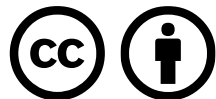


DevOps

Überblick

Prof. Dr.-Ing. Andreas Heil



Dieses Werk ist lizenziert unter einer Creative Commons
Namensnennung 4.0 International Lizenz.

Ziele und Kompetenzen

Grundlegende Probleme der herkömmlichen Entwicklungs- und Betriebsorganisation **kennen lernen**,

verstehen was mit DevOps erreicht werden soll

und

einige Möglichkeiten **kennen lernen** wie diese Ziele erreicht werden können.

Projekt Phoenix



Projekt Phoenix

Gene Kim, Kevin Behr & George Spafford

Deutsche Ausgabe

O'Reilly, 2015

ISBN 978-3958751750

„Projekt Phoenix“ - Ausgangslage

„Bill Palmer ist IT-Manager bei Parts Unlimited. An einem Dienstag morgen erhält er auf der Fahrt zur Arbeit einen Anruf seines CEO. Die neue IT-Initiative der Firma mit dem Codenamen Projekt Phoenix ist **entscheidend für die Zukunft** von Parts Unlimited, aber **das Projekt hat Budget und Zeitplan massiv überzogen**. Der CEO will, dass Bill direkt an ihn berichtet und das ganze **Chaos in neunzig Tagen aufräumt**, denn sonst wird Bills gesamte Abteilung outgesourct.“

„Projekt Phoenix“ – Die ersten Fails (1)

Fail #1

Die Gehaltssoftware spuckt bei einem signifikanten Teil der Mitarbeiter nur Hieroglyphen aus, anstatt das tatsächliche Gehalt zu berechnen.

Gehälter drohen, nicht ausgezahlt zu werden. Der Betriebsrat würde Alarm schlagen und die Presse negative Schlagzeilen zu Parts Unlimited schreiben.

„Projekt Phoenix“ – Die ersten Fails (2)

Fail #2

Weil sich unterschiedliche Teams nicht abgestimmt haben, dauert der Rollout einer Kassensoftware viel zu lange. Die Datenbank-Migration dauert nicht nur Stunden, sondern Tage.

Doof, dass die Kassen am nächsten Tag wieder öffnen und ohne Software nicht arbeiten können. Filialen nehmen die Kreditkartendaten der Kunden entgegen und schreiben sie auf, was aber ein Sicherheitsverstoß ist...

Zitate aus „Projekt Phoenix“ (1)

Über die Zusammenarbeit von Development und Operations

„Es ist erstaunlich, wie die Übergaben zwischen Entwicklung und IT Operations jedes Mal in die Hose gehen. Aber angesichts des dauerhaften Kleinkriegs zwischen den beiden Gruppen sollte ich nicht überrascht sein.“

Zitate aus „Projekt Phoenix“ (2)

Über Brent, dem Alleskönner, von dem alles abhängt

*„Ich sage ja nicht, dass Brent das absichtlich macht, aber ich frage mich, ob Brent sein Wissen als eine Art Superheldenkraft sieht. Vielleicht will ein Teil von ihm das gar nicht aufgeben. **Er hat damit eine Stellung inne, von der ihn niemand vertreiben kann.**“* „Vielleicht, vielleicht auch nicht“, sage ich. „Aber ich werde euch erzählen, was ich sicher weiß: **Jedes Mal, wenn wir Brent etwas fixen lassen, das keiner von uns wiederholen kann, wird Brent ein bisschen klüger und das gesamte System ein bisschen dümmer. Wir müssen dem ein Ende bereiten.**“

Zitate aus „Projekt Phoenix“ (3)

Über erfolgreiche Teams

*„Ein großartiges Team bedeutet nicht, dass es nur aus den klügsten Köpfen besteht. Ein Team wird dann gut sein, **wenn sich alle vertrauen können**. Es entsteht viel Power, wenn diese Magie zwischen den Leuten existiert.“*

Zitate aus „Projekt Phoenix“ (4)

