

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

CI/CD-Pipelines, Qualität und deren Sicherung

Christopher J. Ruwe <cjr@cruwe.de>

21. Juni 2018

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

whoami

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Christopher J. Ruwe
- freiberuflicher DevOps-Consultant mit Schwerpunkten Automatisierung, Continuous-Delivery, Deployment *3} und "Verifikation"
- Slides: <https://www.cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Software- und Systemarchitektur

wie ist eine Applikation eigentlich gebaut?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Front-
end

wie ist eine Applikation eigentlich gebaut?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Front-
end

Back-
end

wie ist eine Applikation eigentlich gebaut?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Front-
end

Back-
end

Daten-
Bank

wie ist eine Applikation eigentlich gebaut?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Front-
end

Back-
end

Daten-
Bank

The screenshot shows a web application interface for a message group titled "Frietings". At the top, it indicates the language is "italiano" and provides a search bar. Below the search bar, there are tabs for "All", "Untranslated", "Outdated", and "Translated". The main content area displays a message titled "A typical view of Frietings" with a description of an image. A yellow box highlights a text input field containing "Your translation" and a button labeled "Save translation". Below the input field, there are buttons for "Skip to next" and "Save translation". The message list below shows several entries, each with a "Description of an image" and a "Description of an image" button. The bottom of the interface features a navigation bar with "List", "Page", and "Proofread" buttons.

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Verantwortlichkeiten

Perspektivwechsel - Organisationsaufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

- wer entwickelt?

Perspektivwechsel - Organisationsaufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- wer entwickelt?

- wer betreibt?

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Perspektivwechsel - Organisationsaufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

- wer entwickelt?
- wer betreibt?
- wer übergibt was an wen?

wie ist eine Applikation eigentlich gebaut?

Firewall

Load-balancer

Front-end

Back-end

Daten-Bank

Message group: **All - Frietings** Translate to: italiano -

Translation of the wiki page **Frietings**

All Untranslated Outdated Translated ...

Frietings / Edit

Translationschritt 1 X Description of an image / Edit description

A typical view of Frietings

Your translation

Press "ALT-SHIFT-S" to save or "ALT-SHIFT-O" to skip to next message

Frietings is a small municipality in Mangeland, located on the Røllinge Island. It has a pop... / Edit

It doesn't have many services. There is a shop and a car ferry visits the island from the mainland... / Edit

Wikipedia events: "The roof of the church fell down in 2009" New Church was built in 2017 / Edit

[Category:Municipalities] / Edit

/

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

wer macht jetzt was?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Front-
end

Frontend-
Entwickler

Entwicklungs-
Team/Firma

Back-
end

Backend-
Entwickler

wer macht jetzt was?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Front-
end

Frontend-
Entwickler

Entwicklungs-
Team/Firma

Back-
end

Backend-
Entwickler

Daten-
Bank

Backend-
Entwickler

wer macht jetzt was?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Front-
end

Frontend-
Entwickler

Entwicklungs-
Team/Firma

Back-
end

Backend-
Entwickler

Daten-
Bank

~~Backend-
Entwickler~~

Daten-
Banker !!!

wer macht jetzt was?

Firewall

Load-Balancer

Front-end

Frontend-Entwickler

Entwicklungs-Team/Firma

Back-end

Backend-Entwickler

Daten-Bank

~~Backend-Entwickler~~

Daten-Banker !!!

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

wer macht jetzt was?

Firewall

<- das ist keine Software-Entwicklung, das macht der Betrieb!

Load-Balancer

Front-end

Frontend-Entwickler
Entwicklungs-Team/Firma

Back-end

Backend-Entwickler

Daten-Bank

~~Backend-Entwickler~~

Daten-Banker !!!

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

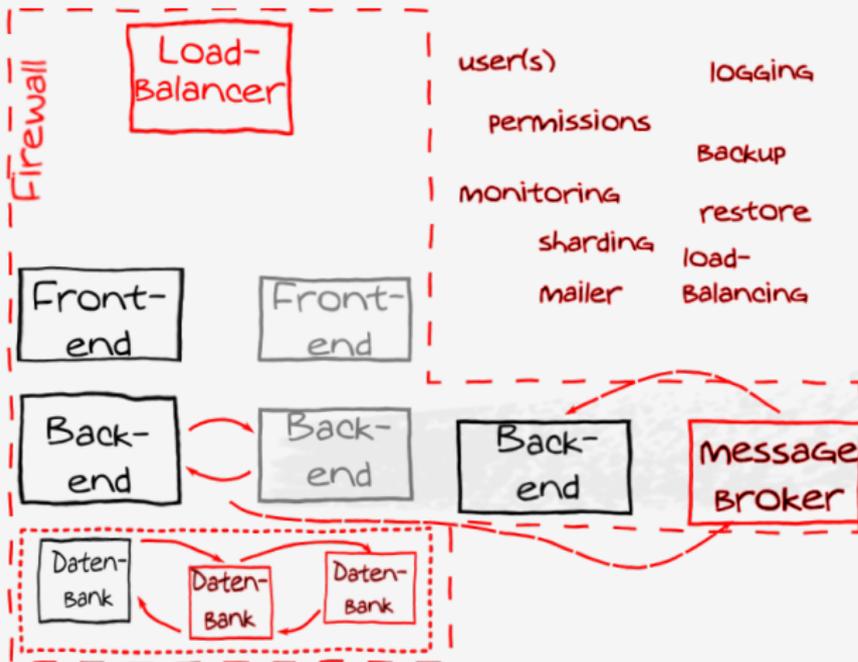
Verifikation

Anspruch und
Thesen

wer macht jetzt was?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

und im Fehlerfall?

- Das war gar nicht meine Applikation, die down war, das war die Datenbank!

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

und im Fehlerfall?

- Das war gar nicht meine Applikation, die down war, das war die Datenbank!
- Das Datenbank war gar nicht down, es konnte nur keiner drauf zugreifen. Der LDAP war nämlich down!

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

und im Fehlerfall?

- Das war gar nicht meine Applikation, die down war, das war die Datenbank!
- Das Datenbank war gar nicht down, es konnte nur keiner drauf zugreifen. Der LDAP war nämlich down!
- Der LDAP war gar nicht down, das war der Netzwerk-Change bei den Kollegen in Portugal, oder vielleicht der in Frankreich.

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

und im Fehlerfall?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

- Also, auf jeden Fall war die Applikation gar nicht down, das fühlte sich BLOß so an. Jawoll ja!

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Applikations-
Verantwortlicher

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Frontend-
Entwickler

Applikations-
Verantwortlicher

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Frontend-
Entwickler

Backend- Applikations-
Entwickler Verantwortlicher

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Frontend-
Entwickler

Backend- Applikations-
Entwickler Verantwortlicher

Daten-
Banker

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

"die anderen"
Entwickler

Frontend-
Entwickler

Backend-
Entwickler Applikations-
Verantwortlicher

Daten-
Banker

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

"die anderen"
Entwickler

Frontend-
Entwickler

Backend- Applikations-
Entwickler Verantwortlicher
Connectivity

Daten-
Banker

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

"die anderen"
Entwickler

Frontend-
Entwickler

Backend-
Entwickler

Applikations-
Verantwortlicher

Connectivity

mobile
Connectivity

Daten-
Banker

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

"die anderen"
Entwickler

Frontend-
Entwickler

Backend-
Entwickler

Applikations-
Verantwortlicher

Connectivity

Firewall-
Team

mobile
Connectivity

Daten-
Banker

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Certificate-
Authority

"die anderen"
Entwickler

Frontend-
Entwickler

Backend-
Entwickler

Applikations-
Verantwortlicher

Connectivity

Firewall-
Team

mobile
Connectivity

Daten-
Banker

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

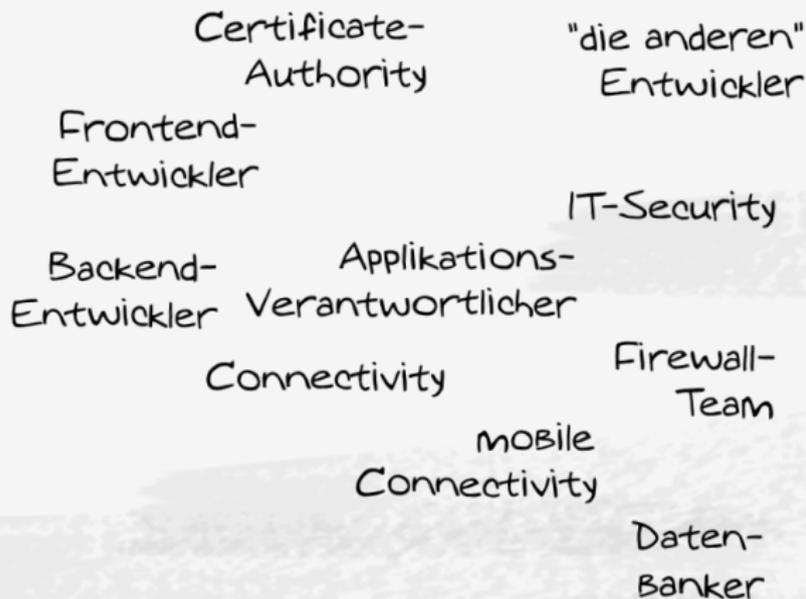
Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

