15.–18. 09. 2008
in Nürnberg

Herbstcampus

Wissenstransfer
par excellence

Liquide Mittel

Kontinuierliche Integration mit Hudson

Martin Heider

infomar software

Referent Martin Heider

Martin Heider arbeitet seit mehr als 15 Jahren im Bereich Software-Entwicklung. Als Freiberufler unterstützt er Kunden in verschiedenen Rollen als SW-Entwickler, SW-Architekt, Testmanager, Team- und Entwicklungsleiter oder Coach. Seine Erfahrung umfasst international verteilte Projekte sowie Teams verschiedener Größen. Sein besonderes Interesse gilt agilen Methoden und der Herausforderung Software-Entwicklung einfach zu machen, damit alle Beteiligten mit mehr Spaß bessere Ergebnisse erzielen.

Sie erreichen mich unter: martin.heider@infomar.de

Agenda

- Kontinuierliche Integration
  - Ziel
  - Grundprinzipien
  - Addressierte Risiken
  - Besser, schneller, günstiger
  - Der Build und seine Inhalte
  - Wie starten?
  - Mit 7 Schritten in den CI Himmel

Agenda

- Hudson
  - Historie …
  - Funktionsweise
  - Features
  - Einrichtung
  - Plugins
  - Erfahrungen
- Ein kleiner Film zum Schluß

Kontinuierliche Integration
Ziel

Wir wollen
Software … entwicklen

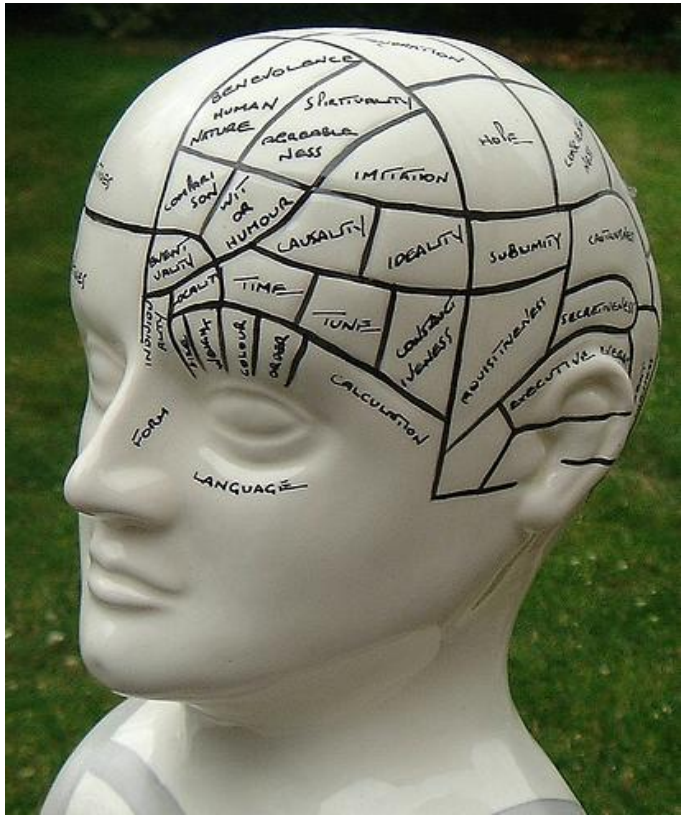
Günstiger
Transparenter
Schneller
Zuverlässiger
Besser

Kontinuierliche Integration
Grundprinzipien

Integration
ist aufwändig
darum machen wir
es jetzt öfter ☺

Kontinuierliche Integration
Grundprinzipien

- Entwicklungsmethode
  - Mehr Mindeset als Technik
  - Unterstützt von Tools



Kontinuierliche Integration
Grundprinzipien

- Entwickler integrieren fortlaufend
  - Kein Problemstau
  - Im Fluß bleiben

Kontinuierliche Integration
Grundprinzipien

- Vollautomatisierter Build
  - „One green button“
  - „Stop the Line“

Kontinuierliche Integration
Grundprinzipien

- Was es nicht ist
  - Nightly Builds
  - Entwickler Branches
  - Vereinbarte Integrationstermine
  - Bauen mit der IDE

