

Christian Köberl
Johannes Grumböck

Digitalisierung des Automobilhandels durch Modularisierung

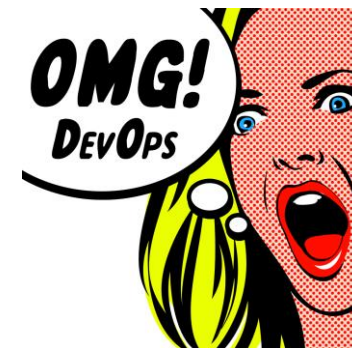




Christian Köberl
Software-Architekt
@derkoe



Johannes Grumböck
Infrastruktur-Architekt
@jgrumboe



Millionen User täglich

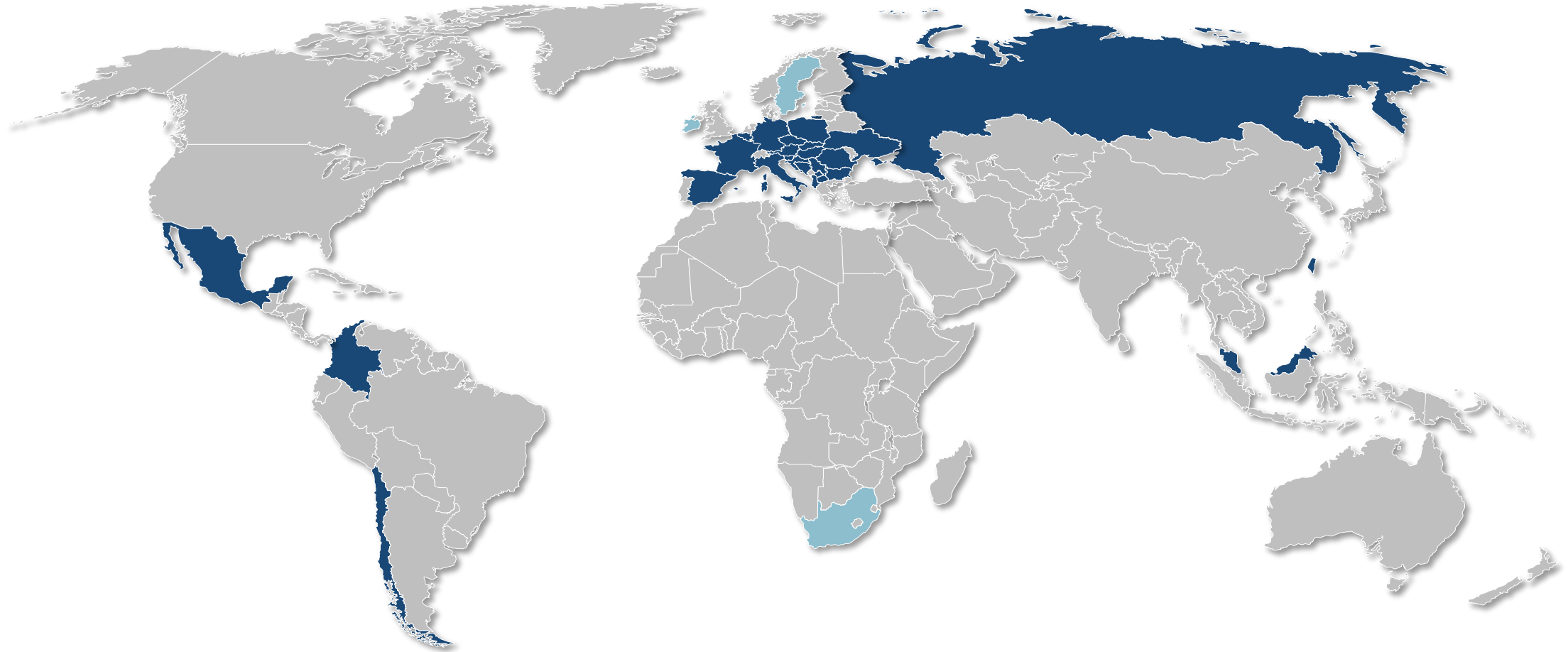
160 Lösungen

26 Länder weltweit

470 Digitalisierungs-Spezialisten

Porsche-Informatik-Systeme in 26 Ländern auf 3 Kontinenten

