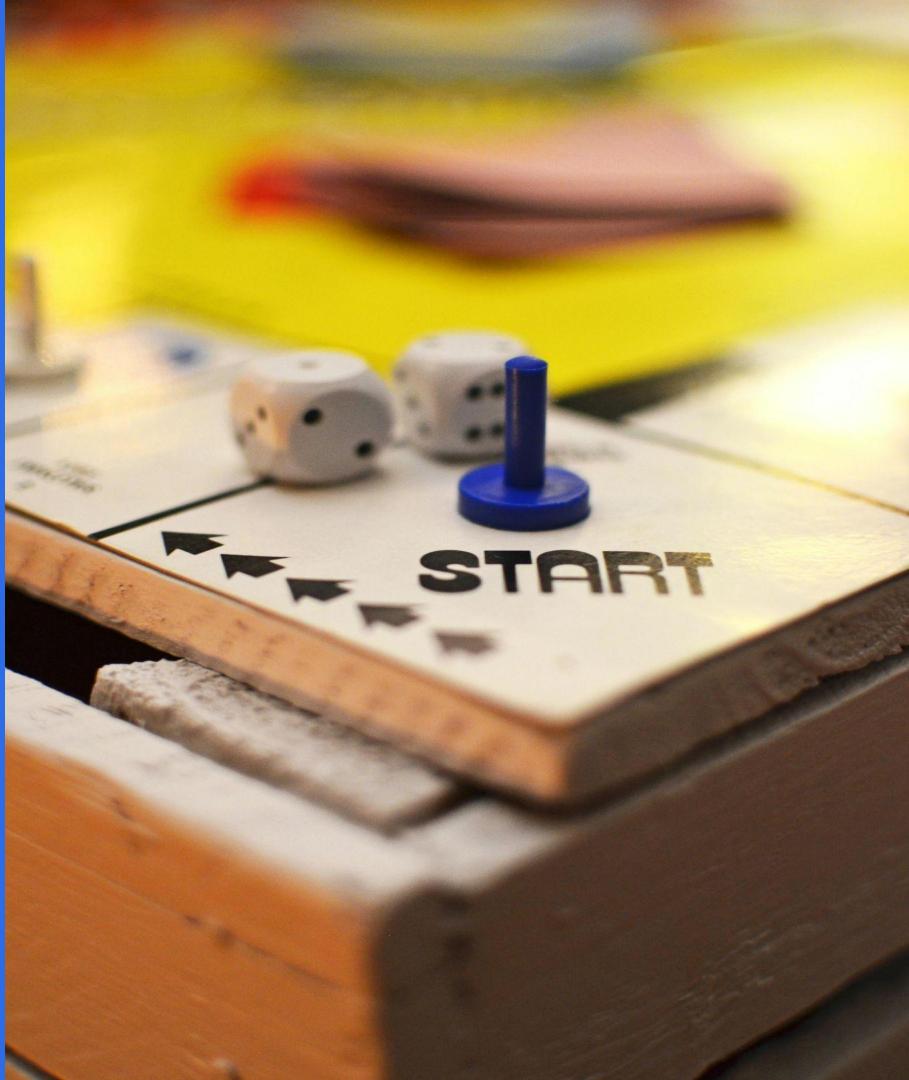




Austrian Platform  
Engineering  
Community  
Easy Dev Team Self  
Service

# Kontext





- Software Projects
- Agile Teams
- Customer specific solutions



- Software Experts
- Freelancer Network



- Product Information Management solutions



open**FORCE**

Entwickelt Produkte und Projekte für  
Kunden.