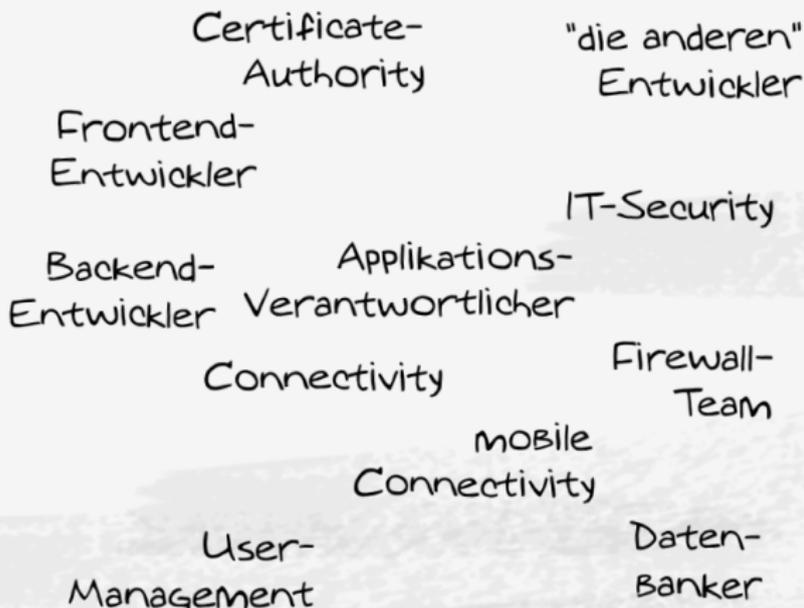
Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

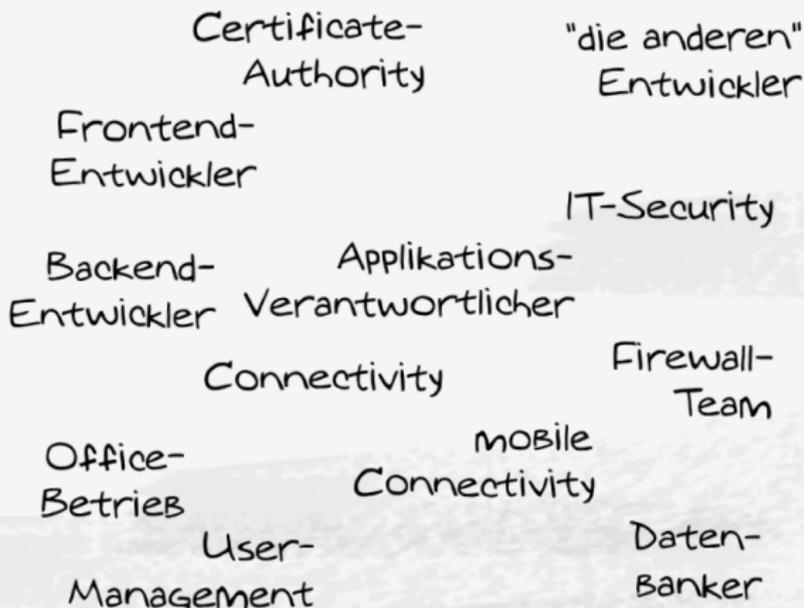
Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- ist Höhepunkt im Unternehmensalltag,

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- ist Höhepunkt im Unternehmensalltag,
- festigt den Zusammenhalt der Belegschaft (Gemeinsamer Feind eint!),

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- ist Höhepunkt im Unternehmensalltag,
- festigt den Zusammenhalt der Belegschaft (Gemeinsamer Feind eint!),
- führt über den Weg der Root-Cause-Analysis zu einem besseren Systemverständnis und

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Das Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- ist Höhepunkt im Unternehmensalltag,
- festigt den Zusammenhalt der Belegschaft (Gemeinsamer Feind eint!),
- führt über den Weg der Root-Cause-Analysis zu einem besseren Systemverständnis und
- damit langfristig zu einem stabileren Gesamt-System! Jawoll ja!

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Versagen

SLA

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- im Ergebnis war die Applikation gar nicht down .

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- im Ergebnis war die Applikation gar nicht down .
- Sie war BLOß nicht BENUTZBAR, aber das ist etwas anderes.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- im Ergebnis war die Applikation gar nicht down .
- Sie war bloß nicht benutzbar, aber das ist etwas anderes.
- Und weil die Applikation nicht down war, habe ich meine Verfügbarkeits-Anforderung gehalten .

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- im Ergebnis war die Applikation gar nicht down .
- Sie war bloß nicht benutzbar, aber das ist etwas anderes.
- Und weil die Applikation nicht down war, habe ich meine Verfügbarkeits-Anforderung gehalten .
- Insofern ist der Bonus meiner Abteilung auch nicht gefährdet.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA - Die Zweite

- Das kann aber nun mit Fug und Recht jeder für sich reklamieren,

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwede.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA - Die Zweite

- Das kann aber nun mit Fug und Recht jeder für sich reklamieren,
- der Frontender, der Backendler, die Datenbank, ...

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA - Die Zweite

- Das kann aber nun mit Fug und Recht jeder für sich reklamieren,
- der Frontender, der Backendler, die Datenbank, ...
- Verteile ich also meine Applikation auf immer Abteilungen, so erreichen immer mehr Abteilungen ihre jeweiliger SLAs.

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA - Die Zweite

- Das kann aber nun mit Fug und Recht jeder für sich reklamieren,
- der Frontender, der Backendler, die Datenbank, ...
- Verteile ich also meine Applikation auf immer Abteilungen, so erreichen immer mehr Abteilungen ihre jeweiliger SLAs.
- Im arithmetischen Mittel über alle Abteilungen steigt also die Erfüllung der SLA.

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

SLA - Die Zweite

- Das kann aber nun mit Fug und Recht jeder für sich reklamieren,
- der Frontender, der Backendler, die Datenbank, ...
- Verteile ich also meine Applikation auf immer Abteilungen, so erreichen immer mehr Abteilungen ihre jeweiliger SLAs.
- Im arithmetischen Mittel über alle Abteilungen steigt also die Erfüllung der SLA.
- Merkwürdigerweise aber sinkt die Verfügbarkeit dieser Applikation gleichzeitig.

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Ursachen

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Rücksprung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Was ist mein Prozeß? -> Service Bereitstellen!

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Rücksprung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Was ist mein Prozeß? -> Service Bereitstellen!
- Gibt es prinzipielle Unterschiede zwischen den Arbeitsschritten der Abteilungen? -> eigentlich machen wir alle IT.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Rücksprung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Was ist mein Prozeß? -> Service Bereitstellen!
- Gibt es prinzipielle Unterschiede zwischen den Arbeitsschritten der Abteilungen? -> eigentlich machen wir alle IT.
- Haben wir verschiedene Ziele? -> Nein?!

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Revisiting the Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

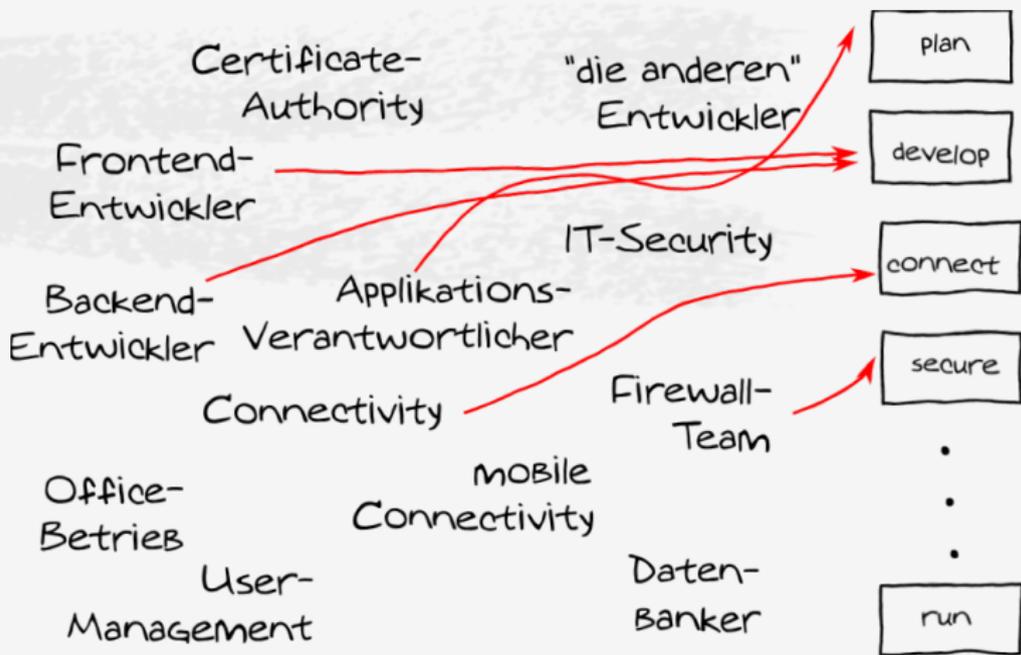
Verifikation

Anspruch und
Thesen

Revisiting the Blame-Game

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

"Werkstück"