Laufende Projekte in drei weitere Ländern

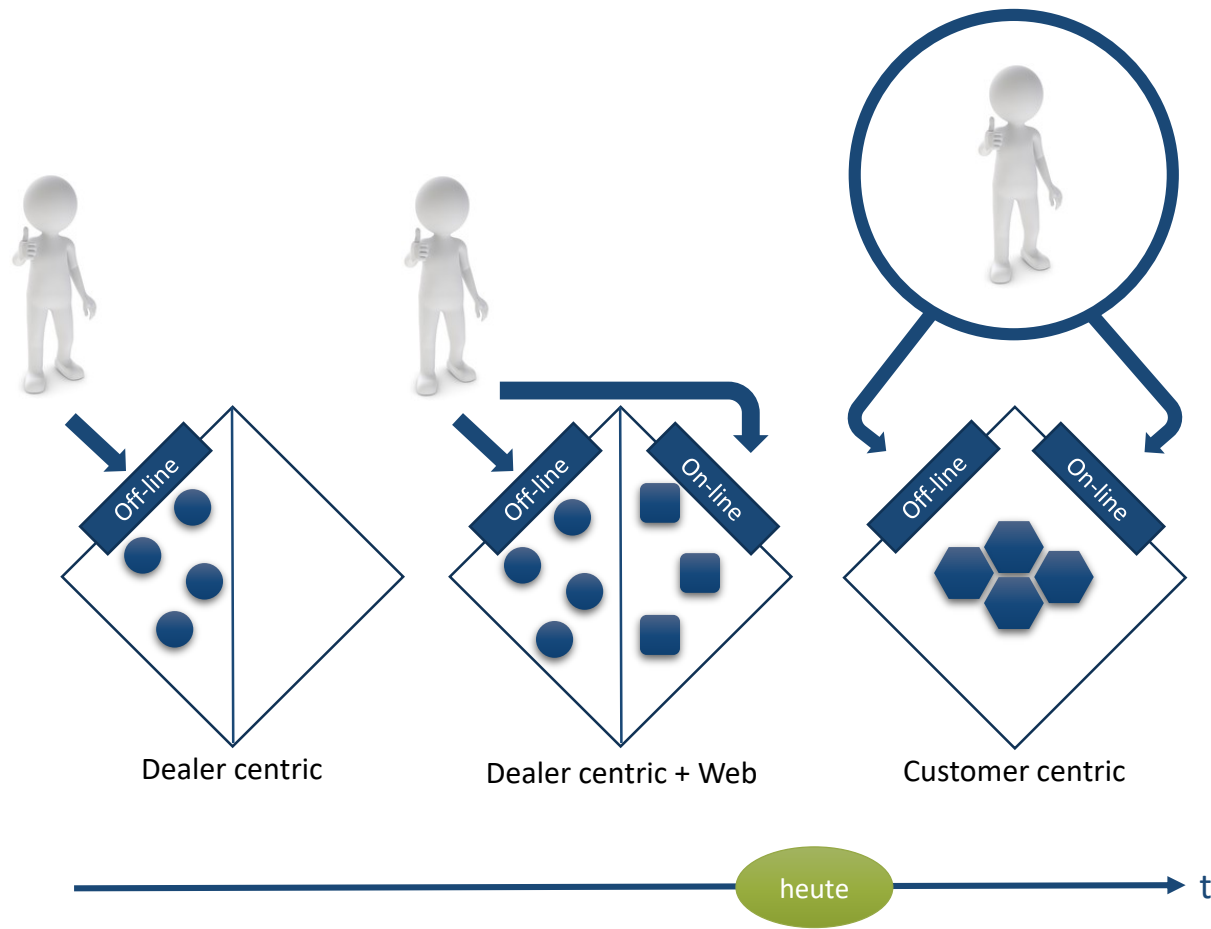


Porsche-Informatik-Systeme in
Produktiveinsatz; laufende Projekte

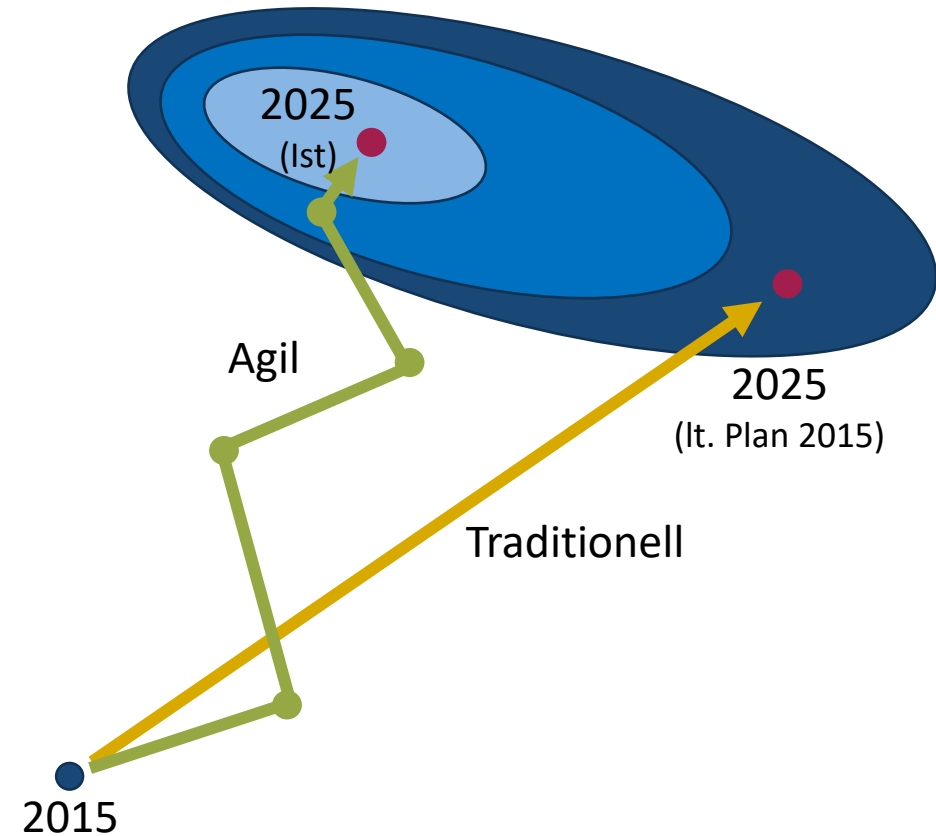
Laufende Projekte

"POI 4.0 - From good to great"