Kommen aus dem Java Enterprise  
Umfeld

Aktuell unterschiedliche Technologie  
Stacks

Betrieb für Development und Produktion



open**200**



open**DEVS**



THE pim**COMPANY**



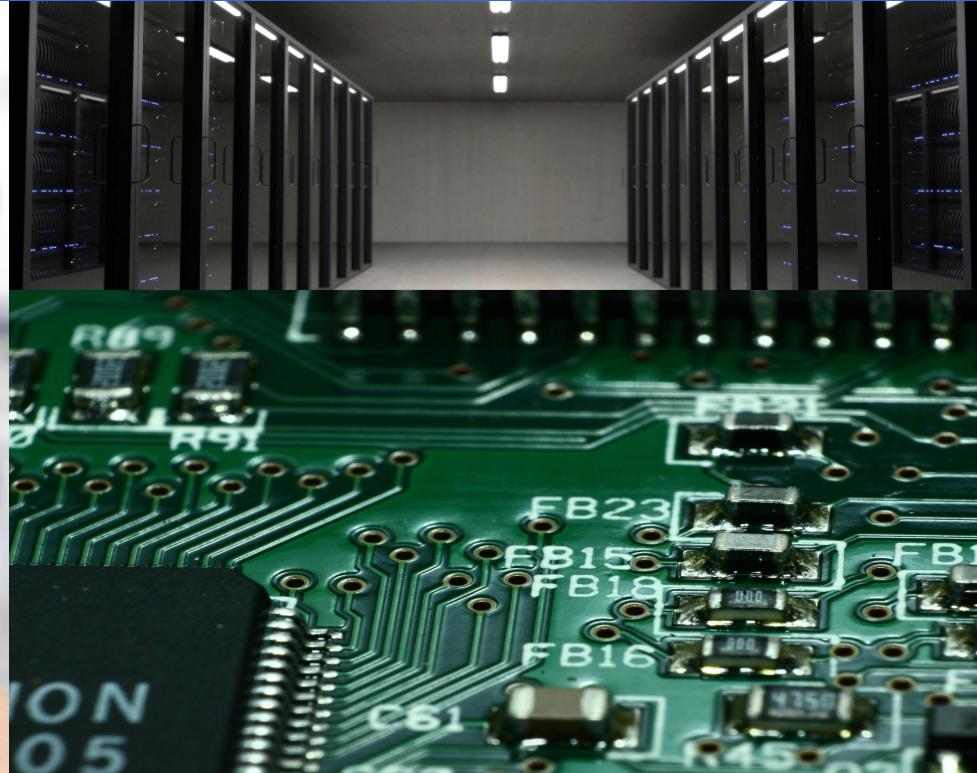
open**FORCE**

Product Information Management  
Integration

Hauptsächlich PHP Tech Stack