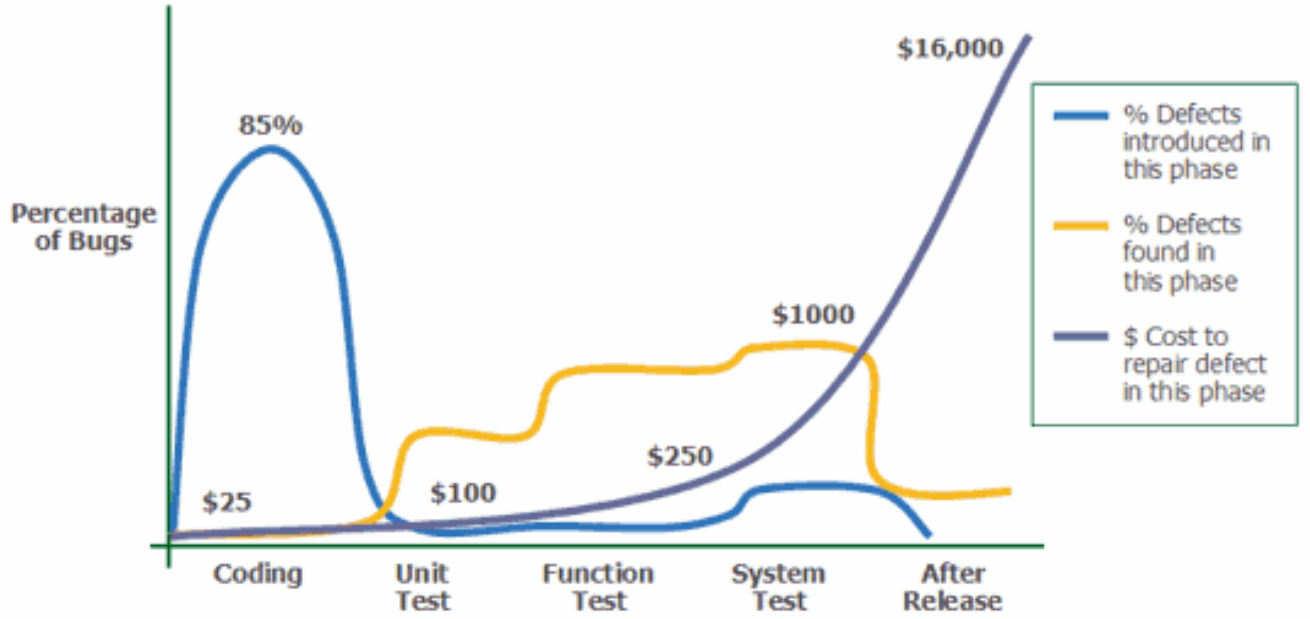
Kontinuierliche Integration
Addressierte Risiken

zur bisherigen Integration

Zu Risiken fragen Sie Ihren
Projektleiter oder Integrator

Kontinuierliche Integration
Addressierte Risiken

- Risiko I: Späte Fehlerbehebung ist teurer



http://www.agitar.com/solutions/why_unit_testing.html

Kontinuierliche Integration
Addressierte Risiken

- Risiko II: Mangelnde Teamabstimmung
  - „Deine Änderung passen nicht mit meinen zusammen“
  - „Hattest Du das nicht bereits vor 2 Monaten gefixt“
- Risiko III: Schlechte Code-Qualität
  - „Wieso machen drei verschiedene Klassen das Gleiche?“
  - „Der Code von Team XY schaut ja ganz anders aus“

Kontinuierliche Integration
Addressierte Risiken

- Risiko IV: Mangelnde Transparenz / Sichtbarkeit
  - „Welche Tests laufen nicht?“
  - „Was beinhaltet Build 1.2.3?“
  - „Wo stehen wir mit der Code-Abdeckung?“
- Risiko V: Nicht verfügbare Software
  - „Bei mir geht‘s“
  - „Ich brauche noch einen Build zum Testen“
  - „Morgen kommt der Chef-Chef, wir brauchen eine Demo“

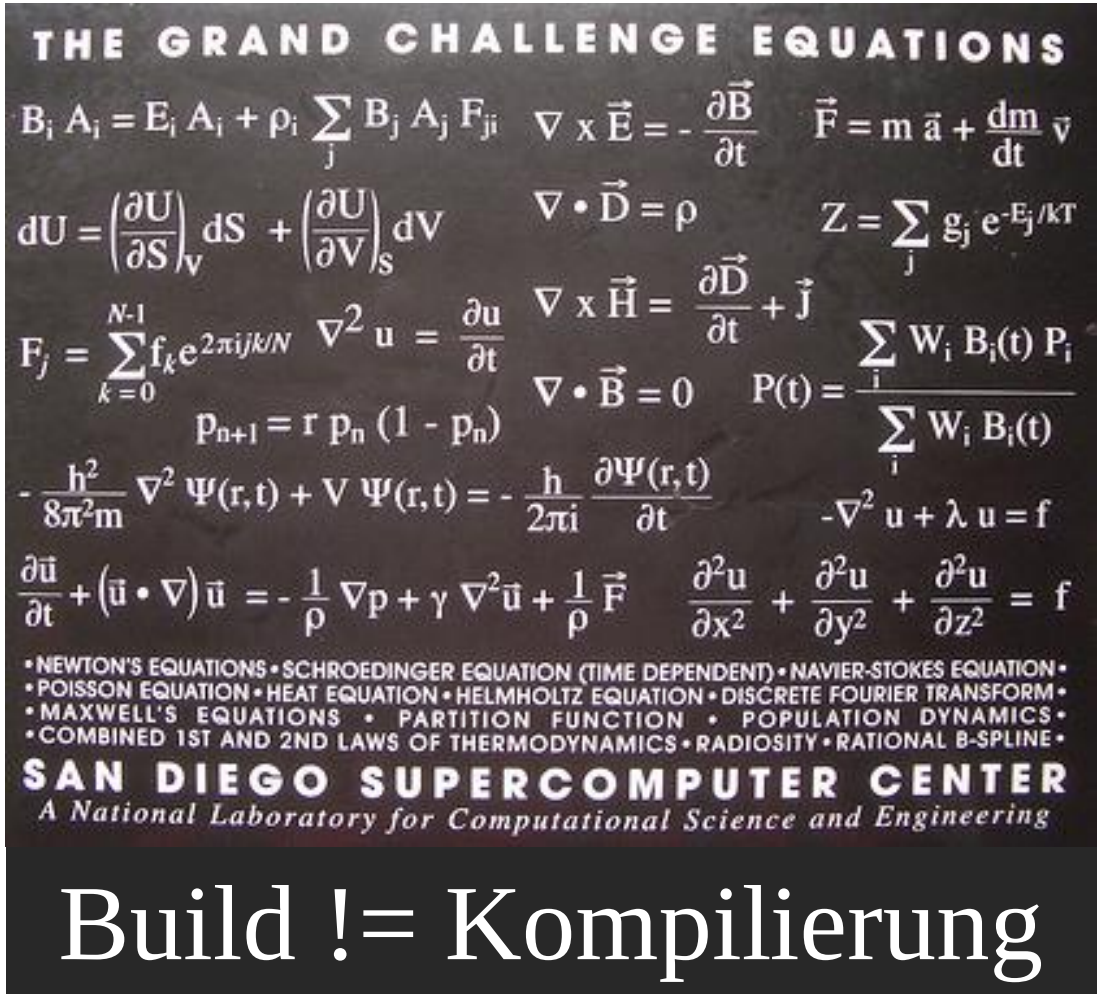
Kontinuierliche Integration
Besser, schneller, günstiger