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe

<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

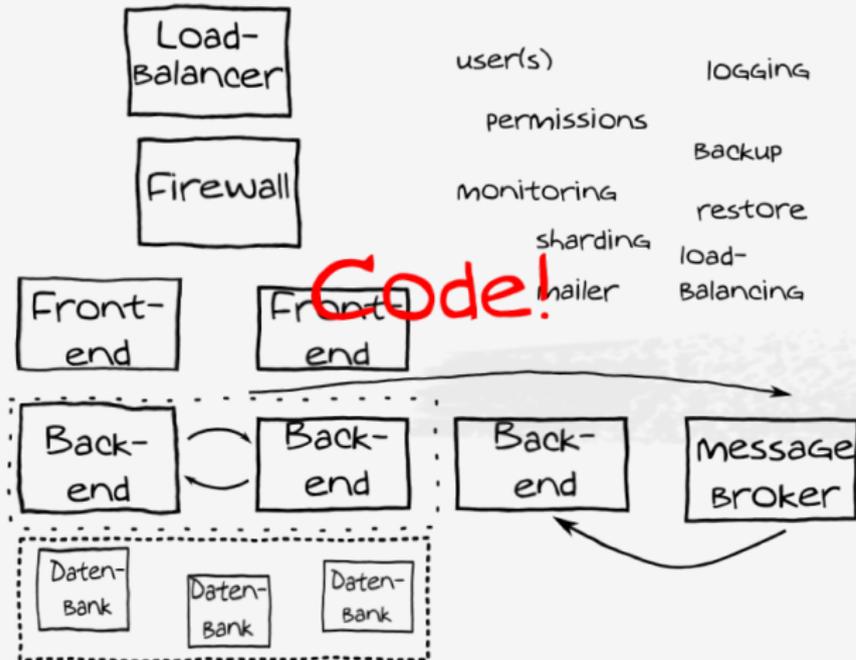
Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen



Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

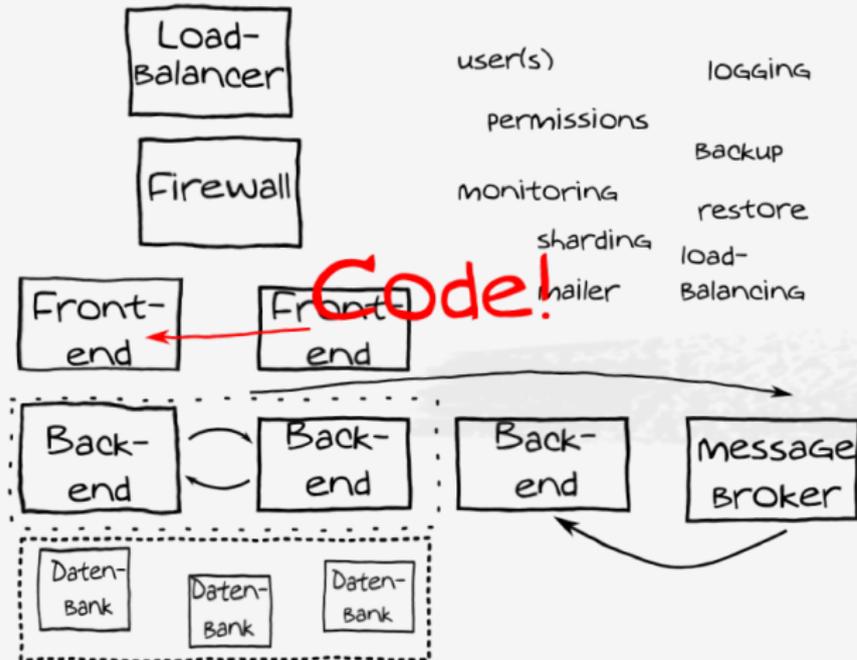
Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen



Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

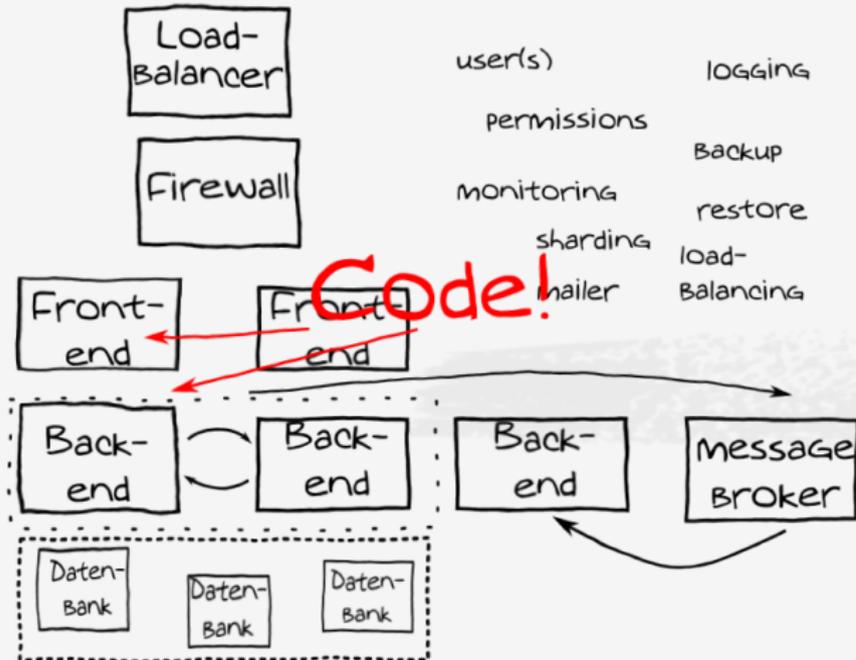
Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen



Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

```
class {'::nginx' }  
  
nginx::resource::server { 'www.cruwe.de'  
  www_root => '/var/www/www.cruwe.de',  
}
```

Calling a Puppet Module - Stub

Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

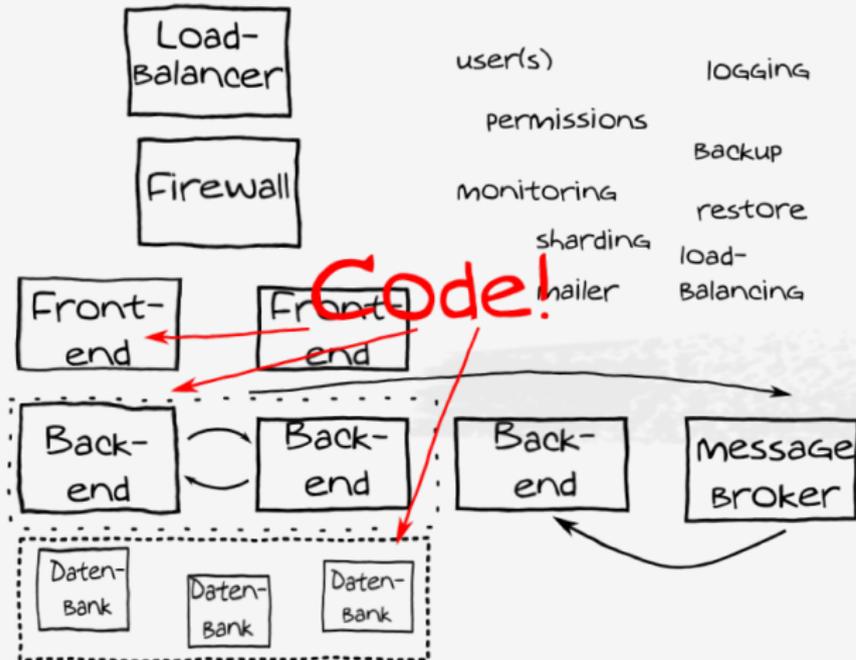
Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen



Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

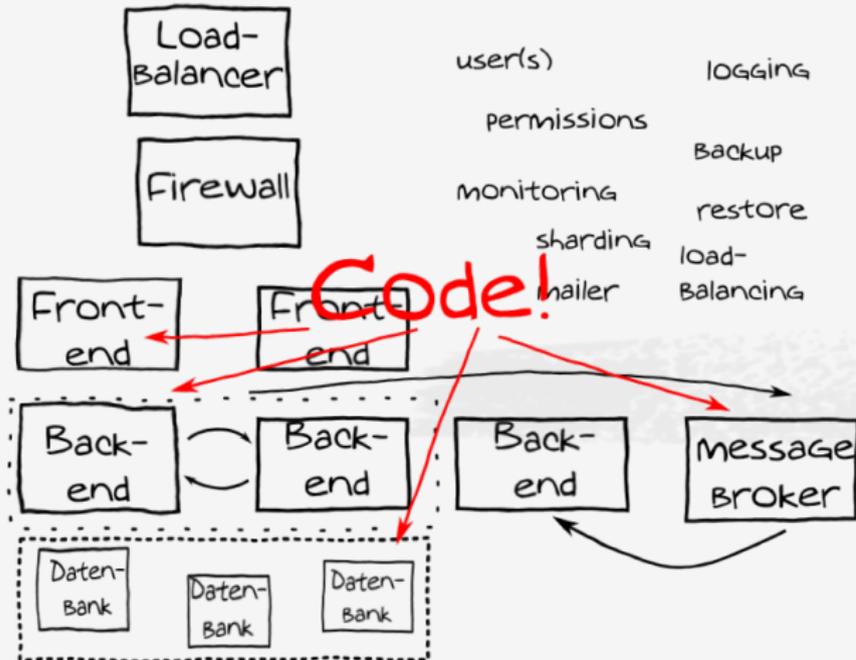
Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen



Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
---
apiVersion: apps/v1beta1
kind: StatefulSet
metadata:
  name: redis-cluster
  labels: <...>
spec:
  serviceName: redis-cluster
  replicas: 6
  template:
    metadata:
      labels: <...>
    spec:
      containers:
      - name: redis-cluster
        image: artifactory.hb22.cruwe.de:8082/redis
        imagePullPolicy: Always
        ports:
        - containerPort: 6379
          name: client
        - containerPort: 16379
          name: gossip
  <...>
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Declaring a Redis Cache - Kubernetes

Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

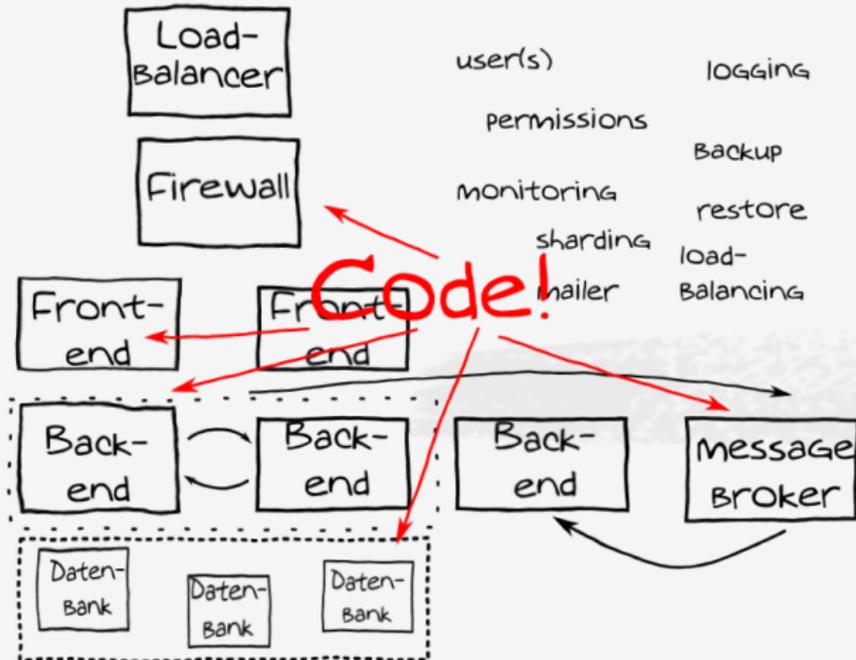
Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

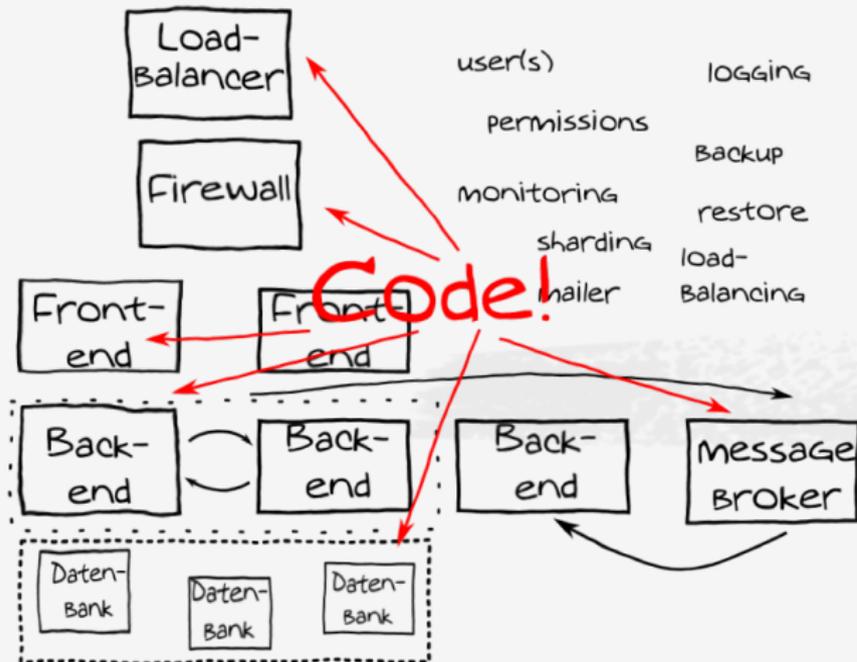
Anspruch und
Thesen



Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

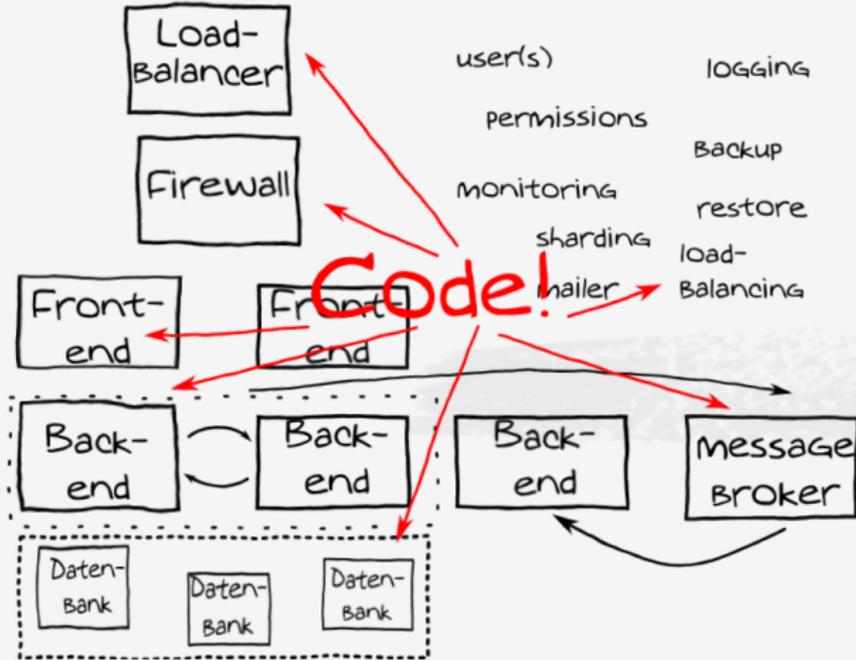
Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen



Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: any-to-ingress-80-443
spec:
  ingress:
  - from:
    ports:
    - protocol: TCP
      port: 80
    - protocol: TCP
      port: 443
  podSelector:
    matchLabels:
      app: app-ingress
<...>
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

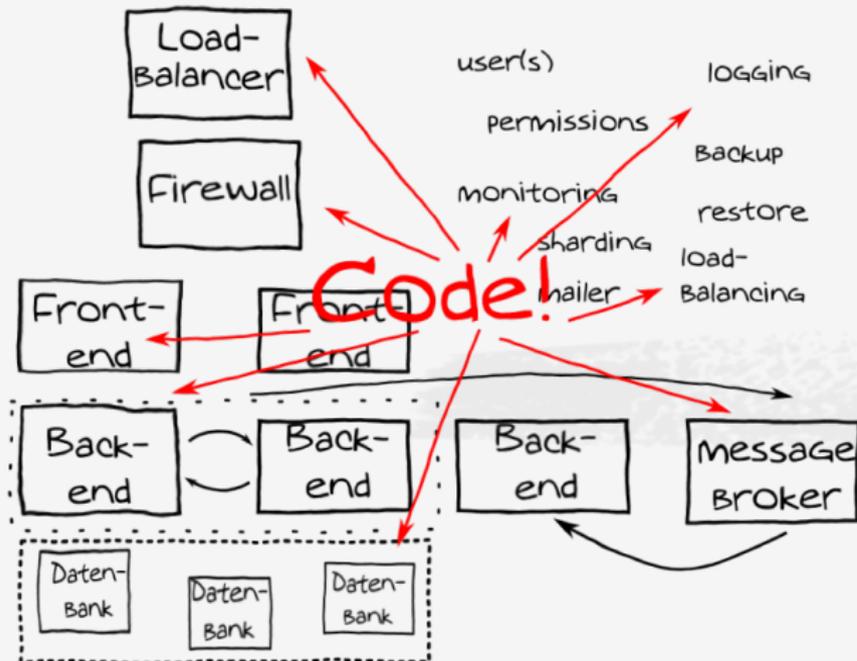
Anspruch und
Thesen

Network Policies - Kubernetes

Everything is Code - Sort Of

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Produktionsstrecke

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

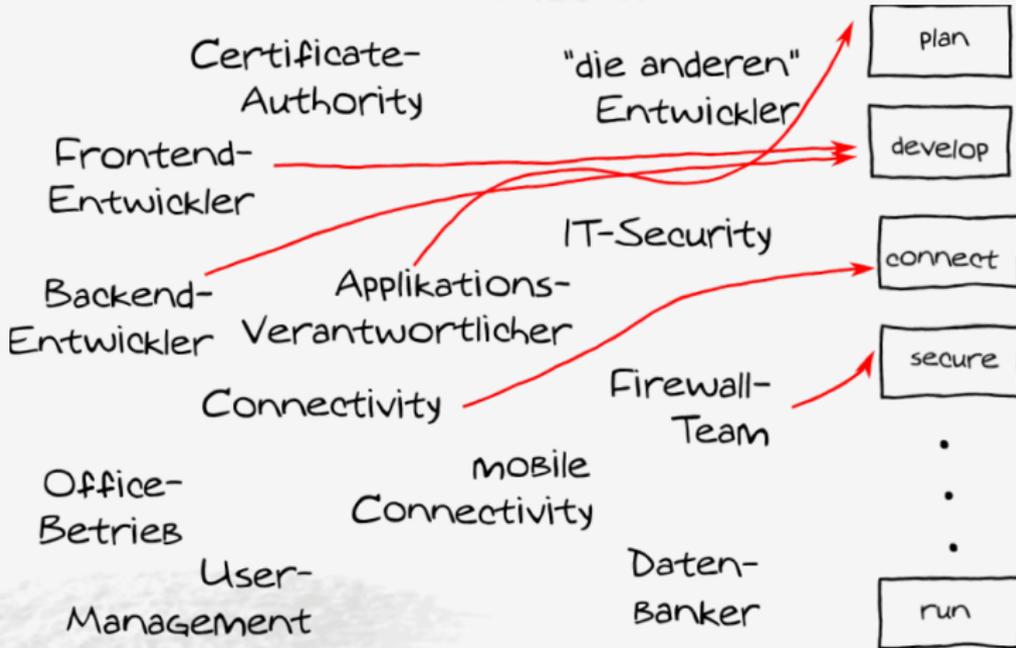
"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Rücksprung



CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwede>

whoami

Software- und
Systemarchitektur
Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Rücksprung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Niemandem ist geholfen, wenn das Wurf-Objekt über den Abteilungs-Zaun statt eines passiv-aggressiven ITIL-Tickets Code ist!