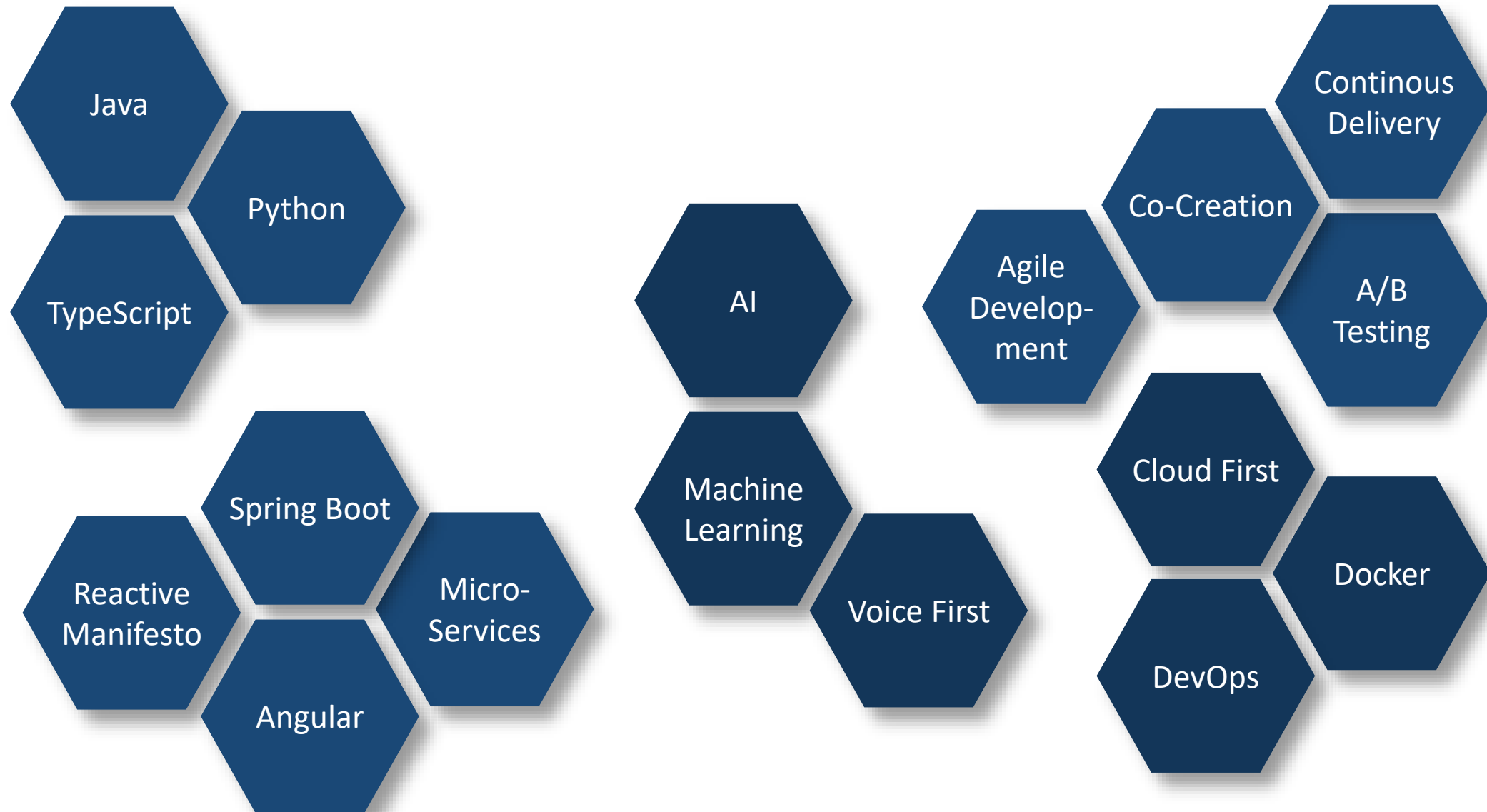
- Von *Dealer centric* zu *Customer centric*