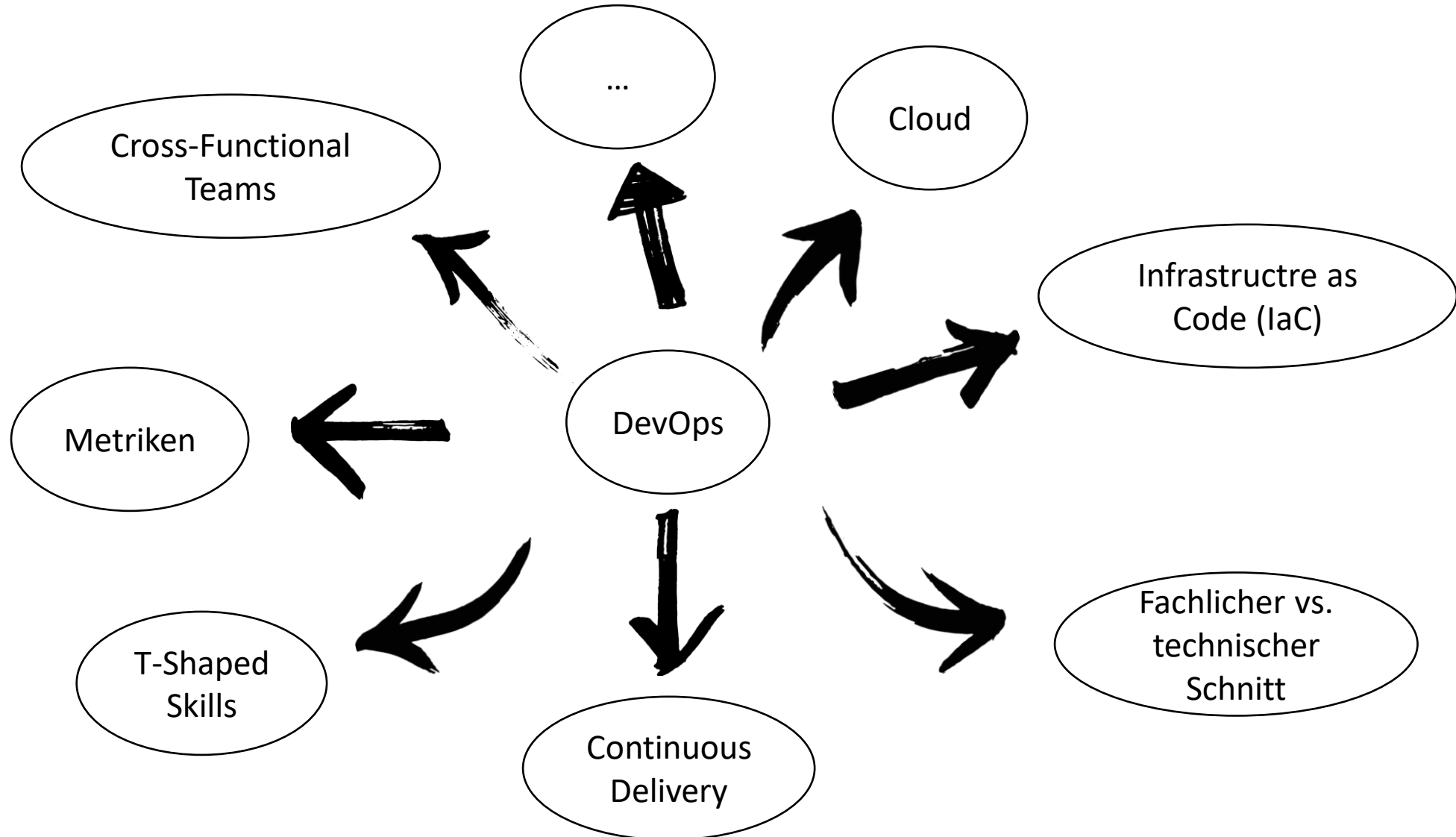
**Über die Zeit, in der ein neues Produkt auf dem Markt sein muss
(Time-to-Market)**

*„**Produkte müssen in sechs Monaten auf dem Markt sein.** Höchstens in neun. Ansonsten schnappt sich eine chinesische Firma unsere Idee, bringt sie in die Läden unserer Konkurrenz und bekommt damit einen großen Marktanteil. In diesen Zeiten starker Konkurrenz muss man schnell im Markt sein und auch die Fehler schnell machen. Wir können uns keine mehrjährigen Entwicklungszyklen mehr leisten und erst am Ende erkennen, ob das Produkt gut ist oder beim Kunden durchfällt. Es müssen kurze und schnelle Zyklen sein, in denen das Feedback aus dem Markt fortlaufend einfließt.“*

DevOps: Definition

- Kunstwort aus engl. »Development« = Entwicklung und »Operations« = Betrieb
- DevOps klingt wesentlich cooler als »EntBet«
- DevOps beschreibt eine Form der Zusammenarbeit, bei der **Entwicklung und Betrieb** einer Software **als untrennbare Einheit** angesehen werden
- Mittlerweile auch Abwandlungen davon, z. B. **DevSecOps**, wo der Security-Aspekt explizit erwähnt wird (Berücksichtigung von Sicherheitslücken in einer Software und Umgebung) → MSEM

Was gehört alles dazu?



Wie wurde früher^{*)} Software entwickelt?