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

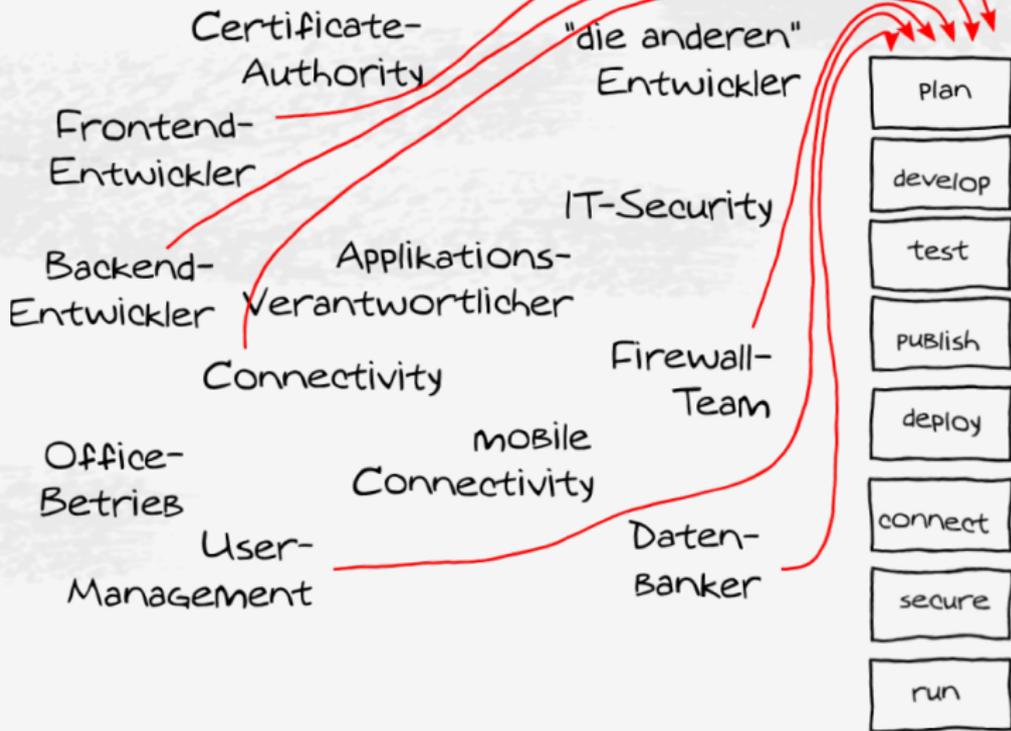
"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Pipelines



CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

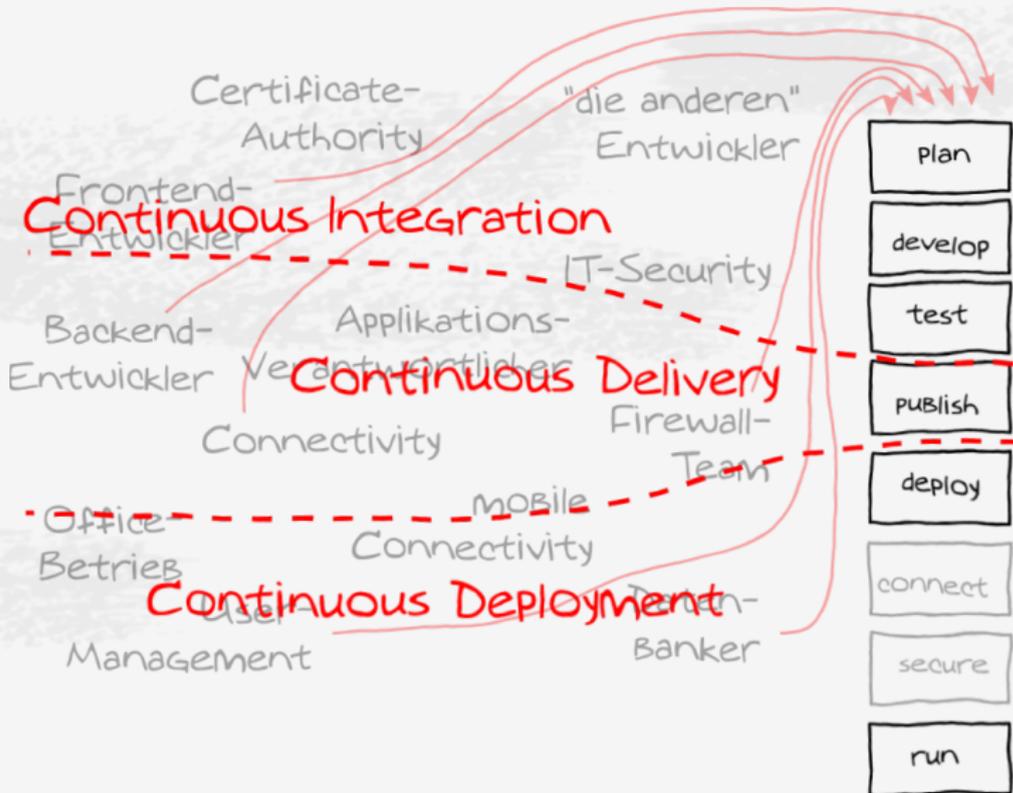
Verifikation

Anspruch und
Thesen

Pipelines

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Automatisierung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Für Continuous {Integration, Delivery,
Deployment, * } brauchen wir Automatisierung.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Automatisierung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Für Continuous {Integration, Delivery, Deployment, * } brauchen wir Automatisierung.
- Automatisierter Mist ist aber immer noch zuerst und zuvorderst ... Mist!

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Automatisierung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Für Continuous {Integration, Delivery, Deployment, * } brauchen wir Automatisierung.
- Automatisierter Mist ist aber immer noch zuerst und zuvorderst ... Mist!
- Das ist aber kein Problem, wir stellen nur fehlerfreie Software her.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Automatisierung

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Für Continuous {Integration, Delivery, Deployment, * } brauchen wir Automatisierung.
- Automatisierter Mist ist aber immer noch zuerst und zuvorderst ... Mist!
- Das ist aber kein Problem, wir stellen nur fehlerfreie Software her.
- Fehler machen immer nur die anderen .

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Verifikation

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Beobachtungen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

- Meist weiß man nicht, was in der Umgebung läuft.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Beobachtungen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

- Meist weiß man nicht, was in der Umgebung läuft.
- Meist weiß man auch nicht, welche Änderungen welche Auswirkungen haben.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Beobachtungen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

- Meist weiß man nicht, was in der Umgebung läuft.
- Meist weiß man auch nicht, welche Änderungen welche Auswirkungen haben.
- Deswegen haben viele Betriebs-Organisationen riesigen Respekt vor Änderungen

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Beobachtungen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

- Meist weiß man nicht, was in der Umgebung läuft.
- Meist weiß man auch nicht, welche Änderungen welche Auswirkungen haben.
- Deswegen haben viele Betriebs-Organisationen riesigen Respekt vor Änderungen
- und erst recht vor schnellen und automatischen Änderungen!

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Beobachtungen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

- Meist weiß man nicht, was in der Umgebung läuft.
- Meist weiß man auch nicht, welche Änderungen welche Auswirkungen haben.
- Deswegen haben viele Betriebs-Organisationen riesigen Respekt vor Änderungen
- und erst recht vor schnellen und automatischen Änderungen!
- These: mit Sichtbarkeit steht und fällt alles

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

(a?)typische Test-Situationen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Backup-Tests ohne Restore,

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

(a?)typische Test-Situationen

- Backup-Tests ohne Restore,
- Last-Tests auf 16 MB DSL,

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

(a?)typische Test-Situationen

- Backup-Tests ohne Restore,
- Last-Tests auf 16 MB DSL,
- Load-Balancer, Sticky-Sessions und mein Browser-Cache,

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

(a?)typische Test-Situationen

- Backup-Tests ohne Restore,
- Last-Tests auf 16 MB DSL,
- Load-Balancer, Sticky-Sessions und mein Browser-Cache,
- Databases in Memory - In Memory only!

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

was wird nicht getestet?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik
Test-Ausführung
Test-Pipeline
Test-Integration

Anspruch und
Thesen

Front-
end

Back-
end

Daten-
Bank

Message group: [All «Friebings»](#) Translate to: [italiano](#)

Translation of the wiki page [Friebings](#).

