



INFOSERVE
>eurodata-Gruppe

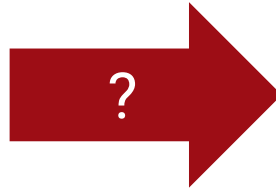
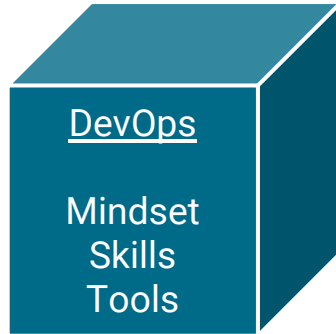
DevOps aus Sicht eines IT-Leiters

DevOpsSaar 31.07.2018

DR. PHILIPP WALTER | Leiter IT | INFOSERVE GmbH

Mein IT-Leiter-Problem

DevOps ist super! Und jetzt?

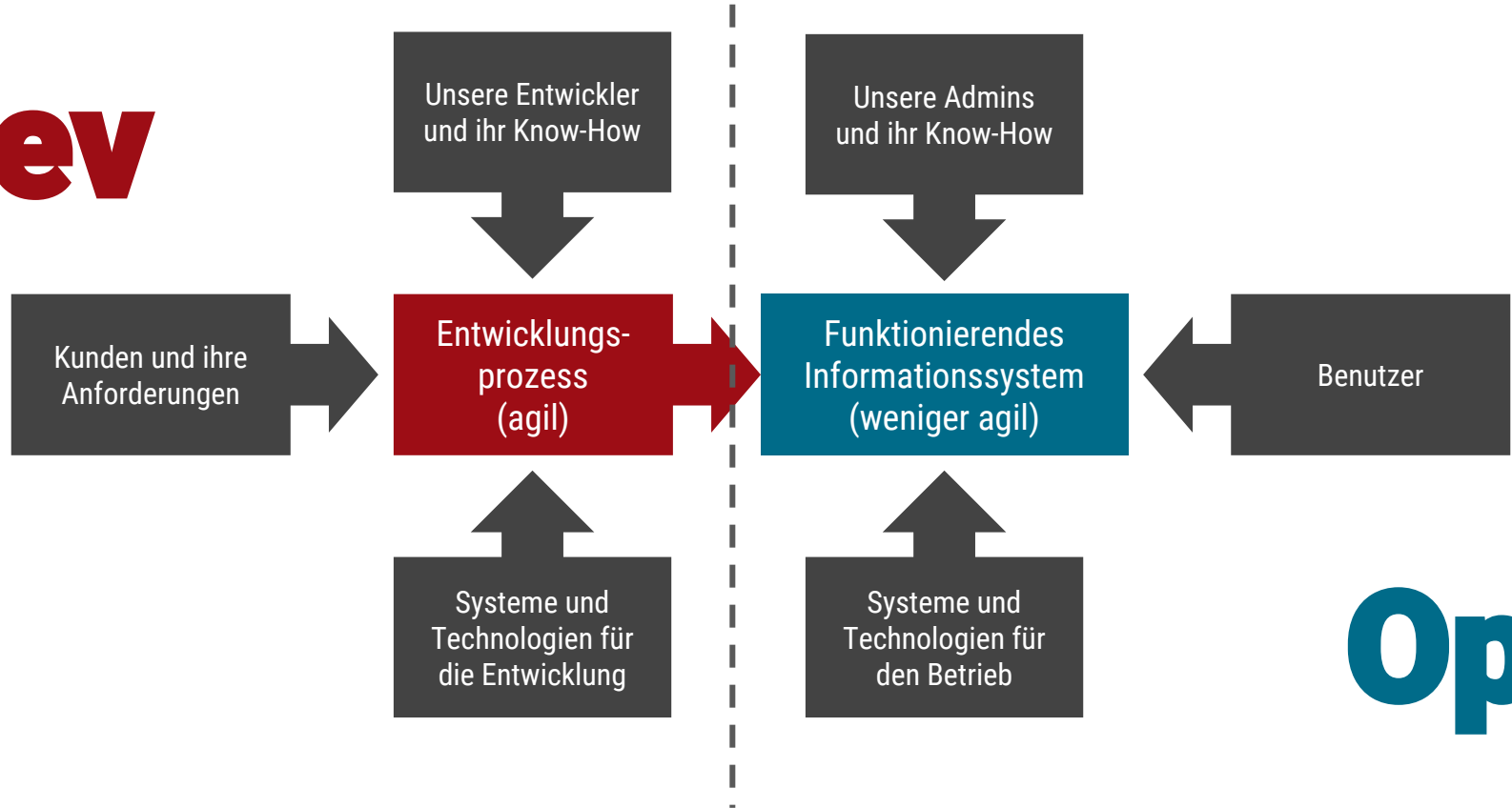


Kunden, Projekte, Mitarbeiter, Prozesse,
Infrastruktur, Sicherheit, ...

Softwareprojekte

herkömmlicher Bauart

Dev



Softwareprojekte

bis vor ~8 Jahren

Akquise

Entwicklung

Deployment

Abnahme

Support

Anforderungen klären:

Was Kunde sagt/meint
!= was Kunde braucht

Angebot erstellen:

Was tun wir zu welchem
Preis? Möglichst zum
Festpreis?

Programmieren:

“Es steht alles im
Angebot, muss man
doch nur runtercoden!”

“Alles lief nach Plan.
Aber der Plan war
scheiße.”

Produktivsysteme aufsetzen:

“Komisch, auf meinem
Entwickler-PC lief es
genau so!”

Stunde der Wahrheit:

“Als wir X gesagt
haben, meinten wir Y!
Das hätte doch klar
sein müssen!”

Upgrades:

Deployments nur in
Randzeiten wegen
Downtimes, Entwickler
muss dabei sein, ...

Kein Plan
überlebt die
erste Feind-
berührung!

Probleme
werden erst
spät erkannt.

Deployment
und Fixes sind
aufwändig und