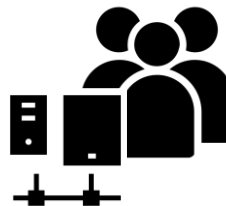
Netzwerkteam



Produktteam



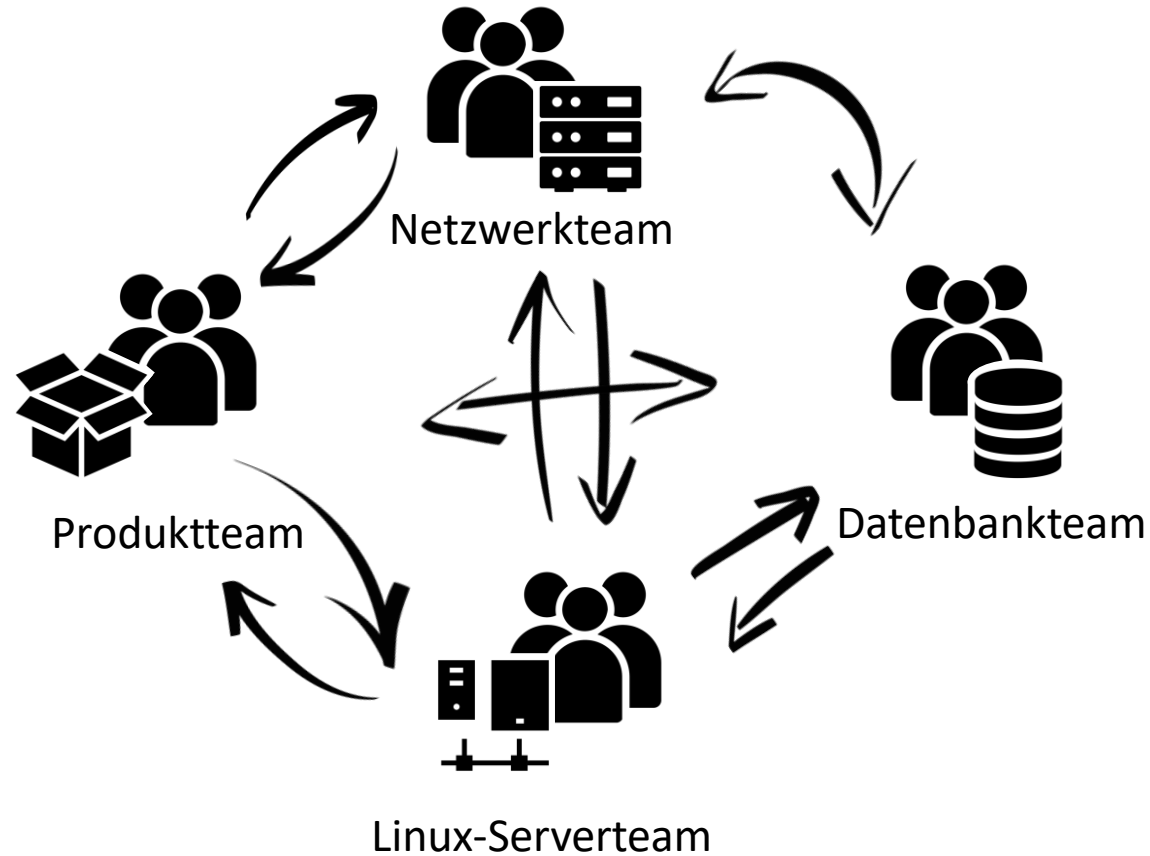
Datenbankteam



Linux-Serverteam

^{*)} häufig ist dies auch heute noch der Fall

Kommunikationswege^{*)}



^{*)} Sie haben sicherlich gedacht, dass das Sie das Zeug aus der Vorlesung nie mehr benötigen...

Der ewige Interessenskonflikt

Siehe Screencasts zum Thema Verschwendung

- Es werden dringliche Zuarbeiten von anderen Teams oder Abteilungen benötigt
- Es werden Tickets erzeugt
- Wartezeiten entstehen

Wie könnte dieser Konflikt gelöst werden?

Kleiner Tipp aus SEKS: Warum war Canon in den 1970ern mit der Canon AE-1 gleich nochmal so erfolgreich?*)