[All Untranslated](#) [Outdated](#) [Translated](#) ... [Filter list](#)

[Friebings](#) / Edit

Translation ✕ Description of an image

A typical view of Friebings

⌵

Press "ALT+SHIFT+S" to save or "ALT+SHIFT/O" to skip to next message

[Friebings is a small municipality in Murgtal, located on the Balgunga Island. It has a pop...](#) Outdated / Edit

[It doesn't have many species. There is a shop and a car ferry along the island from the mainla...](#) Outdated / Edit

[Wald events: "The roof of the church fell down in 2009" | View church was built in 1877](#) / Edit

[\[\[Category:Municipalities\]\]](#) / Edit

was wird nicht getestet?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwede>

Load-
balancer

Front-
end

Back-
end

Daten-
Bank

The screenshot shows a web interface for translating the word 'Frétinga'. At the top, it says 'Message group: All - Frétinga' and 'Translate to: italiano'. Below that, it says 'Translation of the wiki page Frétinga'. There are tabs for 'All', 'Untranslated', 'Outdated', and 'Translated'. A search bar contains 'Frétinga'. The main content area shows a translation form with a text input containing 'Your translation', a 'Paste source text' button, and 'Save translation' and 'Skip to next' buttons. To the right, there is a 'Description of an image' section with an 'Edit description' link. Below the form, there are three list items, each with an 'Outdated' status and an 'Edit' link. The first item says 'Frétinga is a small municipality in Múgla, located on the Bakkungu Island. It has a pop...'. The second item says 'It doesn't have many services. There is a shop and a car ferry to the island from the main...'. The third item says 'Main events: * The roof of the church fell down in 2019 * New church was built in 1877'. At the bottom, there are navigation buttons for 'List', 'Page', and 'Proofread'.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik
Test-Ausführung
Test-Pipeline
Test-Integration

Anspruch und
Thesen

was wird nicht getestet?

Firewall

Load-Balancer

Front-end

Back-end

Daten-Bank

Message group: **All - Freitings** Translate to **italiano**

Translation of the wiki page **Freitings**

All **Untranslated** Outdated Translated ... Filter list

Freitings / Edit

Translation description: **Freitings** Description of an image / Edit description

A typical view of Freitings

Your translation

Press "ALT-SHIFT-S" to save or "ALT-SHIFT-Q" to skip to next message

Freitings is a small municipality in Mangel, and located on the Baskungs Island. It has a pop... / Edit

It doesn't have many services. There is a shop and a car ferry to the island from the main... / Edit

Main events: * The roof of the church fell down in 2009 * New church was built in 1877 / Edit

[Category: municipalities] / Edit

/

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe

<cjr@cruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

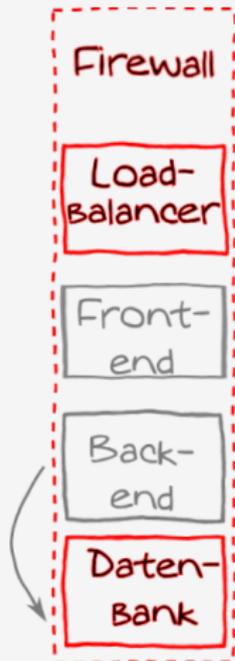
Test-Systematik
Test-Ausführung
Test-Pipeline
Test-Integration

Anspruch und
Thesen

was wird nicht getestet?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwede>



Message group: [All - Frétinga](#) Translate to [italiano](#)

Translation of the wiki page [Frétinga](#).

All [Untranslated](#) [Outdated](#) [Translated](#) ...

Frétinga / Edit

Translationschritt 6/6 ✕ Description of an image

A typical view of Frétinga

Your translation

[Paste source text](#)

[Save translation](#) [Skip to next](#)

Press "ALT-SHIFT-0" to save or "ALT-SHIFT-2" to skip to next message

[/ Edit description](#)

Frétinga is a small municipality in [Municipalities](#), located on the [Blaðungu](#) island. It has a pop... Outdated / Edit

It doesn't have many services. There is a shop and a car ferry visits the island from the mainla... Outdated / Edit

Main events: "The roof of the church fell down in 2009" "New Church was built in 1877" / Edit

[\[Category: Municipality\]](#) / Edit

[List](#) [Page](#) [Proofread](#)

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

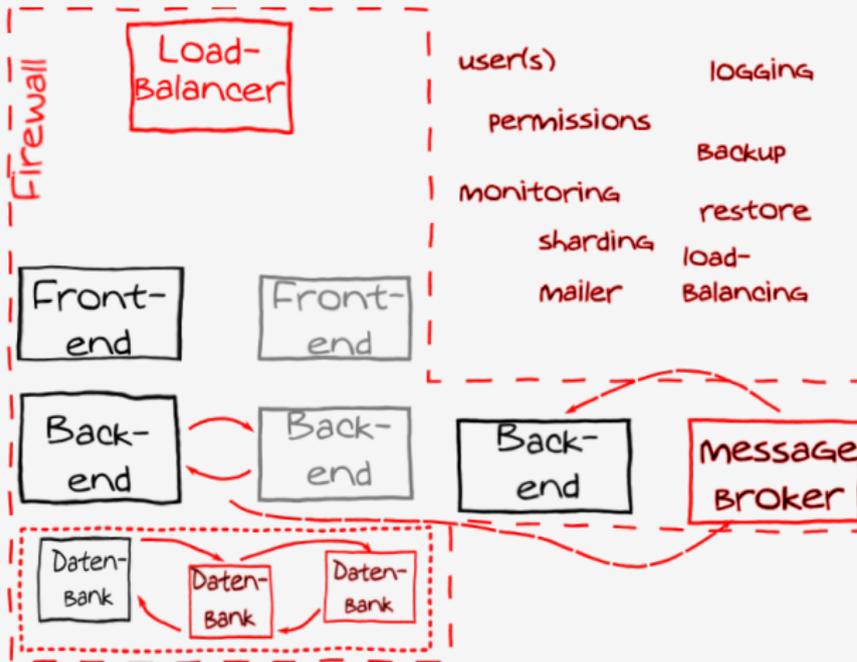
Test-Systematik
Test-Ausführung
Test-Pipeline
Test-Integration

Anspruch und
Thesen

was wird nicht getestet?

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@ruwede>



whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

Verifikation - Test-Systematik

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und

Thesen

Test-Systematik

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

„Testen“ ist eine Blöde Bezeichnung - eigentlich ist es
die Abnahme!

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

„Werkstück“

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Test-Systematik

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Unit-Tests (Modul-Test, Komponenten-Test, ...),

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Test-Systematik

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Unit-Tests (Modul-Test, Komponenten-Test, ...),
- Integrations-Tests

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Test-Systematik

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Unit-Tests (Modul-Test, Komponenten-Test, ...),
- Integrations-Tests
- Akzeptanz-Test (Abnahme-Test)

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Unit

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

NGINX

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
class {'::nginx' }  
  
nginx::resource::server { 'www.cruwe.de'  
  www_root => '/var/www/www.cruwe.de',  
}
```

Calling a Puppet Module – Stub

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

Unit

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

NGINX

```
class {::nginx} {  
  
  nginx_resource_server { 'www.cruwede'  
    www_root => '/var/www/www.cruwede'  
  }  
}
```

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

Unit

Integration /
Komposit

NGINX

NGINX + files
+ user + ...

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

```
class { 'nginx' }
```

```
nginx.resource.server { 'www.cruwede'  
  www_root => '/var/www/www.cruwede'  
}
```

+

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwede.de>

```
$vhostroots = keys($vhosts).map |String $vhost| {  
  {  
    "/var/www/${vhost}" => {  
      ensure => 'directory',  
      owner  => 'www-data',  
      group  => 'adm',  
      mode   => '0770',  
      require => File['/var/www'],  
    },  
  }  
}.reduce({}) |$coll, $elem| {  
  merge($coll, $elem)  
}  
ensure_resources('file', $vhostroots)
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und

Thesen

Configuring a Webserver Root Directory - Stub

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<ejr@cruwede>

Unit

Integration /
Komposit

NGINX

NGINX + files
+ user + ...

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

```
class { 'nginx' }
```

```
nginx.resource.server { 'www.cruwede'  
  www_root => '/var/www/www.cruwede'  
}
```

+

```
$hostroots = keys($hosts).map |String $host| {  
  {  
    "/var/www/${host}" => {  
      ensure => 'directory',  
      owner  => 'www-data',  
      group  => 'adm',  
      mode   => '0710',  
      require => File['/var/www'],  
    }  
  }  
}  
$reduce($?) |$coll, $elem| {  
  merge($coll, $elem)  
}  
ensure_resources('file', $hostroots)
```

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

Unit

Integration /
Komposit

Gesamt-
System

NGINX

NGINX + files
+ user + ...

NGINX + files
+ user + ...  + files
+ user + ...

 MySQL + files
+ user + ...

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
classes:  
  - role::www  
  
role::www::vhosts:  
  'www.cruwe.de':  
    www_root: '/var/www/www.cruwe.de'  
    rewrite_to_https: true  
    ssl: true  
    ssl_cert: <...>  
    ssl_key: <...>  
    ipv6_enable: true  
    ipv6_listen_options: ''  
    add_header:  
      X-Clacks-Overhead: 'GNU Terry Pratchett'
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
'cruwe.de':  
  use_default_location: false  
  ipv6_enable: true  
  ipv6_listen_options: ' '  
  index_files: [ ]  
  location_custom_cfg: { }  
  vhost_cfg_append:  
    rewrite:  
      '^/(.*) https://www.cruwe.de/$1 permanent'
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Configuring a Webserver - Stub

System-Aufbau

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

Unit

Integration /
Komposit

Gesamt-
System

NGINX

NGINX + files
+ user + ...