riskant.



Change Management?

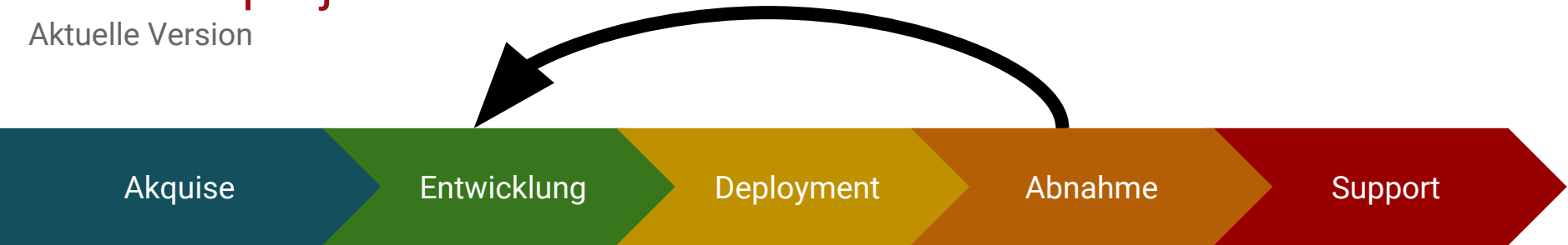
Klassische vs. agile Entwicklung

Oder: wie sag ich es meinem Kunden?

- Klassisches Wasserfallmodell: Lastenheft, Pflichtenheft, Angebot
 - Anforderungen müssen vollständig bekannt und formuliert sein
 - Fachliche und technische Abstimmung sowie die gesamte Entwicklung müssen vollständig vorweg genommen werden
 - Ergebnisse werden erst spät sichtbar
 - Nachträgliche Verfeinerungen oder Ergänzungen von Anforderungen sind im Prozess nicht vorgesehen
- Agile Entwicklung
 - Anforderungsklärun und Aufwandsschätzung liefern einen Projektumfang
 - Entwicklung erfolgt in "Sprints" mit ca. 3 Wochen Länge
 - Es gibt immer einen lauffähigen Zwischenstand in einer Staging-Umgebung, der regelmäßig aktualisiert wird
 - Kurze Feedbackzyklen stellen sicher, dass das Entwicklungsergebnis allen Anforderungen (insbesondere später erst erkannten) genügt

Softwareprojekte

Aktuelle Version



Anforderungen klären:

Was Kunde sagt/meint
!= was Kunde braucht

Angebot erstellen:

Realistischer
Budgetrahmen mit
Spielraum für neue und
geänderte Anforderungen

Programmieren:

Erst minimale Basis,
dann inkrementelle
Erweiterungen.