- Agile Strategie



Unsere Entwicklungsmethoden und Technologien





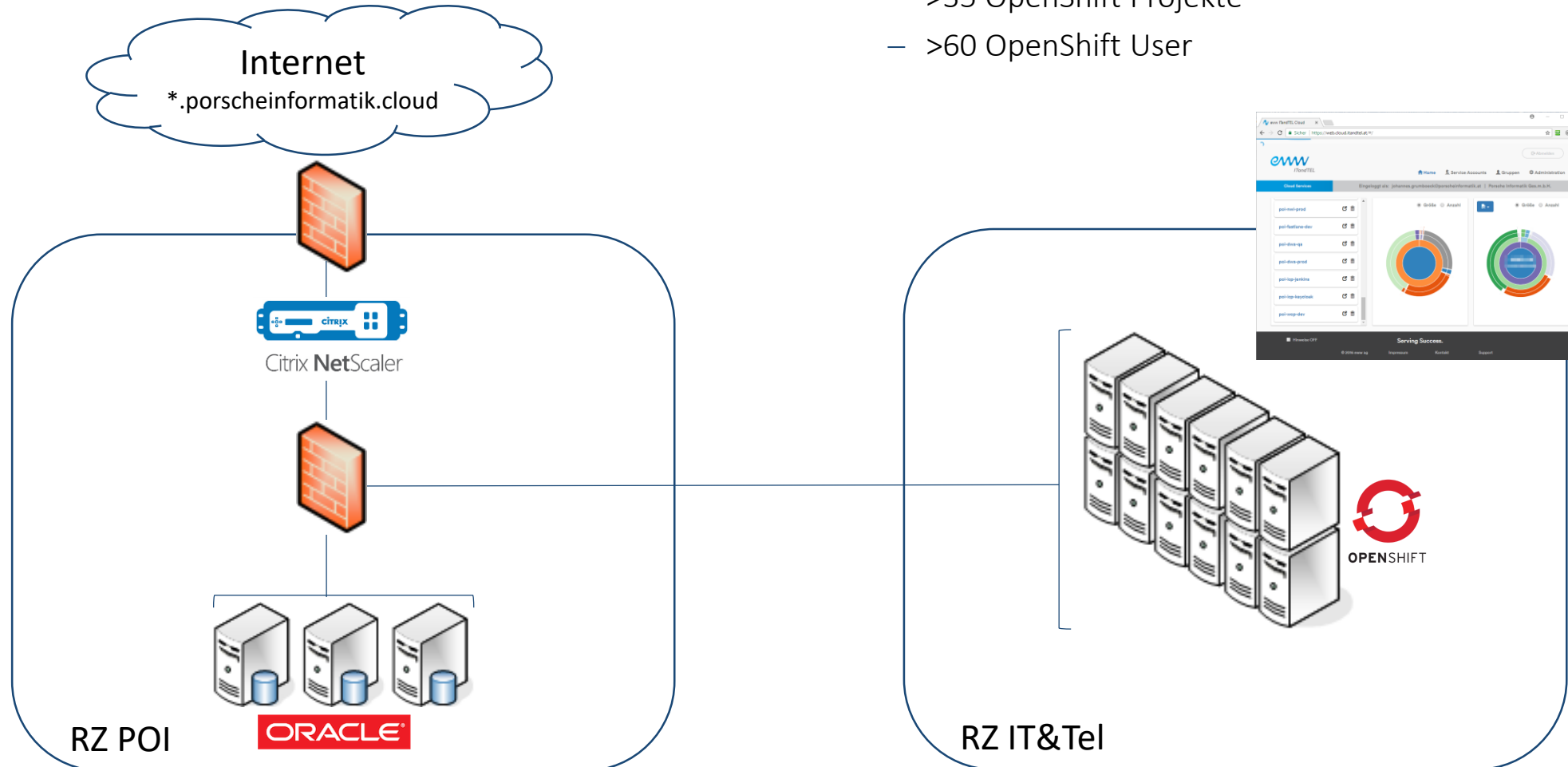
kubernetes

Production-Grade Container Orchestration



Red Hat® OpenShift is a container application platform that brings Docker and Kubernetes to the enterprise.