NGINX + files
+ user + ...  + files

 + files
+ user + ...

classes:

- role:www

role:www:vhosts:

'www.cruwede':

www_root: '/var/www/www.cruwede'

rewrite_to_https: true

ssl: true

ssl_cert: '/etc/ssl/certs/le_www.cruwede-fullchain.pem'

ssl_key: '/etc/ssl/certs/le_www.cruwede-privkey.pem'

ipv6_enable: true

ipv6_listen_options: "

add_header:

X-Clacks-Overhead: 'GNU Terry Pratchett'

'cruwede':

use_default_location: false

ipv6_enable: true

ipv6_listen_options: ''

index_files: []

location_custom_ofs: { 3 }

vhost_ofs_append:

rewrite: "/(.*?) https://www.cruwede/ permanent"

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und

Thesen

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

Verifikation - Test-Ausführung

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Unit Test - RSpec

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede>

Unit



Unit
Test

NGINX

```
class {::nginx} {
```

```
  nginx_resource.server { 'www.cruwede'
```

```
    www_root => '/var/www/www.cruwede'
```

```
}
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Unit Test - RSpec

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
describe 'nginx::resource::server' do
  let(:title) { 'www.cruwe.de' }
  let(:params) {
    { 'www_root' => '/var/www/www.cruwe.de' }
  }

  it { is_expected.to contain_class('nginx') }
  it do
    is_expected.to
      contain_file('/etc/nginx/sites-enabled')
        .with(
          'ensure' => 'present',
          'owner'  => 'root',
          <...>
        )
  end
end
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Integration Test - ServerSpec

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwede.de>

Integration /
Komposit



Integration
Test

NGINX + files
+ user + ...

```
class :nginx {
```

```
  nginx::resource::server { 'www.cruwede'  
    www_root => '/var/www/www.cruwede'  
  }  
}
```

```
$vhostroots = keys($hosts).map |String $vhost| {  
  {  
    "/var/www/$vhost" => {  
      ensure => 'directory',  
      owner  => 'www-data',  
      group  => 'adm',  
      mode   => '0710',  
      require => File['/var/www'],  
    },  
  },  
}  
}reduce($?) { $coll, $elem } {  
  merge($coll, $elem)  
}  
ensure_resources('file', $vhostroots)
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Integration Test - ServerSpec

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

```
describe 'Check the webserver via ssh.' do
  describe service('nginx') do
    it { should be_enabled }
    it { should be_running }
  end

  describe process('nginx') do
    its(:user) { should eq 'www-data' }
    its(:count) { should eq 4 }
  end

  describe port(80) do <...> end
end
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Test a Webserver Locally - Stub

Integration Test - ServerSpec

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
describe 'Check the webserver via http.' do
  describe command("curl -I http://www.cruwe.de") do
    its(:stdout) { should match /HTTP\/1.1 301/ }
    its(:stdout) { should match
      /Location: https:\\\/<...>/ }
    its(:stdout) { should match /Server: nginx/ }
    its(:stdout) { should match
      /Content-Type: text\/html/ }
    its(:stdout) { should match /X-Clacks-Overhead:
      GNU Terry Pratchett/ }
  end
end
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

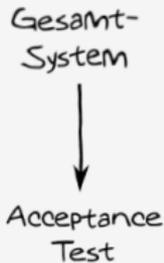
Anspruch und
Thesen

Test a Webserver Remotely - Stub

Acceptance Test

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>



NGINX + files + files
+ user + ... + user + ...

MySQL + files
+ user + ...

classes:

- role:insultsweb

role:insultsweb:domain: insults.cruwe.de

< >

classes:

- role:insultapp

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Acceptance Test

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
describe 'Check www from localhost.' do
  <...>
end

describe 'Check app-server from frontend.' do
  <...>
end
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Walk the Servers

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Verifikation - Test-Pipeline

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Test Pipeline

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@ruwede>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

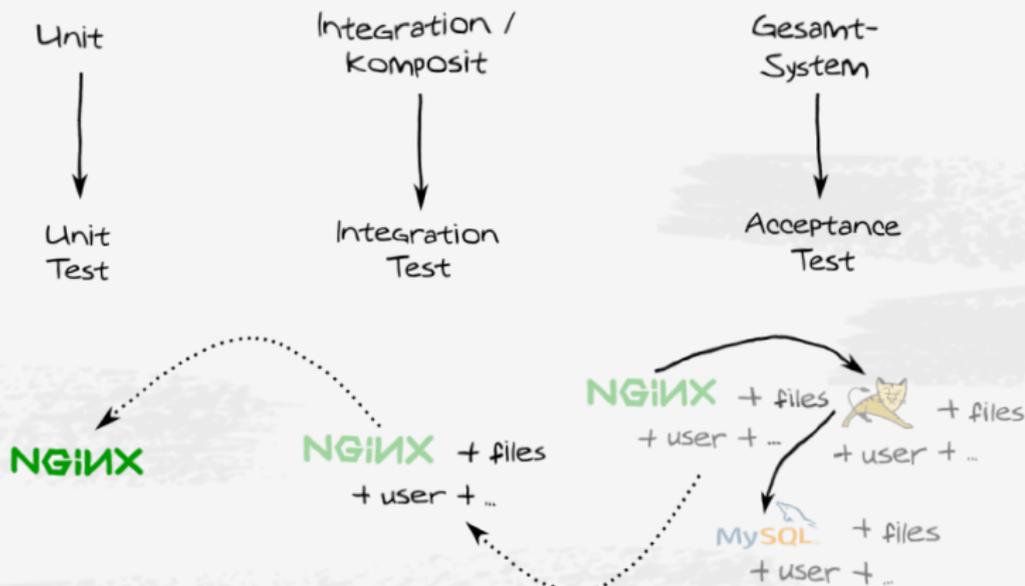
Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen



Test Pipeline

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
node('buildslave') {  
  
    properties([  
        buildDiscarder(logRotator(numToKeepStr: '25')),  
        disableConcurrentBuilds(),  
        <...>  
    ])  
  
    stage('Checkout') {  
        cleanWs()  
        checkout(scm)  
    }  
}
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Pipeline – Properties and Checkout

Test Pipeline

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

```
stage('Prepare environment') {  
  parallel(  
    'build_docker_image': {  
      sh '''#!/bin/bash -l  
        docker build \  
          -t cjr/centos_w_puppet:7.3.1611 \  
          -f Dockerfile.centos73 \  
          .  
      '''  
    }  
  )  
}
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Pipeline - Preparing the Environment

Test Pipeline

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@ruwede.de>

```
stage('Test Dynamically') {  
  try {  
    sh('''#!/bin/bash -l  
        eval $(ssh-agent)  
        bundle exec kitchen create docker  
        bundle exec kitchen converge docker  
        bundle exec kitchen verify docker  
        ''')  
  }  
  catch (err) { } finally {  
    step(  
      [  
        $class      : 'JUnitResultArchiver',  
        allowEmptyResults: true,  
      ]  
    )  
  }  
}
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Pipeline – Simulating a Host with Docker

Test Pipeline

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwede.de>