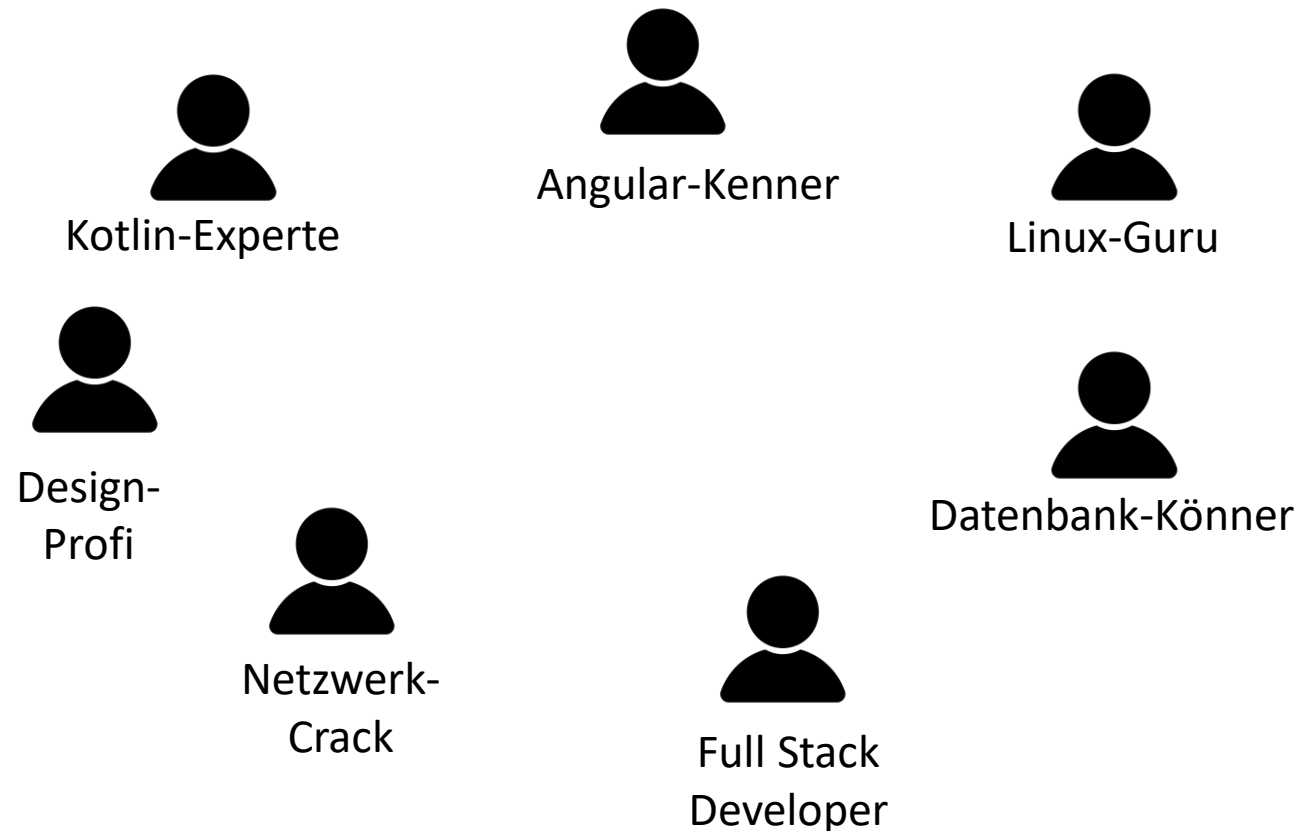
30 Sek. Bedenkzeit



*) Schon wieder was aus der alten Vorlesung...