Kontinuierliche Integration
Besser, schneller, günstiger

- Besser
  - Oft und frühzeitig getestet
  - „Coding Standards“ und „Best Practises“ einhaltend
- Schneller
  - Tests finden parallel zur Entwicklung statt
  - Aufwändige Integrationen werden zum „Nicht-Ereignis“
- Günstiger
  - Fehler werden früher gefunden
  - Behebung der Fehler zum frühesten, günstigsten Zeitpunkt
  - Einfach wiederholbare Tests

Kontinuierliche Integration

Der Build und seine Inhalte …



Kontinuierliche Integration
Der Build und seine Inhalte …

- Kompilierung
- Test Ausführung
  - Unit Tests
  - Akzeptanztests
- Integration
  - Datenbank
  - Drittsysteme
- Statische Analysen
  - Code
  - Architektur
- Automatisches Deployment
- Generierung der Dokumentation

Kontinuierliche Integration
Wie starten?

Kontinuierliche Integration Wie starten?

- Wann baue ich?
  - Nach jeder Code Änderung
  - Nach jeder Änderung von Abhängigkeiten
- Wie baue ich?
  - Mit einem einzigen Build-Skript
  - Startbar auf der Kommandozeile
  - Nicht in Abhängigkeit einer DIE
- Was brauche ich dazu?
  - Konfigurationsmanagement Software
  - CI Software und Server

Kontinuierliche Integration
Wie starten?

- Worauf ist noch zu achten?
  - Schnelles Feedback
  - Einfach zugreifbar
  - Kein Aufwand für Entwickler
  - Schlüsselmetriken identifizieren
  - Wichtiges deutlich visualisieren
  - Auf Schlüsselmetriken sofort reagieren

Ruth Ablett, Ehud Sharlin, Jörg Denzinger, Frank Maurer, Craig Schock
Department of Computer Science, University of Calgary

Build notifications for Agile software development teams

Java Lava Lamps use simple red and green lights to convey build state.
BuildBot uses sounds, lights and movement.
Both compared with Email

Accountability for failed tests in a playful, fun way

Self-supervision: An fun reminder is better than an angry manager!

SPECIAL ROBOT TIMES. EDITION

25 June 2007. Your Best Source of News-worthy Items. Price Two Cents.

ROBOT ATTACKS DEVELOPER!

BuildBot, cute puppy overseer, alerts team to potentially disasterous error.

Software developer slightly embarrassed, unhurt after breaking software build.

Build fixed, development continuing as normal, biscuits enjoyed.

It was a scene of deafening calm with a smattering of giggles Thursday when Bob Bobson, developer employed by YesWe-AreAgile Software, made a small error in his code, accidently omitting a small 23-character method call that would have ensured that his code pass the tests he had previously written for it.

"Well, that was silly, wasn't it," commented the smiling 25-year old native of Agileville, Alberta, Canada, also home to the small software development firm.

When Bobson's code was uploaded to the team repository, the error was caught by the continuous integration server when sixteen of the forty-three prewritten tests failed due to the one-line omission. It was then that the robot watchdog, BuildBot, began to move from its power station towards Bobson's desk.

"At first I didn't think it was me" said Bobson, whose code had not been the cause of any previous build breakages. "It started moving, and we could all hear it, and we all started wondering who broke the build. I was thinking, 'It couldn't be me, or could it?'"

Susie Derkins, the project manager, shrugged when hearing about the breakage. "They break it, they fix it. As long as they are aware and someone's taking steps to fix the situation, I'm not worried at all."