Private Cloud mit IT&Tel



■ Kennzahlen:

- 24 Nodes (48 vCPU, 384GB RAM, geteilt in Prod/Non-Prod)
- >35 OpenShift Projekte
- >60 OpenShift User

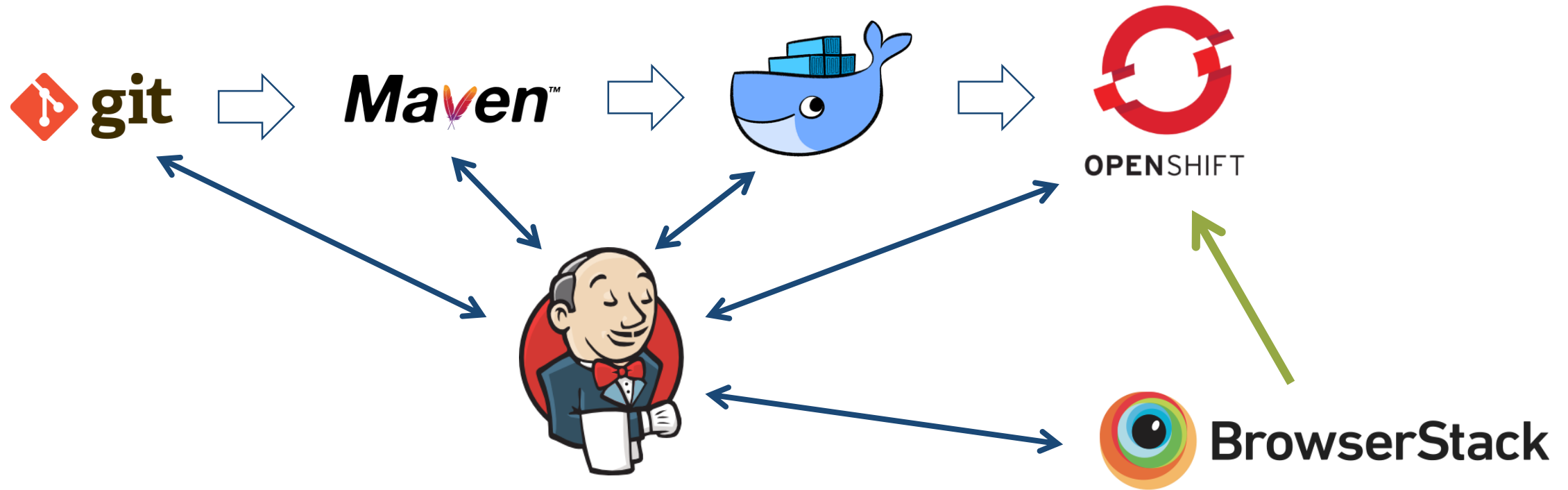
OpenShift – Use Case 1

Neue Applikationen

- Einfaches Zero-Downtime Deployment - Continuous Delivery
- Neue Technologien können leicht betrieben werden
 - NodeJS, Ruby, Python
 - Andere Datenbanken (Postgres, MongoDB, Cassandra, ...)
 - Caches (Redis, ...)
- Entwickler können komplett eigenständig Applikationen / Middleware aufsetzen ohne auf die darunterliegende Infrastruktur eingehen zu müssen