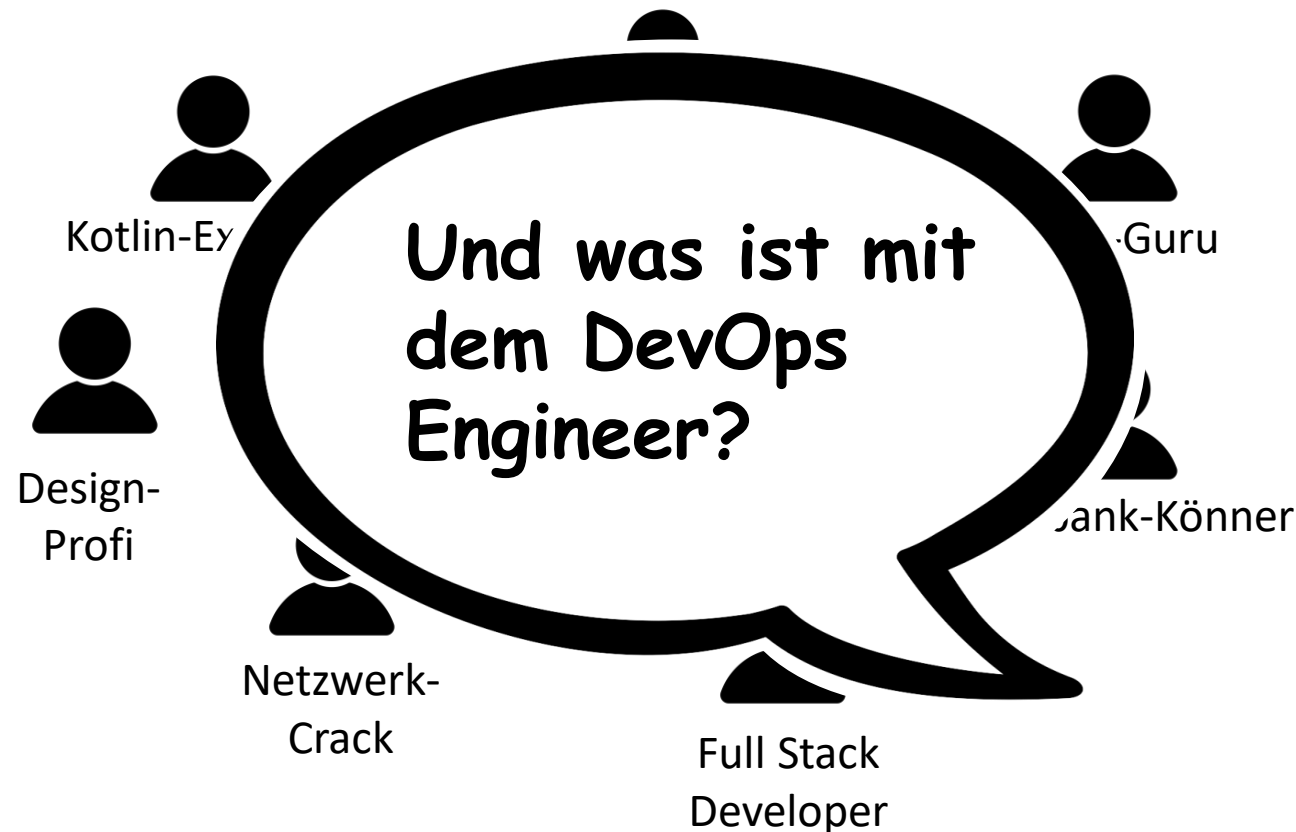
Lösung: Cross-funktionale Teams

Ein **gemeinsames** Produktteam



Lösung: Cross-funktionale Teams

Ein **gemeinsames** Produktteam



Lösung: Cro

Kotlin-









Design-
Profi

MONSTER

Jobs suchen Karriereberatung Erste Schritte

Filter ▾ Ihre letzten Suchen ▾ **devops engineer Jobs (445 Jobs gefunden)**

	DevOps Engineer (w/m/d) opta data Abrechnungs GmbH Essen, Nordrhein-Westfalen	Heute veröffentlicht
	DevOps Engineer (m/w/d) plentysystems AG Kassel, Hessen	Vor 3 Tagen veröffentlicht
	DevOps Engineer (m/w/d) Bundesdruckerei GmbH Berlin, Berlin	Vor 5 Tagen veröffentlicht
	DevOps Engineer (m/w/d) Bundesdruckerei GmbH Hamburg, Hamburg	Vor 6 Tagen veröffentlicht
	DevOps Engineer (m/w/d) Platform Services Bundesdruckerei GmbH Berlin, Berlin + weitere Standorte	Vor 6 Tagen veröffentlicht
	Pre-Sales / DevOps Engineer (m/f/d) Service Layers GmbH Späningen, Sachsen-Anhalt + weitere Standorte	Vor 6 Tagen veröffentlicht

Abrech

Arb
Ma

Karrier
Gesun

Die
hina
digi
1970
Grun
jahr
werd
ausg
für i

Du
Zuku
Esse

De

Quelle: www.monster.de (Abgerufen 22.02.2021)

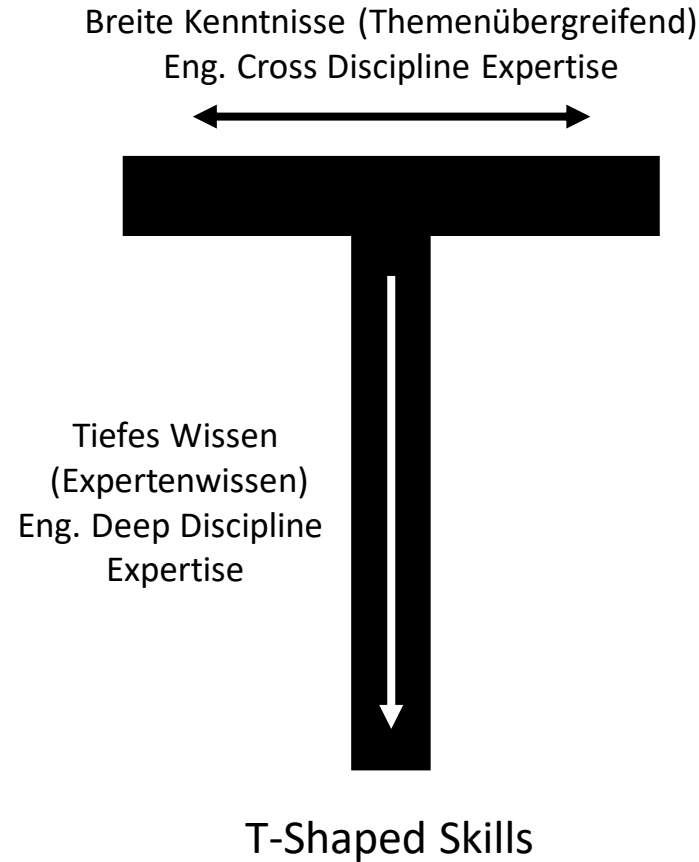
Was soll mit DevOps erreicht werden?

- Team ist eigenständig lieferfähig
- Alle benötigten Skills sind im Team vorhanden
- Team hat alle erforderlichen Befugnisse
- Umdenken in der Organisation: »Betroffene zu Beteiligten machen«
- Schnelle Lieferung von Software-Änderungen optimieren
- »Lead Time« soll möglichst gering gehalten werden, daraus resultierende kürzere Time-to-Market soll Marktvorteile sichern

Wie erreichen wir die vorherigen Ziele?

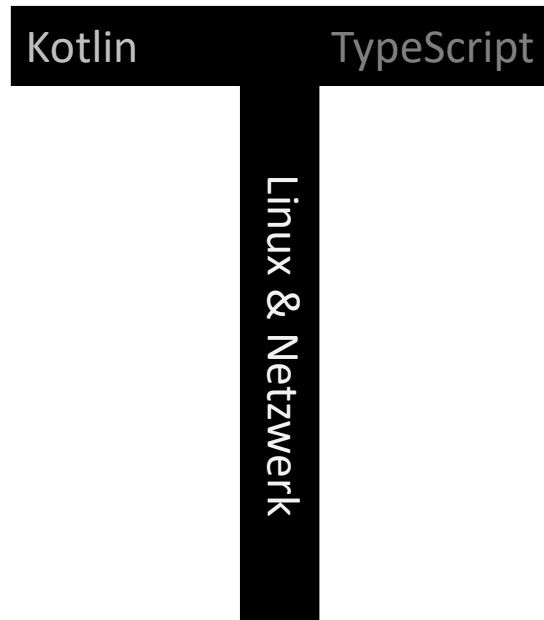
- **Automatisieren** von Prozessen
- **Minimieren** von **Übergaben**
- **Cross-funktionale** Teams
- **Agile** Vorgehensmodelle (Scrum oder Kanban)
- Gute **Feedback-** und **Fehler-Kultur** (vgl. Screencasts)
- **Fachlicher Schnitt** anstelle eines techn. Schnitts (vgl. SEKS) als Kernelement