As soon as the cuddly robot approached Bobson's desk, it slowly sat down. BuildBot alerted him via a cute but menacing growl that it was in fact Bobson's breakage. Bobson then corrected the error, BuildBot made its way back to its station and life continued more or less as normal. At that point, Bobson took a break to enjoy some of his mother's homemade ginger-snap biscuits, which he had been saving for just such an occasion.

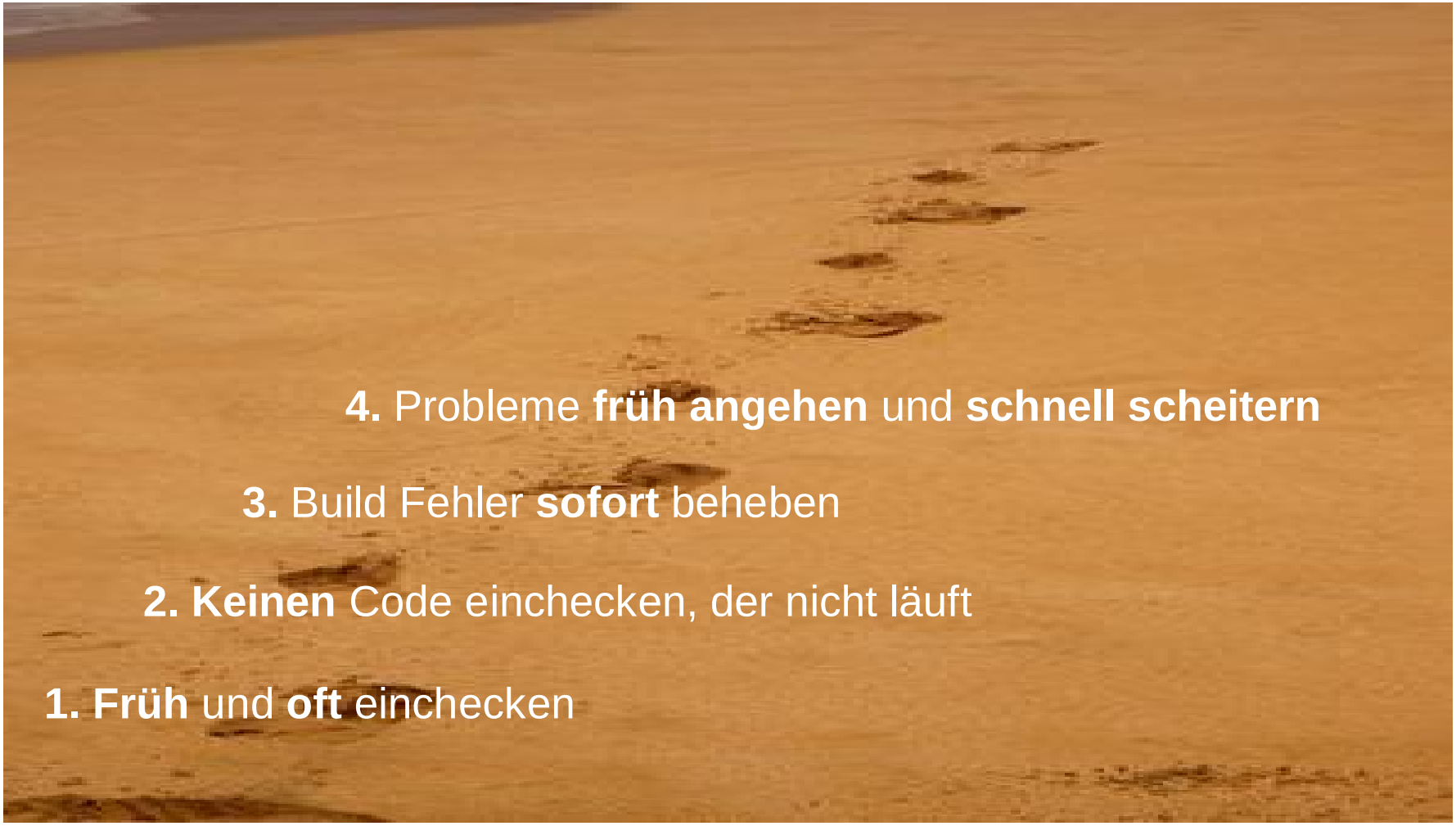
Kontinuierliche Integration 7 Schritte

Kontinuierliche Integration 7 Schritte

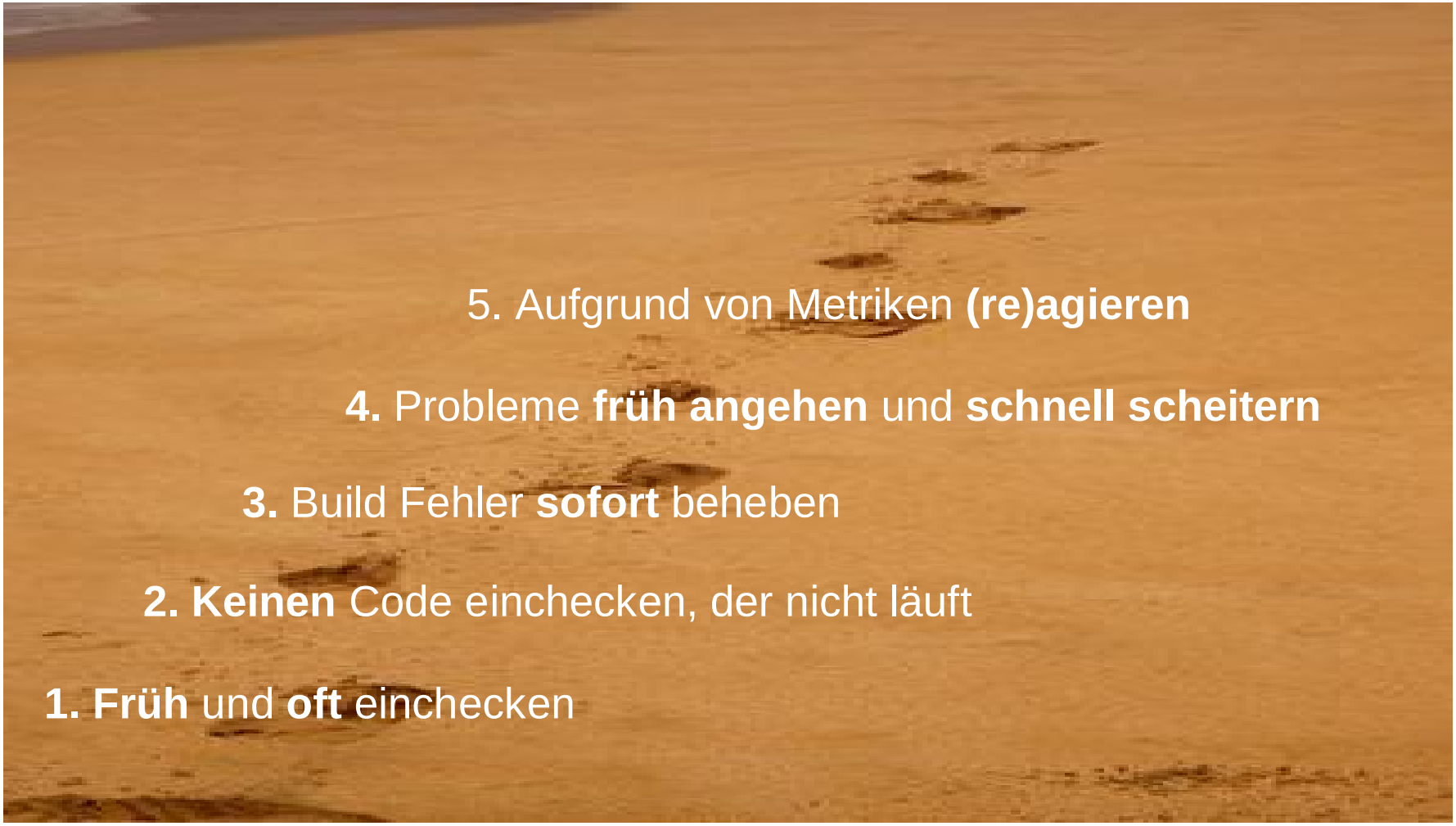
Kontinuierliche Integration 7 Schritte

Kontinuierliche Integration 7 Schritte

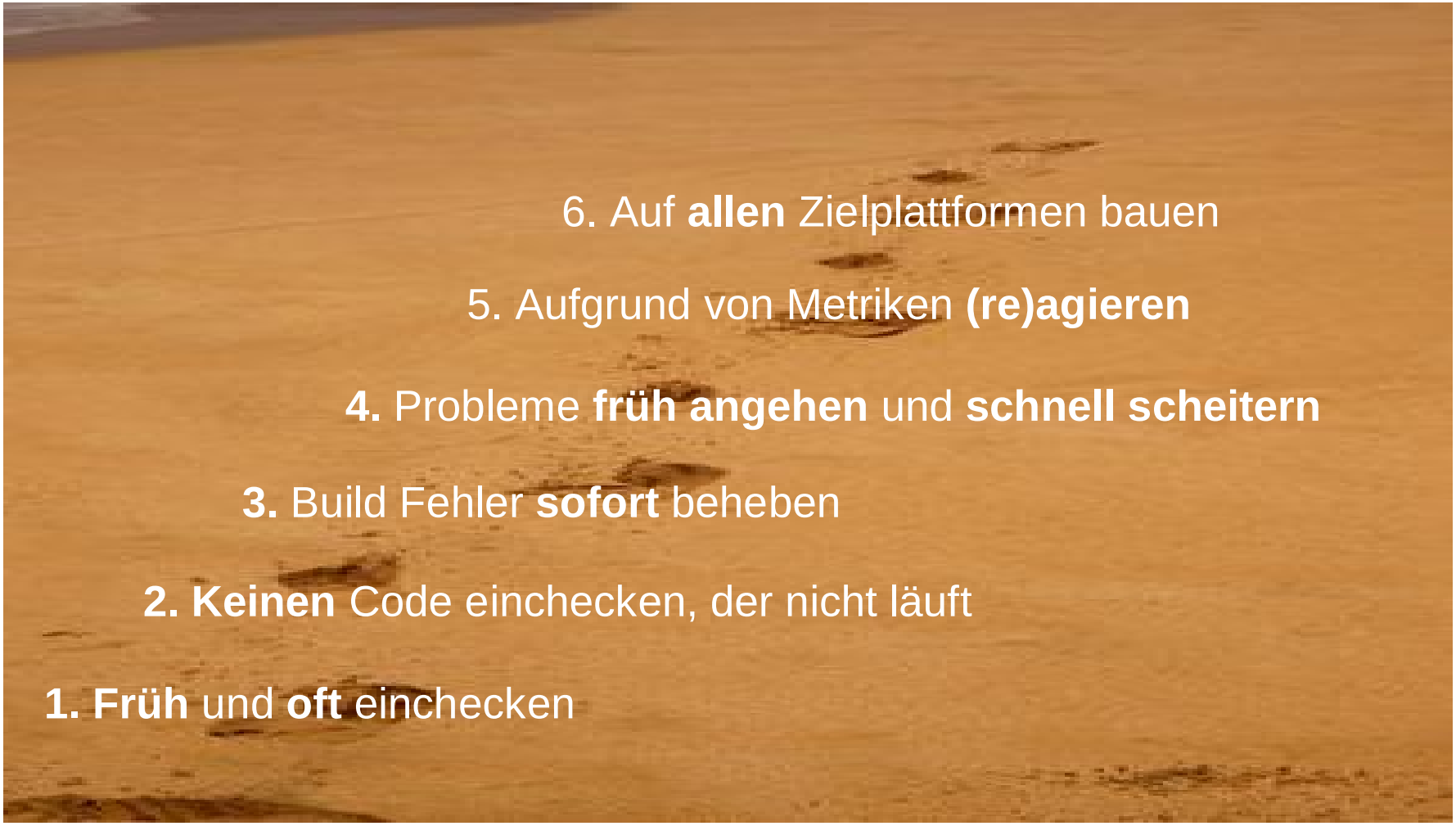
