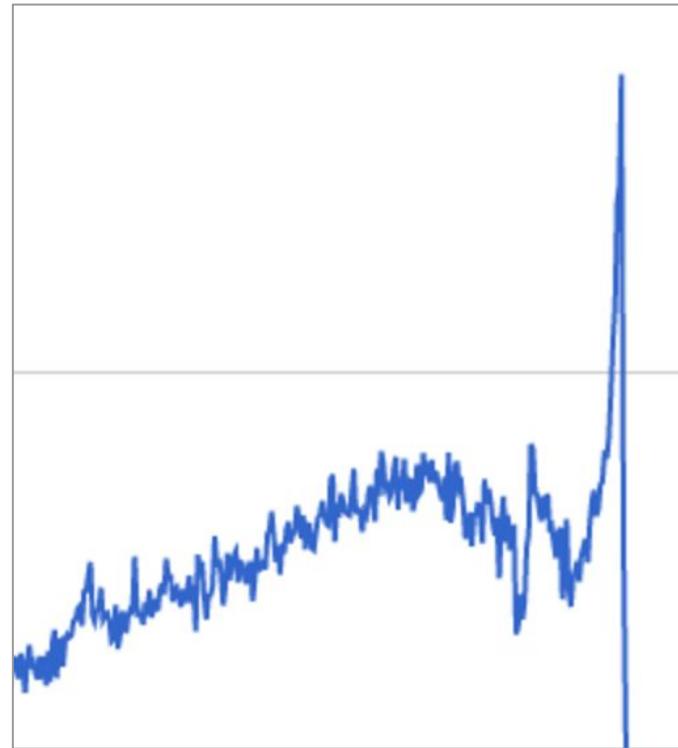
# Checklist for every **microservice** involved in Black Friday

- Is the **architecture** and **dependencies** reviewed?
- Are the possible **point of failures** identified and mitigated?
- Are **reliability patterns** implemented?
- Are the **configurations** **adjustable** without need of deployment?
- Do we have **scaling** strategy?
- Is **monitoring** in place?
- Are all **alerts** actionable?
- Is our team prepared for **24x7** incident management?

# Situation room



# Black Friday pattern of requests



> 4,200  
orders/m

## My summary of learnings

- Think outside the happy path and mitigate failures with reliability patterns
- Services are scalable proportionally with their dependencies
- Monitor the microservice ecosystem

# Resources

- [Service reliability engineering](#)
- [Production ready micro services](#)
- [Monitoring and alerting Tool used by Zalando](#)
- [Taylor](#)
- [Skipper](#)
- [Load testing in Zalando](#)
- [Kubernetes in Zalando](#)

**Obrigada  
Thank you  
Danke**

Contact

Pamela Canchanya  
[pam.cdm@posteo.net](mailto:pam.cdm@posteo.net)  
@pamcdm

# Building and running applications at scale in Zalando

Online fashion store Checkout case

By Pamela Canchanya