Zielbild: T-Shaped Skills

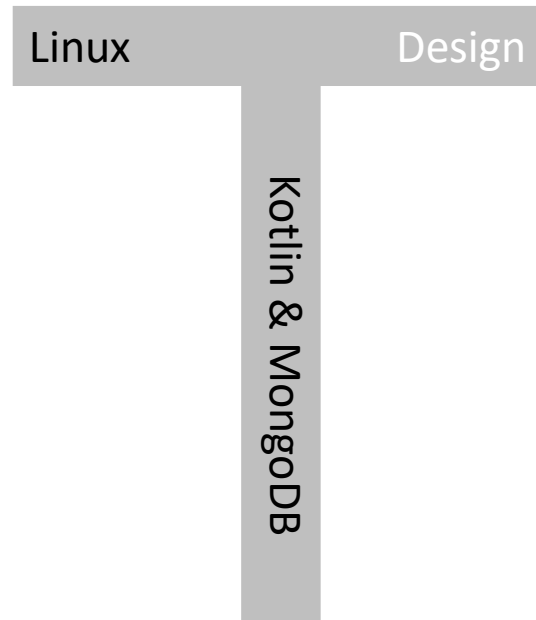


Erstmals 1991 von David Guest eingeführt, von Tim Brown CEO von IDEO Design Consultancy für Bewerberprofile eingesetzt.

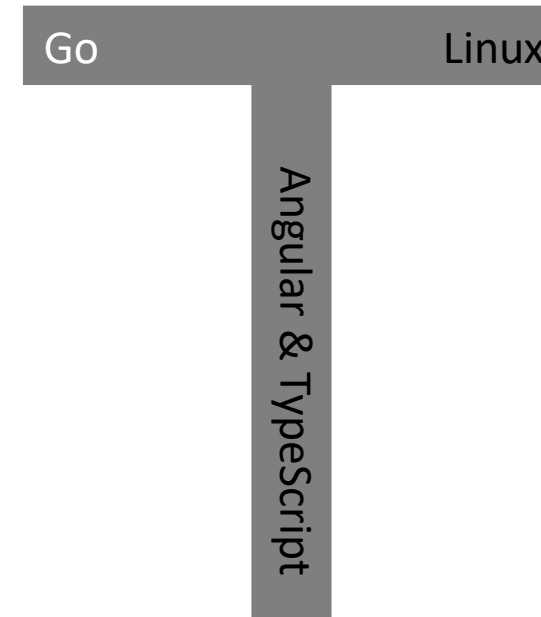
Beispiel: T-Shaped Skills



Roy Trenneman



Maurice Moss



Jen Barber

Wie können Teams selbstständig arbeiten

- Abhängigkeiten reduzieren
- Laufzeitumgebungen (Server, Container) selbstständig definieren
- Netzwerkimplementierungen anpassen (CDN)
- SelfServices für alles was nicht in der Verantwortung des Teams liegt
- Mehr als ein Kollege kennt sich in (Teil-)Bereichen der Software/Infrastruktur aus (vgl. T-Shaped Skills)
- Team »fühlt sich verantwortlich« für das gesamte Softwareprodukt (Entwicklung, Betrieb, und das 24x7)

Infrastructure as Code (IaC)

- Gesamte Infrastruktur wird in Konfigurationsdateien beschrieben
 - Bild-Umgebungen
 - Laufzeitumgebungen
 - Firewall-Regeln
 - Skalierungsparameter
 - Konfigurationen für Anwendungen
- Spezielle Auszeichnungssprache (JSON, YAML) in Form einer Domain Specific Language (DSL)
- Beschreibungsdateien werden versioniert und liegen wie Code in einem Repository (z.B. Git)

Metriken - Begriffserklärung

Metriken: Helfen Sachverhalte zu verstehen und zu bewerten.
(vgl. SEKS: Software-Metriken)

Monitoring: Visualisierung von Metriken

Alarmierung: Aktiver Hinweis beim Unter- oder Überschreiten von Schwellwerten (E-Mail, SMS, Push Notification, Ampelsystem)

Metriken

Metriken geben ein Einblick wie es unserer Applikation geht, indem kontinuierlich Messwerte geliefert werden, die visualisiert werden können. Außerdem werden wir alarmiert, wenn Messwerte bestimmte Schwellwerte unter- oder überschreiten.

Wo werden Metriken eingesetzt?