Kontinuierliche Integration 7 Schritte



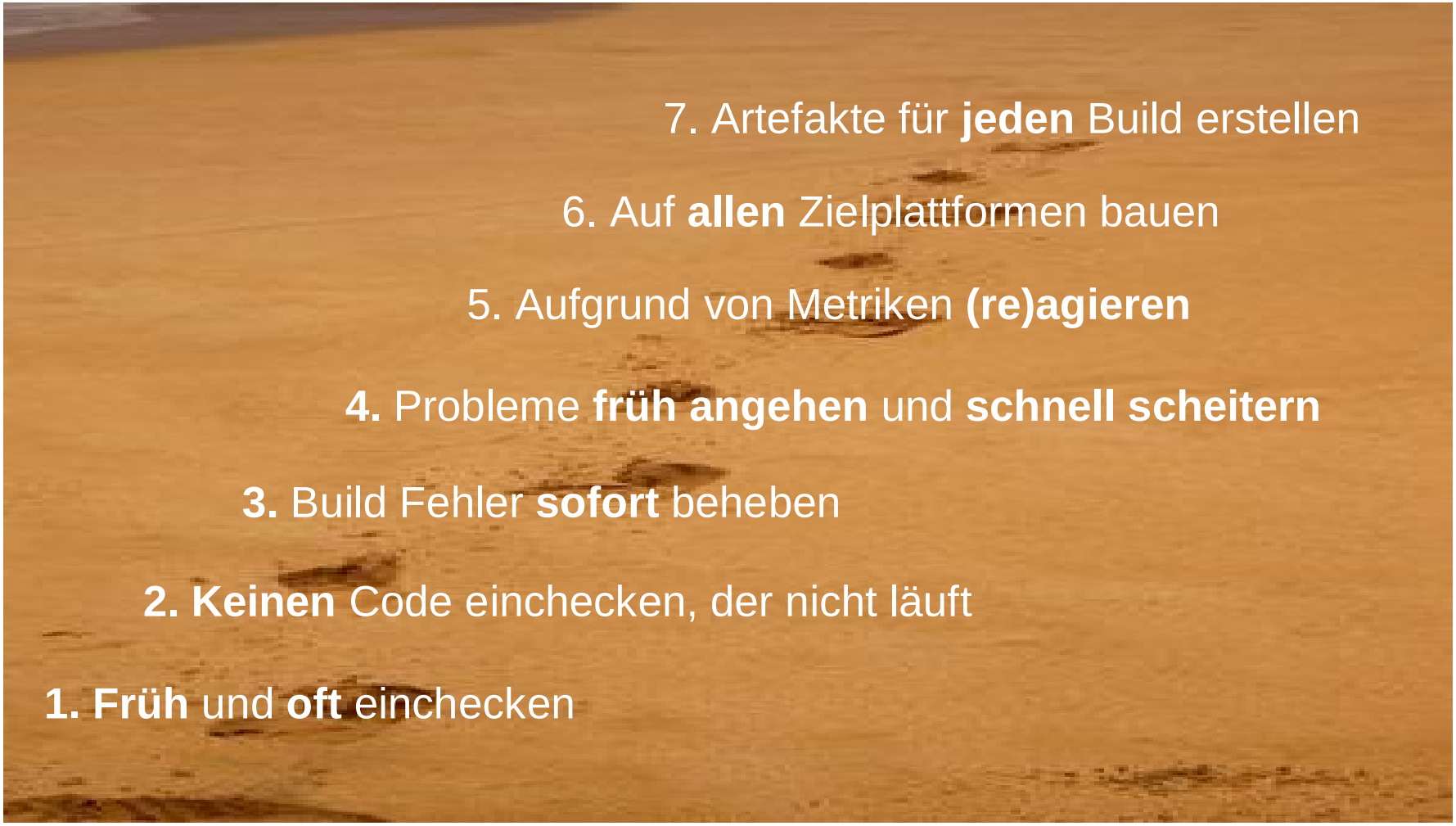
Kontinuierliche Integration 7 Schritte



Kontinuierliche Integration 7 Schritte



Kontinuierliche Integration 7 Schritte



Hudson

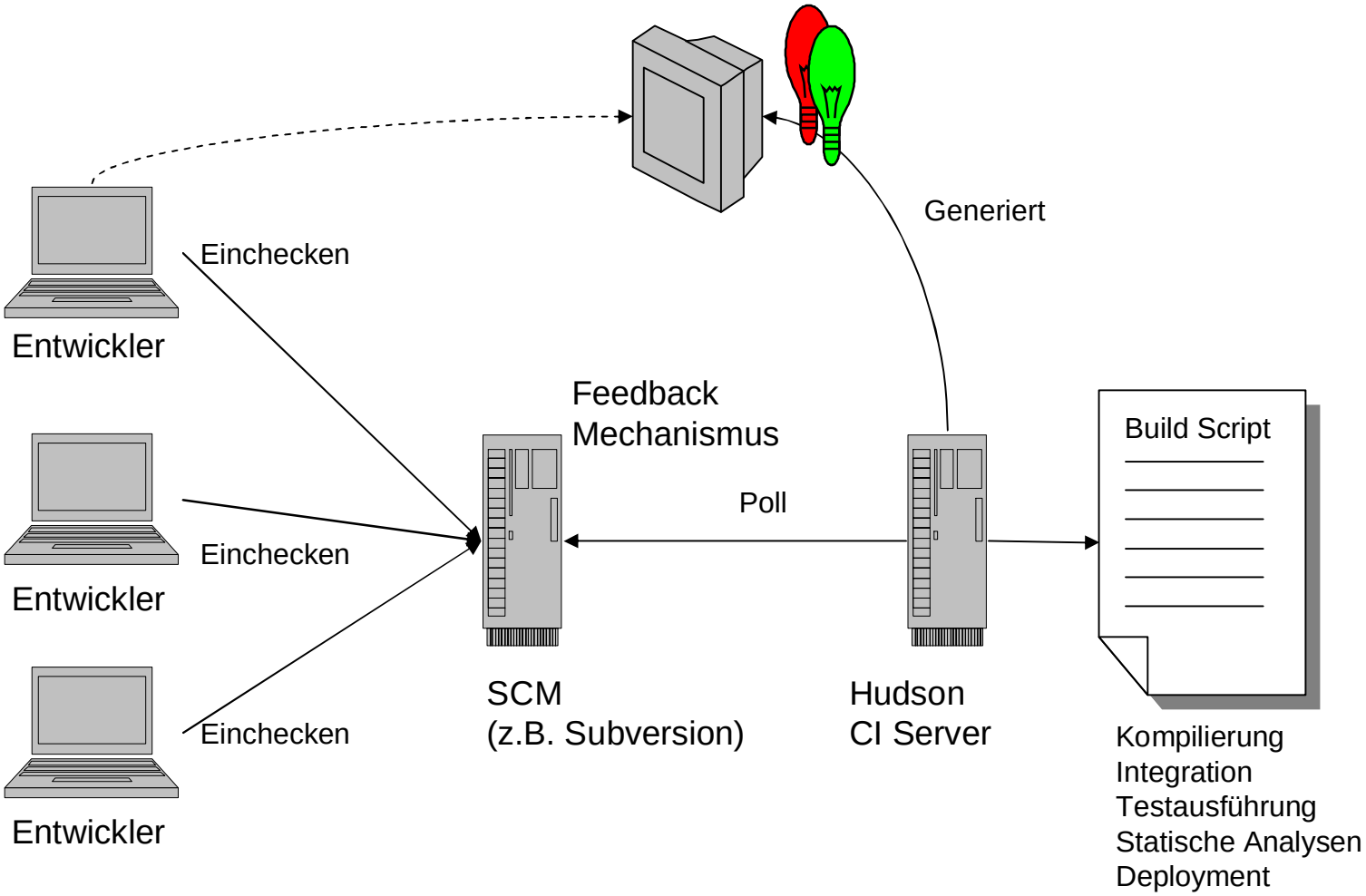

2008 DUKE'S CHOICE AWARDS
Java
WINNER


SOAWORLD
2008
SOA
READERS'
CHOICE
AWARD
RCA FINALIST
SYS-CON

Hudson
Historie …

- Vorfahren
  - Urvater CruiseControl
- Verwandte
  - Continuum, Bamboo, TeamCity, Luntbuild, AnthillPro
- Vater
  - Kohsuke Kawaguchi
- Geburtsort und Datum
  - Sun Microsystems, seit 2005,
    aber richtig populär seit diesem Jahr
- Geburtsurkunde (Lizenzmodell)
  - Mischung aus MIT und Commons Creative
- Notwendig für die ersten Schritte
  - J2SE 1.5