# Zielplattformen



# Developer Experience

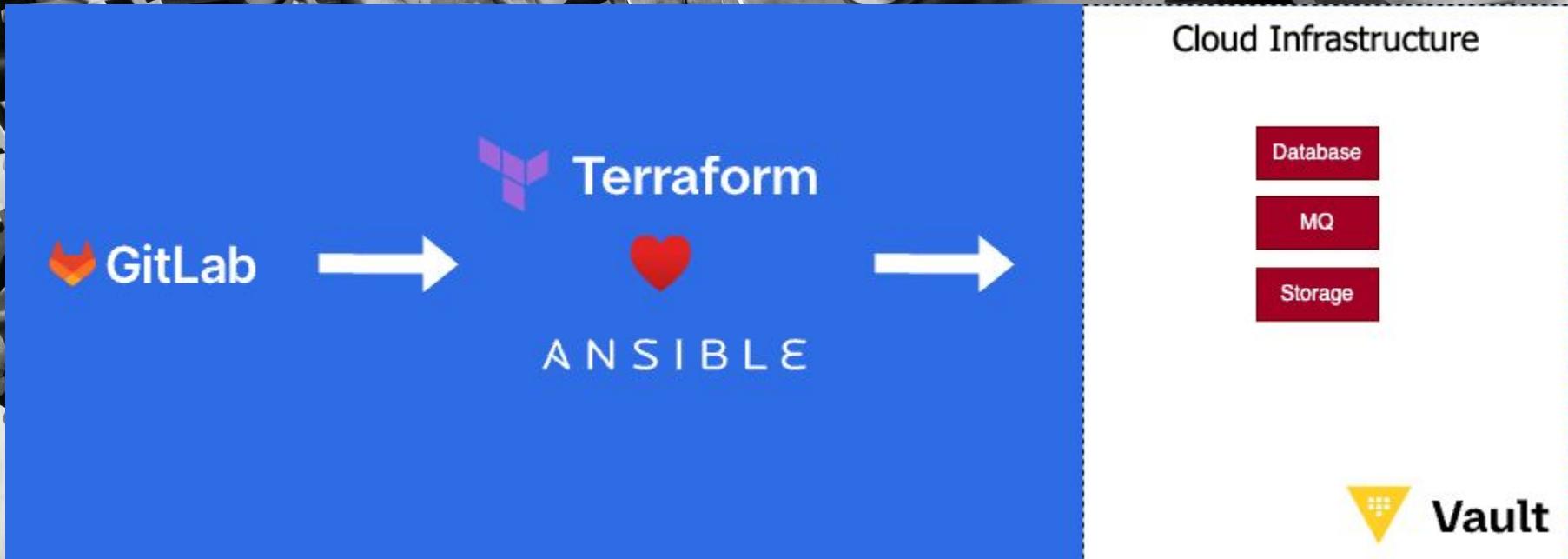


# Ops Strategie

- Virtualisierung
- keine eigene Hardware
- Containerisierung
- Orchestration



# Infrastructure as Code

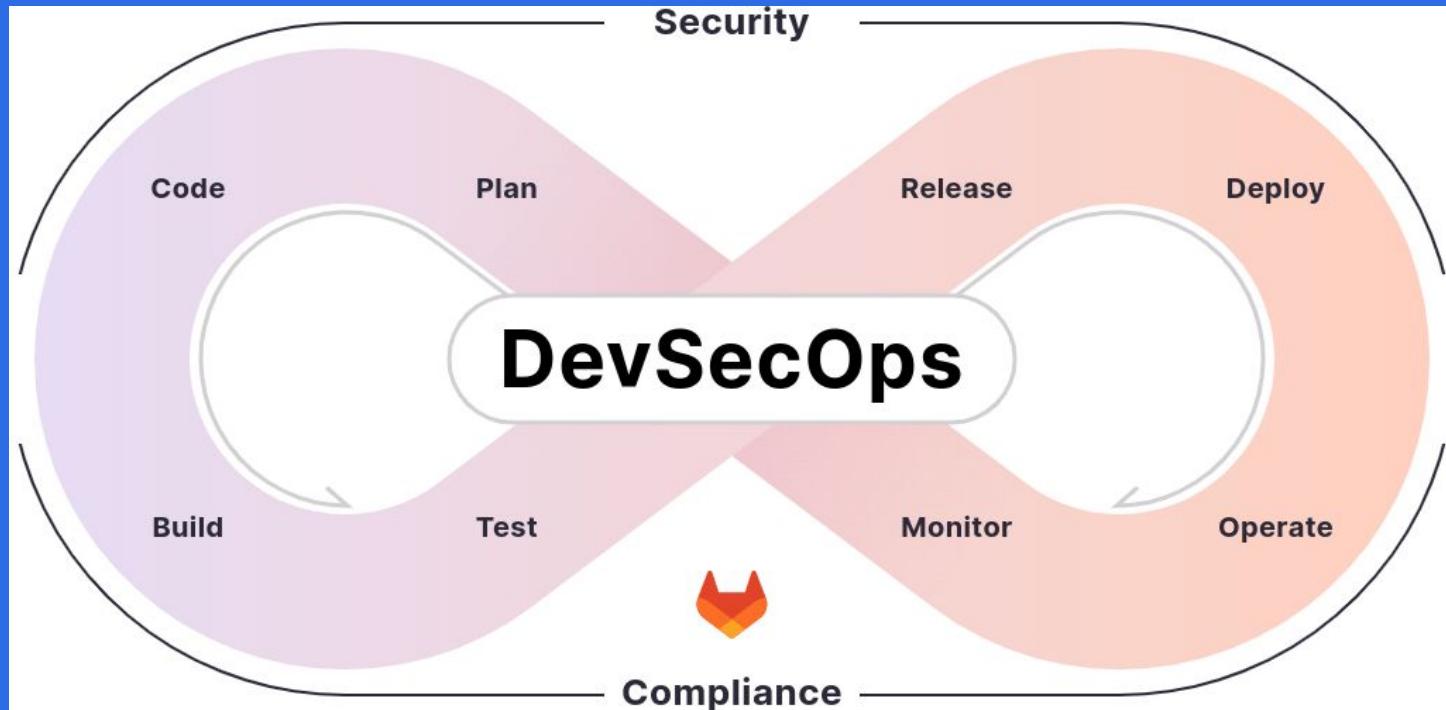


# Dev Strategie

- Local Containers
- Test Containers
- Shift Left Konzept
- Twelve-Factor Apps
- Environment as a Service