```
stage('Publish artifact') {  
  withCredentials(  
    [  
      usernamePassword(  
        credentialsId: 'jenkins2nexus',  
        passwordVariable: 'NEXUSPASS',  
        usernameVariable: 'NEXUSUSER'  
      )  
    ]  
  ) {  
    nexusArtifactUploader(  
      artifacts:  
        [[  
          artifactId: "${artifactId}",  
          <...>  
        ]  
      ]  
    )  
  }  
}
```

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Publishing a Packaged Module

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Verifikation - Test-Integration

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Integration in ein Repository

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<crj@cruwe.de>

<reporoot>

└─ kitchen.yml, hosts.ini, ...

└─ test/integration/serverspec

└─ spec_helpers.rb

└─ webserver_spec.rb

└─ Rakefile, Makefile, pom.xml, ...

└─ Jenkinsfile, .travis.yml, Buildspec.yml, ...

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Test-Systematik

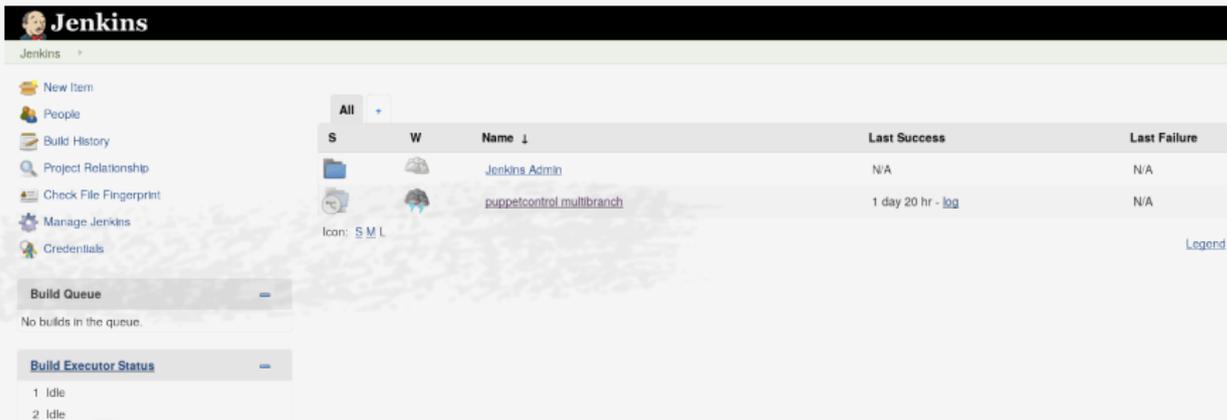
Test-Ausführung

Test-Pipeline

Test-Integration

Anspruch und
Thesen

Ergebnis: Überblick über die Projekte



The screenshot shows the Jenkins web interface. At the top left is the Jenkins logo and name. Below it is a navigation menu with items like 'New Item', 'People', 'Build History', 'Project Relationship', 'Check File Fingerprint', 'Manage Jenkins', and 'Credentials'. On the left side, there are two summary boxes: 'Build Queue' (showing 'No builds in the queue') and 'Build Executor Status' (showing '1 Idle' and '2 Idle'). The main content area displays a table of projects under the 'All' filter. The table has columns for 'S' (Success), 'W' (Warning), 'Name', 'Last Success', and 'Last Failure'. Two projects are listed: 'Jenkins Admin' and 'puppetcontrol multibranch'. Below the table, there are links for 'Icon: S M L' and a 'Legend' link.

Jenkins

- New Item
- People
- Build History
- Project Relationship
- Check File Fingerprint
- Manage Jenkins
- Credentials

Build Queue

No builds in the queue

Build Executor Status

- 1 Idle
- 2 Idle

All

S	W	Name ↓	Last Success	Last Failure
		Jenkins Admin	N/A	N/A
		puppetcontrol multibranch	1 day 20 hr - log	N/A

Icon: [S](#) [M](#) [L](#)

[Legend](#)

Ergebnis: Einzelne Projekt-Sicht

 **Jenkins**

[Jenkins](#) > [puppetcontrol multibranch](#)

[Up](#)

[Status](#)

[Configure](#)

[Scan Multibranch Pipeline Now](#)

[Scan Multibranch Pipeline Log](#)

[Multibranch Pipeline Events](#)

[Delete Multibranch Pipeline](#)

[People](#)

[Build History](#)

[Project Relationship](#)

[Check File Fingerprint](#)

[Move](#)

[Pipeline Syntax](#)

[Credentials](#)

Build Queue —

No builds in the queue.

Build Executor Status —

1 Idle
2 Idle

 **puppetcontrol multibranch**

puppetcontrol from <http://gogs.local:3000/jenkinsdemo/puppetcontrol.git>

Branches (3)

S	W	Name ↓	Last Success	Last Failure
		cjr_triggeredjenkins	1 day 20 hr - #4	1 day 20 hr - #3
		develop	N/A	1 day 19 hr - #2
		master	N/A	N/A

Icon: [S](#) [M](#) [L](#)

Ergebnis: Testläufe auf einen Branch

Average stage times:
(Average full run time: ~3min 39s)

	Checkout	Prepare environment	Test Statically	Test Dynamically	Prepare release	Publish artifact
#25 Oct 13 13:51 1 commits	290ms	18s	3s	2min 37s	334ms	38s
#23 Oct 13 13:45 1 commits	330ms	19s	4s	2min 48s	633ms	22ms failed
#22 Oct 13 13:39 2 commits	303ms	57s	5s	2min 33s		
#1 Oct 13 13:27 No Changes	395ms	18s failed				

Ergebnis: Einzelner Testlauf auf einen Branch

The screenshot shows the Jenkins web interface for a specific build. The top navigation bar includes the Jenkins logo and the breadcrumb path: Jenkins > puppetcontrol nullbranch > cjr_triggerjenkins > #4. A left-hand sidebar contains various navigation links: Back to Project, Status, Changes, Console Output, Edit Build Information, Delete Build, Parameters, Git Build Data, No Tags, Test Result, Replay, Pipeline Steps, and Previous Build. The main content area is titled "Build #4 (Oct 13, 2017 1:51:32 PM)". Below the title, there are several sections: "Changes" with one change listed (repair curbranch); "Started by upstream project Jenkins Admin - gagstriggered" with build number 59, originally caused by a generic cause; "Revision" information from git (7bb823bec008ea7859916a279e81862207ddcd06) for the cjr_triggerjenkins branch; "Puppet-Lint Warnings" showing 0 warnings, with a note that no warnings have occurred since build 4 and that a new zero warnings highscore has been achieved; "Checkstyle" also showing 0 warnings from one analysis, with a similar note about the highscore; and finally, a "Test Result" section indicating no failures.

Jenkins

Jenkins > puppetcontrol nullbranch > cjr_triggerjenkins > #4

- Back to Project
- Status
- Changes
- Console Output
- Edit Build Information
- Delete Build
- Parameters
- Git Build Data
- No Tags
- Test Result
- Replay
- Pipeline Steps
- Previous Build

Build #4 (Oct 13, 2017 1:51:32 PM)

Changes

- repair curbranch ([detail](#))

Started by upstream project [Jenkins Admin - gagstriggered](#) build number [59](#)
originally caused by:

- Generic Cause

Revision: 7bb823bec008ea7859916a279e81862207ddcd06

- cjr_triggerjenkins

Puppet-Lint Warnings: 0 warnings.

- No warnings since build 4.
- New zero warnings highscore: no warnings since yesterday!
- During parsing an [error](#) has been reported.

Checkstyle: 0 warnings from one analysis.

- No warnings since build 4.
- New zero warnings highscore: no warnings since yesterday!

[Test Result](#) (no failures)

Ergebnis: Einzelner Test

Test Result : tests from test/integration/base

0 failures

11 tests

Took 1.1 sec.

 add description

All Tests

Test name	Duration	Status
SSH Configuration Protocol should cmp == 2	0.45 sec	Passed
User bath gid should eq 666	0 ms	Passed
User bath group should eq "bath"	0 ms	Passed
User bath groups should eq ["bath", "adm", "wheel", "users", "systemd-journal"]	0 ms	Passed
User bath home should eq "home/bath"	0.25 sec	Passed
User bath maxdays should be <= 90	2 ms	Passed
User bath mindays should eq 0	0.23 sec	Passed
User bath shell should eq "/bin/bash"	0 ms	Passed
User bath should exist	0.21 sec	Passed
User bath uid should eq 666	0 ms	Passed
User bath windays should eq 7	0 ms	Passed

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Anspruch und Thesen

Tests ...

- ... decken alle Soft- und Hardware-Komponenten über alle beteiligten Systeme ab,

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Tests ...

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- ... decken alle Soft- und Hardware-Komponenten über alle beteiligten Systeme ab,
- ... sind, wenn einmal sauber erstellt, preiswert,

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Tests ...

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- ... decken alle Soft- und Hardware-Komponenten über alle beteiligten Systeme ab,
- ... sind, wenn einmal sauber erstellt, preiswert,
- ... werden bei jeder Änderung vollständig durchgeführt und

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Tests ...

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- ... decken alle Soft- und Hardware-Komponenten über alle beteiligten Systeme ab,
- ... sind, wenn einmal sauber erstellt, preiswert,
- ... werden bei jeder Änderung vollständig durchgeführt und
- ... verschaffen Transparenz - Blind fliegt man schlecht auf Sicht.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Full-Stack-Tests

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Man kann den IT-Stack nicht einfrieren.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Full-Stack-Tests

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Man kann den IT-Stack nicht einfrieren .
- Aufbau und Änderungen müssen aber verifiziert
-> abgenommen werden .

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Full-Stack-Tests

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Man kann den IT-Stack nicht einfrieren .
- Aufbau und Änderungen müssen aber verifiziert
-> abgenommen werden .
- Diese Abnahme muß in jeden Prozeß-Schritt
eingebaut werden .

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Full-Stack-Tests

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Man kann den IT-Stack nicht einfrieren .
- Aufbau und Änderungen müssen aber verifiziert
-> abgenommen werden .
- Diese Abnahme muß in jeden Prozeß-Schritt
eingebaut werden .
- Tests führen zu Sicherheit, dies hält man im
Allgemeinen für a Good Thing™.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Test-Integration

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Tests sind eine automatische Prüfung gegen die Spezifikation, geschickterweise gehört das in den Infrastruktur-Code integriert:

- Test-Aufbau -> hosts.ini, kitchen, ...

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Test-Integration

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Tests sind eine automatische Prüfung gegen die Spezifikation, geschickterweise gehört das in den Infrastruktur-Code integriert:

- Test-Aufbau -> hosts.ini, kitchen, ...
- Test-Spezifikation -> rspec, serverspec, inspec, ...

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Test-Integration

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Tests sind eine automatische Prüfung gegen die Spezifikation, geschickterweise gehört das in den Infrastruktur-Code integriert:

- Test-Aufbau -> hosts.ini, kitchen, ...
- Test-Spezifikation -> rspec, serverspec, inspec, ...
- Ausführungs-Logik -> Rakefile, makefile, JOB-DSL, custom code ... ,

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Test-Integration

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

Tests sind eine automatische Prüfung gegen die Spezifikation, geschickterweise gehört das in den Infrastruktur-Code integriert:

- Test-Aufbau -> hosts.ini, kitchen, ...
- Test-Spezifikation -> rspec, serverspec, inspec, ...
- Ausführungs-Logik -> Rakefile, makefile, JOB-DSL, custom code ... ,
- Test-Ablauf -> Jenkins Workflow Pipeline, travis, ...

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Continuous { Integration, Delivery, Deployment, * }

- Continuous * ist in erster Linie eine prozessuale und organisatorische Änderung, erst danach eine technologische.

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Continuous { Integration, Delivery, Deployment, * }

- Continuous * ist in erster Linie eine prozessuale und organisatorische Änderung, erst danach eine technologische.
- Geschwindigkeit wird nicht durch schnelles, sondern durch nachgewiesen passendes und damit robusteres Deployment gewonnen .

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen

Continuous { Integration, Delivery, Deployment, * }

CI/CD-Pipelines,
Qualität und deren
Sicherung

Christopher J.
Ruwe
<cjr@cruwe.de>

- Continuous * ist in erster Linie eine prozessuale und organisatorische Änderung, erst danach eine technologische.
- Geschwindigkeit wird nicht durch schnelles, sondern durch nachgewiesenes passendes und damit robusteres Deployment gewonnen.
- Regelmäßige Tests unterscheiden sich nicht vom Zustands- und Verfügbarkeits-Monitoring. Monitoring ist der Input, um ein IT-System zu fahren. Ohne Input wird es zufällig.

whoami

Software- und
Systemarchitektur

Verantwortlichkeiten

Versagen

Ursachen

"Werkstück"

Produktionsstrecke

Verifikation

Anspruch und
Thesen