Hudson
Funktionsprinzip



Hudson
Features

- Einfache Installation und Konfiguration
- Unterstützung verschiedener SCM Systeme
- Permanente Links (z.B. für LATEST builds)
- Verschiedenste Benachrichtigungsmechanismen (z.B. RSS/E-mail/IM Integration
- Tagging von Builds mit besonderer Bedeutung
- JUnit/TestNG Test Reporting
- Fingerprinting
- Umfangreicher Plugin Support

Hudson
Live Demo

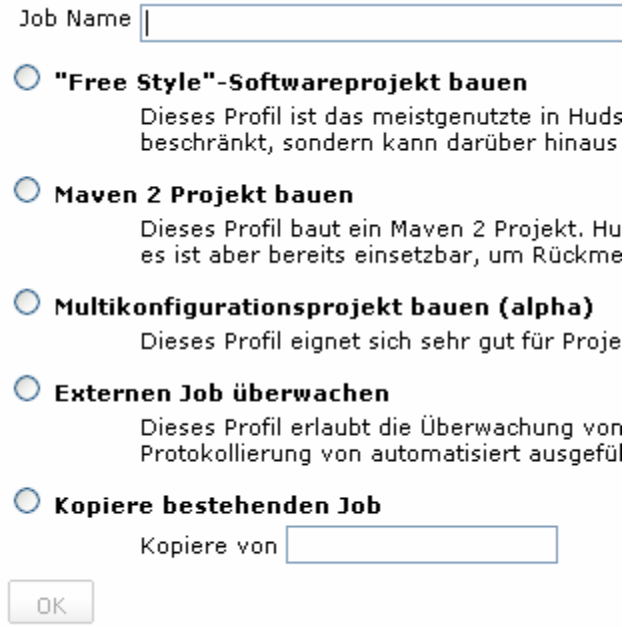
http://localhost:8080

Hudson
Allgemeine Einrichtung

- Hudson verwalten
  - System konfigurieren
  - Plugins verwalten
  - Parallelität
- Betrieb
  - Konfiguration neu laden
  - Herunterfahren vorbereiten
- Diagnose
  - Logging
  - Skript-Konsole
  - System Log

Hudson
Job Einrichtung

- Job Typen
  - Free Style
  - Maven 2
  - Multikonfigurationsprojekt
  - Externen Job überwachen



Hudson
Erweiterungen über interessante Plugins

- Source Code Management
- Build
  - Triggers
  - Tools
  - Wrapper
  - Notifier
  - Reports
- Slave Launcher und Controller
- Build Reports
- External site integrations
- Artifact uploaders
- Und, und , und

Hudson Erfahrungen

- Positiv aufgefallen ☺
  - Berechtigungskonzept
  - Eine Oberfläche für alles
  - Sehr schnell produktiv
  - Interessantes Konzept für Multikonfigurationen
  - Schnelle Bearbeitung von Fehlern und Feature Wünschen
- Wo es noch klemmt ☹
  - Manche Plugins funktionieren nicht
  - Flüchtigkeitsfehler (z.B. in 1.219 bei gleichen Build-Verfahren und Build-Skript aber unterschiedlichen Targets)
  - Sobald Hudson und Plugins verschiedene Versionen der gleichen Library verwenden wollen (z.B. commons-lang)

Ein kleiner Film zum Schluß …

Literatur und Links

- Bücher
  - Continuous Integration, Paul M.Duval
- Links
  - https://hudson.dev.java.net/
  - http://www.martinfowler.com/articles/continuousIntegration.html
  - http://www.slideshare.net/drluckyspin/continuous-integration
  - http://www.slideshare.net/carlo.bonamico/continuous-integration-with-hudson/
  - http://jboss-qa.blogspot.com/2007/10/taking-continuous-integration-to.html
  - http://www.it-agile.de/build-flashfilm.html

Literatur und Links

- Bilder Referenzen (cc creative commons)
  - http://www.flickr.com/photos/carstingaxion/1103931822/
  - http://www.flickr.com/photos/lukepdq/99418297/
  - http://creativecommons.org/licenses/by-nc-nd/2.0/deed.en
  - http://www.flickr.com/photos/paopix/2413495787/
  - http://www.flickr.com/photos/sonosalvo/171714927/
  - http://www.flickr.com/photos/heyjules/2144592427/
  - http://www.flickr.com/photos/destinme/1267500829
  - http://www.flickr.com/photos/dullhunk/359634390/
  - http://www.flickr.com/photos/k9/556002530/
  - http://www.flickr.com/photos/ppdigital/2329376071/
  - http://www.flickr.com/photos/iko/739595/

15.–18. 09. 2008
in Nürnberg

Herbstcampus

Wissenstransfer
par excellence

Vielen Dank!

Martin Heider

infomar software