- Alle Cloud-Anbieter unterstützen unterschiedlichste Messtechnologien- und Plattformen
- Jedes Rechenzentrum nutzt Metriken zur Überwachung
- Auch ein NAS im privaten Einsatz nutzt Metriken (Durchsatz, verfügbarer Speicherplatz etc.)
- Es gibt technische als auch fachliche Metriken (vgl. SEKS Business- und Projektmetriken)
- Große Plattformen (Amazon, etsy etc.) werten Metriken automatisch aus und spielen Vorgängerversionen automatisiert ein, sollte in Update der Shop-Software Probleme verursachen
- Metriken können z.B. für A/B-Testing genutzt werden

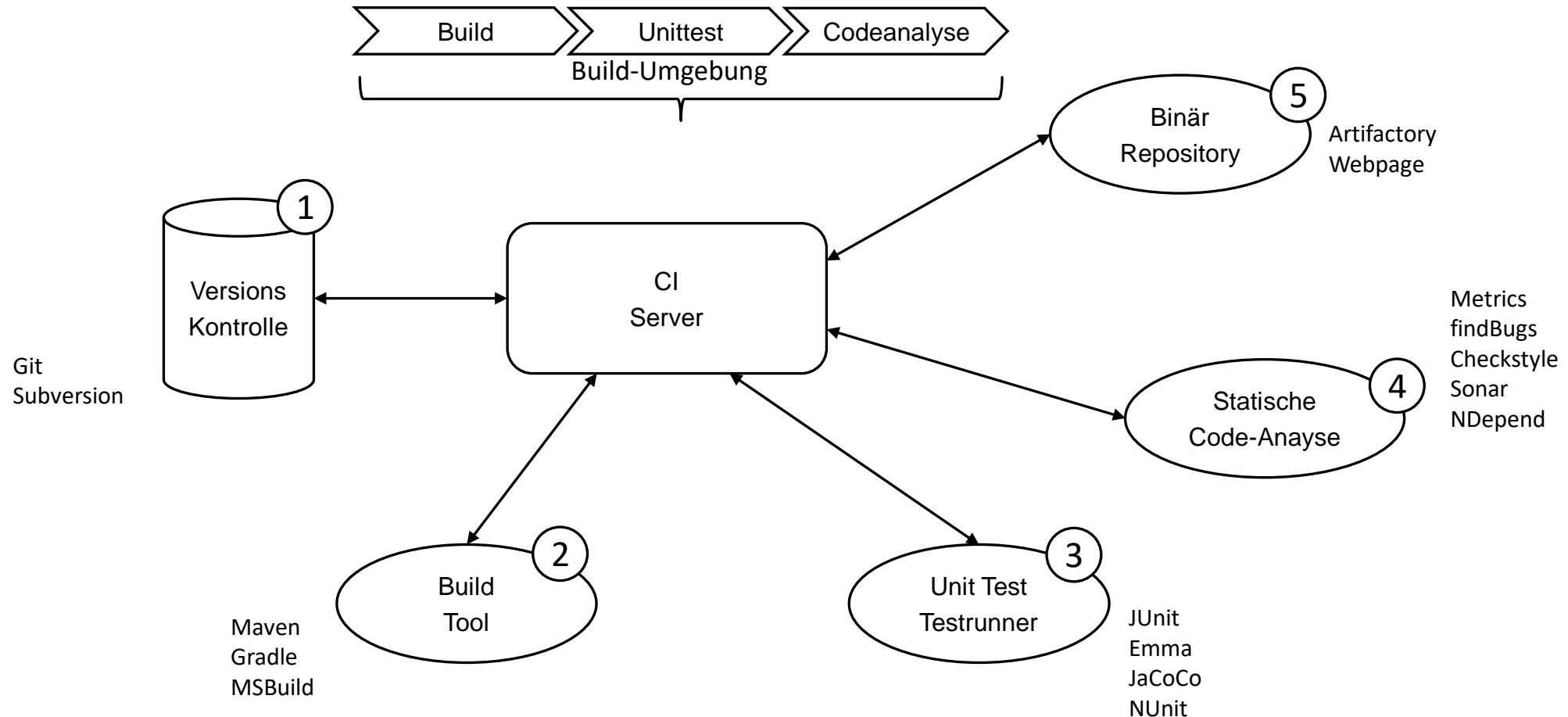
Beispiel: Technische Metriken

- Durchschnittlich verarbeitet HTTP-Requests
- Durchschnittliche Verarbeitungsdauer eines HTTP-Requests
- Anzahl von HTTP-Requests, die mit einem Fehler beantwortet werden
- CPU-Auslastung (über die Zeit gemessen)
- Arbeitsspeicherauslastung (steigend, greift der Garbage Collector etc.)
- Netzwerkauslastung (Peaks)
- Latenzen
- ...