# Show don't tell - agile



# CI/CD vollautomatisiert



Cloud Infrastructure

Application

Database

MQ

Storage



# Environment Setup Speed

- 2002 - Tage bis Wochen (Hardwarebestellung)
- 2010 - Tag(e) für Cloud VMs (tw. manuell)
- 2020 - Minuten bis die Cloud VMs (automatisch)
- 2024 - Millisekunden (FaaS)



# openFORCE

goes

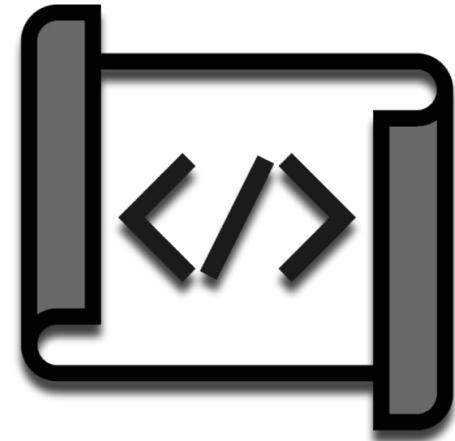


# Challenges:

- “How to” Kubernetes
- Standardisierung des Deployment Prozesses
- Cloud native-mindset

# Dinge die wir leben:

- CI/CD driven
- Infrastructure as Code
- Git OPS



# Ergebnis der Prinzipien

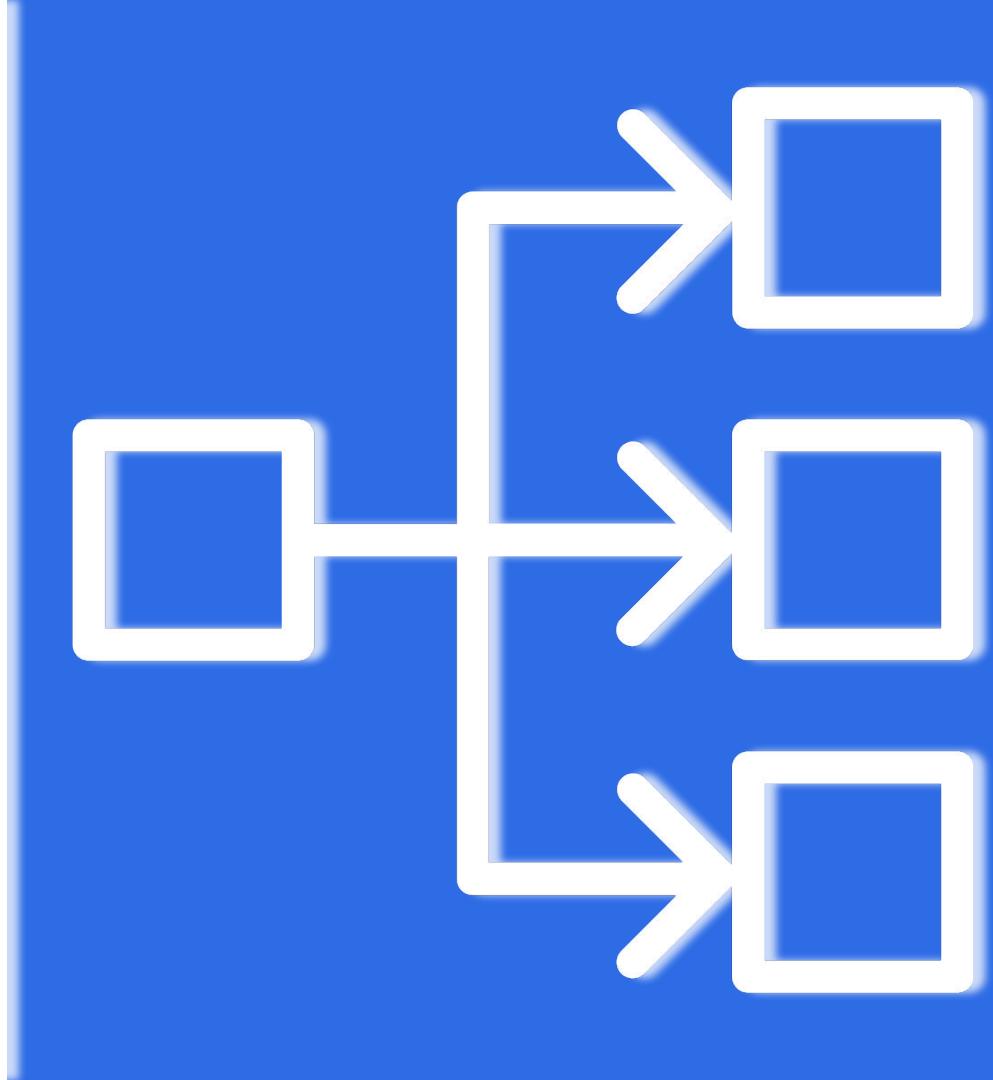
- “Base Repo“ wird geforkt
- Nachziehen von Änderungen durch Update des forks
- Deklarative Konfiguration des Clusters