OpenShift – Use Case 2

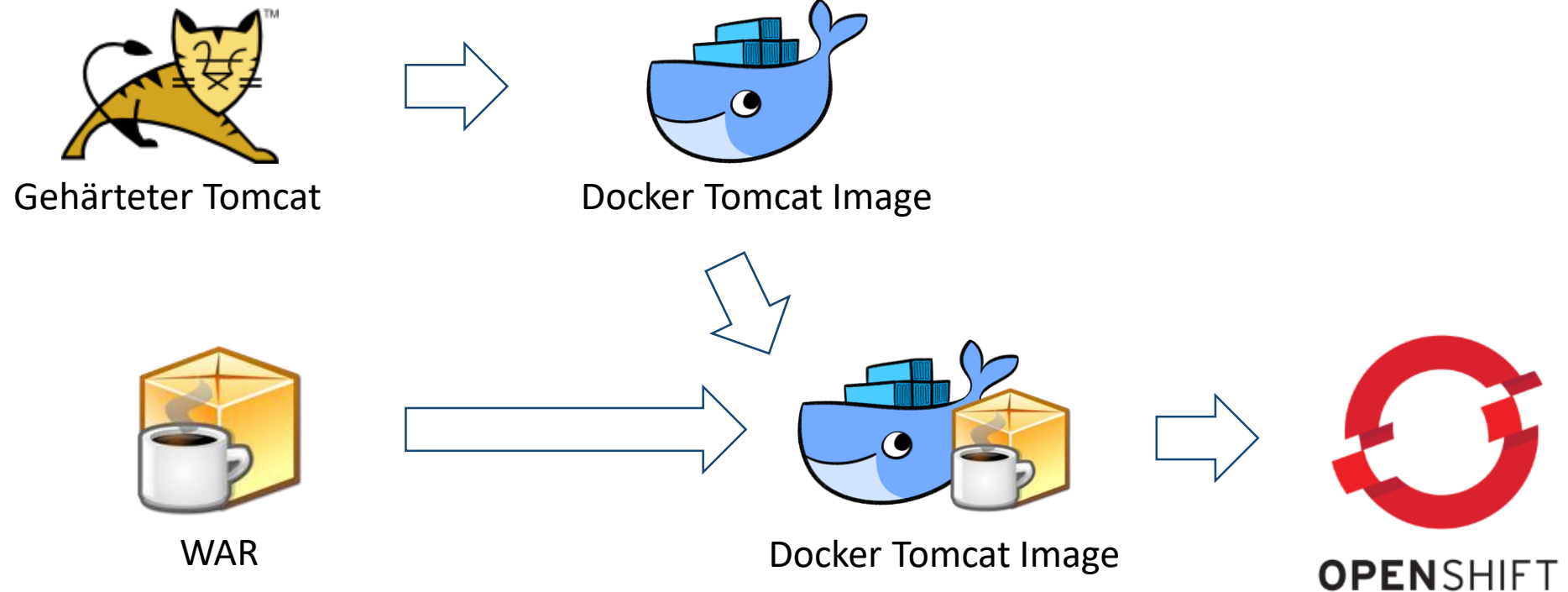
Testumgebungen / Automatisierte Tests



- Plan: Testumgebung je Change/Pull-Request

OpenShift – Use Case 3

Legacy Tomcat Applikationen



OpenShift – Use Case 4

Neue interne Tools in Minuten online

- Spring Boot Applikation oder Docker Images
- Schnell zum Ergebnis kommen

- Beispiele
 - Jenkins für Team deployen
 - Wekan für Social Media Team
 - Von Praktikant entwickelte Reporting Applikation für SonarQube
 - Keycloak für Cloud Team

Probleme / Lessons Learned

- Java in Docker Containern
 - Memory (Container-Limit vs. Host-Limit)
 - CPU Anzahl (ForkJoin Pools, etc.)
- Viele Server, Frameworks, Plattformen nicht 100% für Container geeignet
- Standard-Images laufen nicht immer out-of-the-box
 - User non-root
 - Filesystem read-only per default
- Entwicklung muss sich um viel mehr kümmern
 - Updates/Patches (von Images), Backup/Restore von Daten, ...
 - Verantwortung verschiebt sich ("Shift left")
- Betrieb und Integration von OpenShift nicht immer trivial
 - Dedicated vs. Shared-Resources

Zusammenfassung

- Kubernetes (K8s) ist **DER** Standard für Container in Produktion
- K8s/OpenShift ermöglicht einfaches Continuous Delivery
- OpenShift ist K8s mit sicherer Standard-Konfiguration
- OpenShift bietet mit Builds/ImageStreams die Basis für PaaS
- Cloud-Technologien und DevOps helfen uns die Digitalisierung und Modularisierung voranzutreiben

Danke!
Q&A?

Christian Köberl, @derkoe
Johannes Grumböck, @jgrumboe
www.porscheinformatik.at