Proof of Concepts:

Technische
Möglichkeiten werden
in Code statt auf
Papier evaluiert.

Produktivsysteme aufsetzen:

“Komisch, auf meinem
Entwickler-PC lief es
genau so!”

Stunde der Wahrheit:

in etwa wöchentlich,
manchmal öfter

Budget monitoren:

Kunden tendieren zu
“wünsch dir was”,
daher sind sinnvolle
Vorschläge immer
wichtig

Upgrades:

Deployments nur in
Randzeiten wegen
Downtimes, Entwickler
muss dabei sein, ...

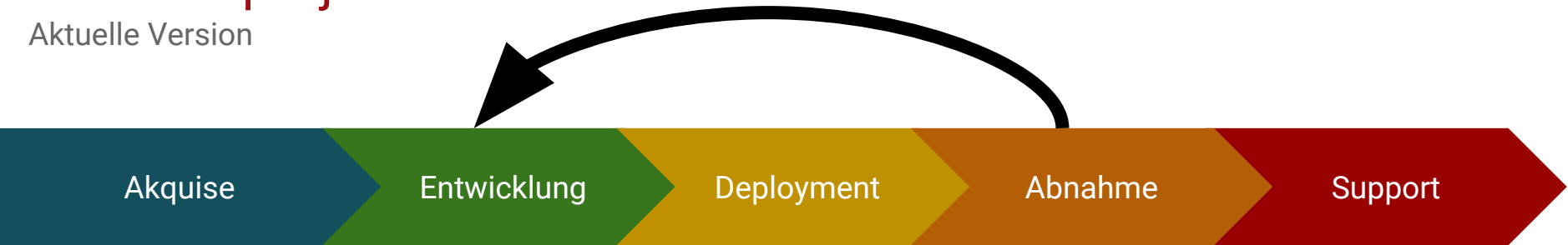
NEVER GIVE UP



ON YOUR DREAMS

Softwareprojekte

Aktuelle Version



Anforderungen klären:

Was Kunde sagt/meint
!= was Kunde braucht

Angebot erstellen:

Realistischer
Budgetrahmen mit
Spielraum für neue und
geänderte Anforderungen

Programmieren:

Erst minimale Basis,
dann inkrementelle
Erweiterungen.

Proof of Concepts:

Technische
Möglichkeiten werden
in Code statt auf
Papier evaluiert.

Erstes Deployment:

Admin setzt
K8s-Cluster auf,
Entwickler Helm-Chart.
Dann nur noch
`helm install`

Ab dann:

`mvn docker`
`helm upgrade`

Stunde der Wahrheit:

in etwa wöchentlich,
manchmal öfter

Budget monitoren:

Kunden tendieren zu
"wünsch dir was",
daher sind sinnvolle
Vorschläge immer
wichtig

Jederzeit:

`mvn docker`
`helm upgrade`

- DevOps = Ausdehnung agiler Methoden von Entwicklung auf Betrieb
- Google-mäßige Skalierbarkeit ist nur Beifang, wichtiger für uns sind:
 - Cloud-Technologien in der Entwicklung, z.B. Session Storage und SSO
 - Automatisierung des Deployments durch Entwickler
 - Verbessertes Monitoring und Selbstheilung
- Nächster Schritt: Kunden vom DevOps-Mehrwert überzeugen
 - wie damals bei der Einführung agiler Entwicklung, nur mit mehr Vertriebssupport
 - schwierig, weil Kunden häufig gar nicht das Problem verstehen
- Danach: Umsetzung starten
 - Technologieauswahl ist getroffen (K8s, Helm, usw.)
 - Erste Deployments sind bereits als PoCs gelaufen
 - Entwickler und Admins müssen Routine entwickeln (inkl. ein paar unvermeidbarer Krisen)

Danke für Ihre Aufmerksamkeit! Fragen?