Was passiert wenn  
wir einen Cluster  
spawnen?



# Load balancing



# Virtuelle Maschinen



Netzwerk  
Kommunikation  
K8S



# Setup: S3 - Backups



# Basiskonfiguration



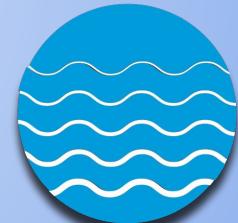
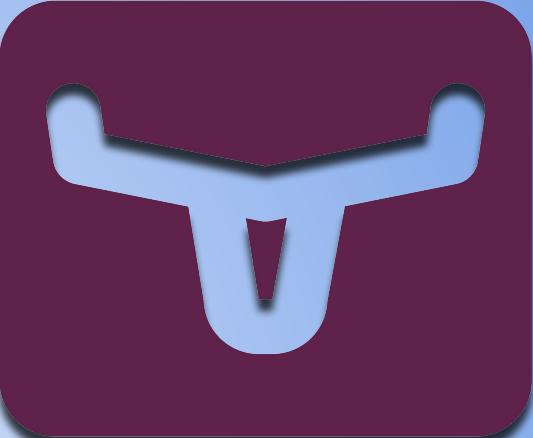
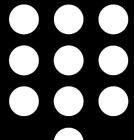
K8S

# Basis Konfiguration K8S:

- Anbindung Vault (Kubernetes Auth Methode)
- External Secrets Operator
- Certbot
- Ingress Nginx
- Longhorn
- Secrets-reflector
- Velero



# NGINX

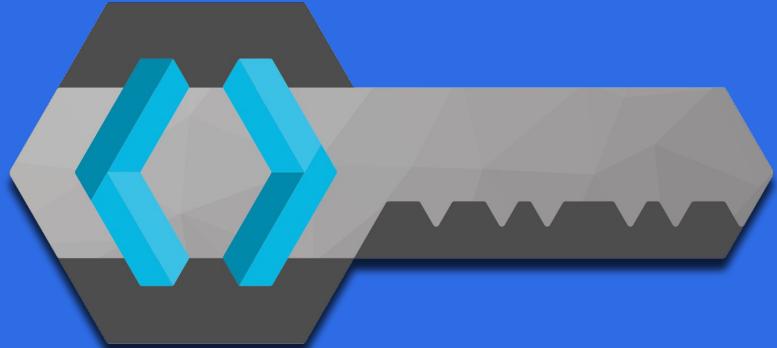


# VELERO

# Berechtigungsmanagement

# OIDC Login

- Kubernetes API
- Argo CD
- Monitoring



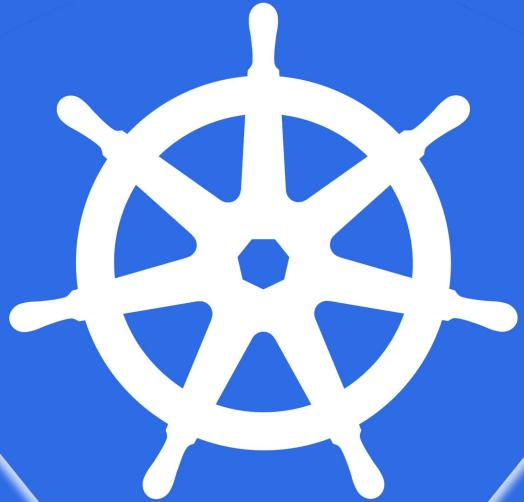
*Und in weiterer Folge alle Applikationen im Cluster*