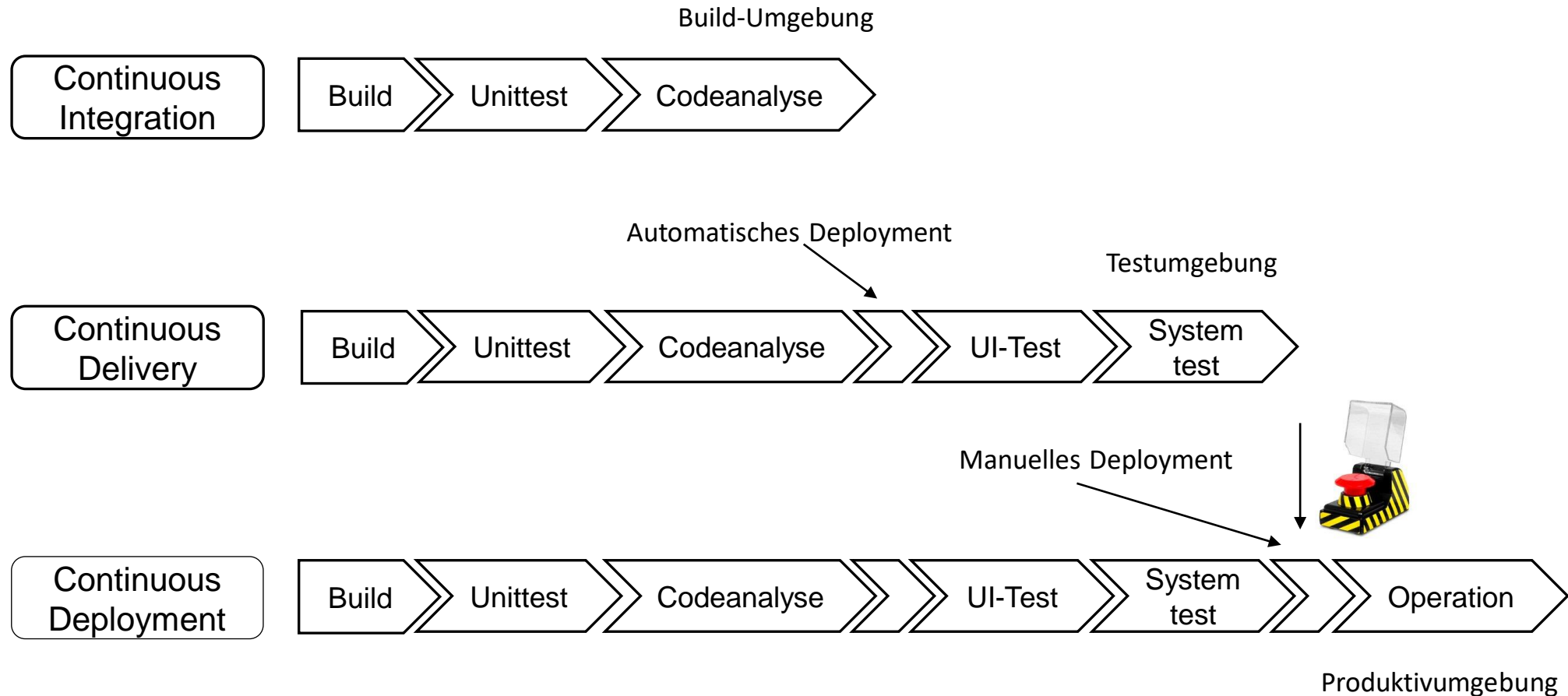
Beispiel: Fachliche Metriken eines Web Shops

- Anzahl Neuregistrierung innerhalb der letzten Stunde
- Heutige Kündigungsquote im Vergleich zum selben Tag der Vorwoche
- Anzahl an- bzw. abgemeldeter Kunden am Newsletter
- Durchschnittlicher Einkaufs-/Warenkorbwert am heutigen Tag
- Umsatz in € über den Web Shop an diesem Tag
- Anzahl der abgeschickten Bestellungen in der letzten Stunde und deren Bezahlungsart
- ...

Continuous Integration



Continuous Deployment



Feedback und verbessertes Lernen

- Produktideen sind nichts anderes als Hypothesen.
- Je schneller eine Organisation Hypothesen einzeln oder parallel in Produktion bringt und aussagekräftige Metriken über den Erfolg oder Misserfolg sammeln kann, desto früher können aus diesen Informationen neue Hypothesen gebildet werden, und der Zyklus beginnt erneut.
- Das Ergebnis ist eine Organisation, die ihre Kunden nach und nach immer besser versteht.
- Dasselbe gilt auch für externe Impulse durch Wettbewerber oder neue Technologien: Je kürzer der Feedbackzyklus ist, desto schneller kann die Organisation reagieren.

Zusammenfassung

- Sie haben einige Probleme kennen gelernt, denen mit DevOps begegnet werden kann
- Sie haben gelernt, was mit DevOps erreicht werden kann
- Sie haben einige organisatorische als auch technische Aspekte kennen gelernt, die im DevOp-Ansatz angewandt werden
 - Cross Functional Teams
 - T-Shaped Skills
 - Selbstständiges Arbeiten
 - Metriken
 - CI/CD

Hausaufgabe

Lesen Sie

Kasteleiner, B., Schwartz, A. *DevOps*. Informatik Spektrum **42**, 211–214 (2019). <https://doi.org/10.1007/s00287-019-01173-2>

Link: <https://rdcu.be/b709M>

Welche Punkte sind Ihnen in dem Artikel nicht klar?

Benennen Sie konkret drei Sachverhalte / Fragen zur Besprechung in der nächsten Einheit!

Drei Punkte als Dokument in ILIAS in entsprechender Übungsaufgabe hochladen!

Acknowledgments

Diese Vorlesungseinheit basiert auf einem Gastvortrag von Markus Kiss der Fa. diva-e an der Fakultät für Informatik der Hochschule Heilbronn.

Literatur

- [1] Kim G., Behr K., Spafford G., Projekt Phoenix, O'Reilly, 2015, ISBN 978-3958751750
- [2] Kasteleiner, B., Schwartz, A. *DevOps*. Informatik Spektrum **42**, 211–214 (2019). <https://doi.org/10.1007/s00287-019-01173-2>