# Konsequenzen unseres Handelns

# Resultate:

- Single source of truth: Terraform state
- so wenig wie möglich, so viel wie notwendig  
manuelle Konfiguration /  
Intervention des deployment Prozesses

# Erwartungen an K8S



# Erwartungen an K8S:

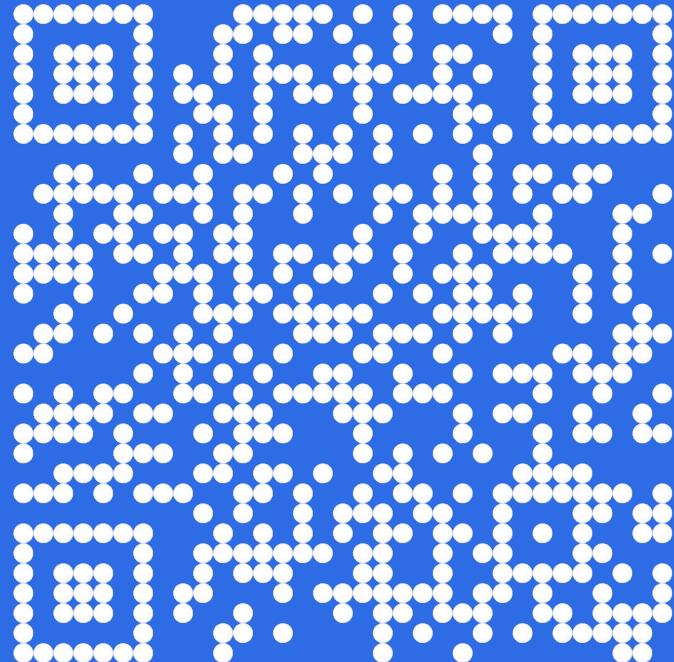
- Ausfallsicherheit (High Availability)
- besseres Lifecycle-Management von Applikationen:
  - Ausrollen von Applikationen
  - Updaten
  - Streamlining der involvierten Prozesse

Was wir daraus  
gelernt haben?

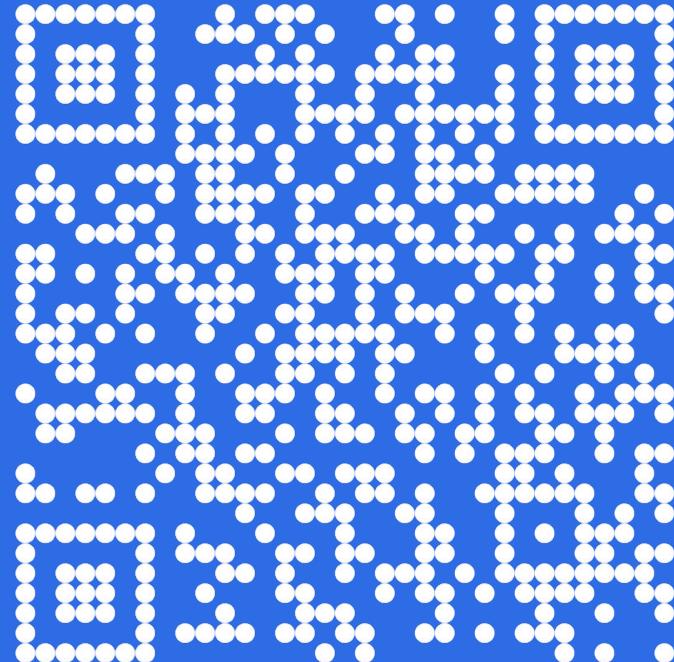
# Keyword: Standardisierung



# Demo



Gerhard Hipfinger



Cevin Freitag