

Jenkins
in drei
Akten



Dinner
for One



Prolog

Ein Wort, um Jenkins zu
beschreiben?

sin

$\sin(t)$





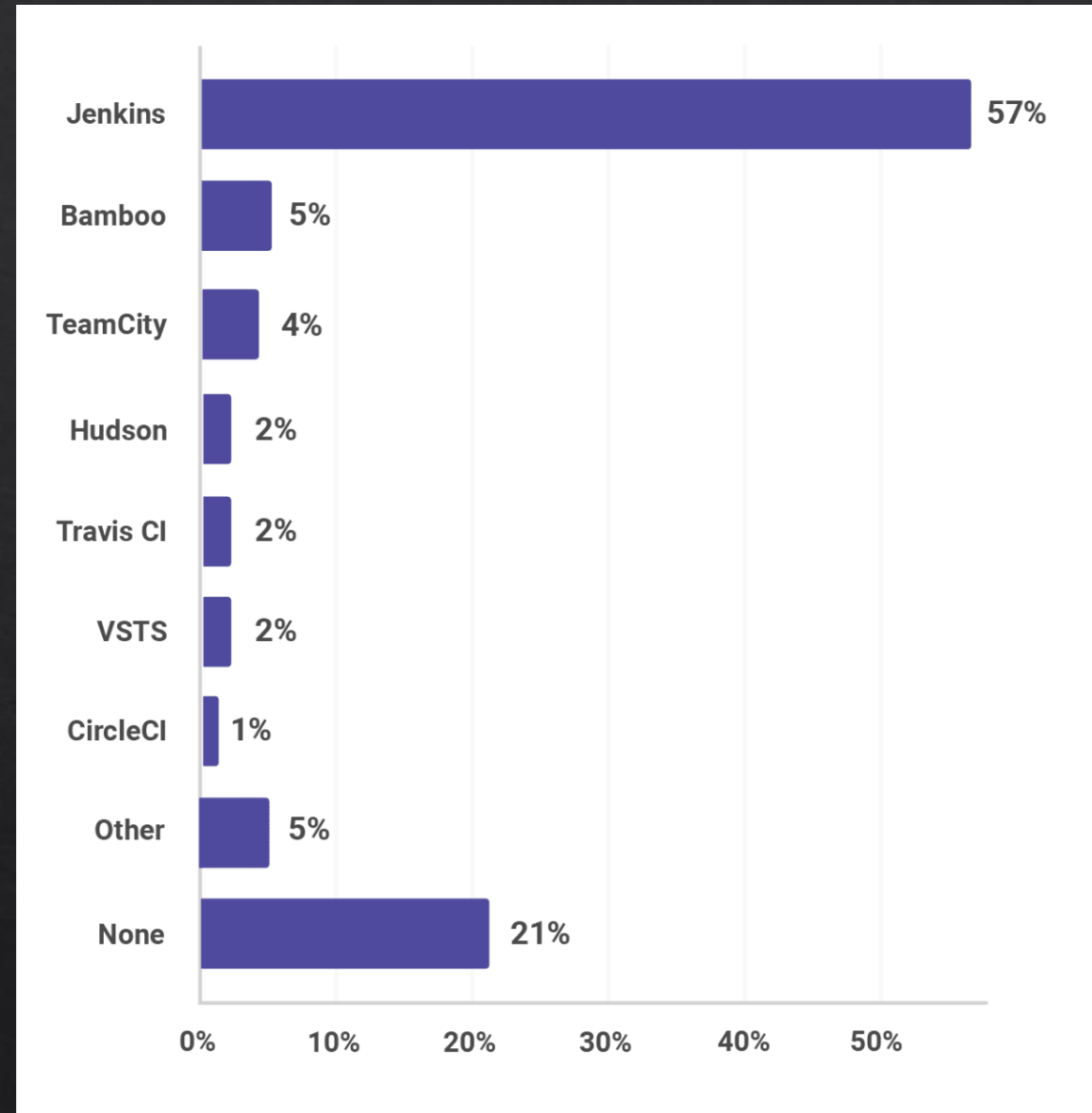
Erster Akt

Die Guten Seiten

Jenkins

- ◇ Continuous Integration Server
- ◇ Open Source
- ◇ Läuft auf den meisten Plattformen (Windows, *nix, ...)
- ◇ Geschrieben in Java
- ◇ Weite Verbreitung und grosse Community
- ◇ Unzählig viele Plugins
- ◇ Groovy DSL für Pipeline as Code

Which CI Server do you use?





Coole Features

Shared Libraries

- ◇ Zentrale Scripte um Pipelines zu vereinfachen oder wiederzuverwenden
- ◇ Bestehende DSL leicht mit eigenen Funktionen erweiterbar
- ◇ Ganze Pipelines und Stages können ausgelagert werden
- ◇ «Jeder gültige Groovy Code»

Unlimited

- ◇ Unbegrenzte Anzahl von Jobs, Slaves und Executors
 - ◇ Lokal
 - ◇ Dedizierte Server
 - ◇ Docker Slaves (auch auf Kubernetes)
- ◇ Zum Vergleich:
 - ◇ Bamboo: unbegrenzte lokale Agents, 1 Remote Agent ab 800\$/Jahr
 - ◇ TeamCity: keine lokalen Agents, 100 Jobs & 3 Remote Agents kostenlos, jeder weitere 300\$/Jahr

Ausserdem ...

- ◇ Plugins, Plugins, Plugins
 - ◇ Für praktisch jedes Problem gibt es ein Plugin (Problem → Plugin)
- ◇ Blue Ocean
 - ◇ Optisch ansprechende Pipeline Visualisierung
- ◇ Pipeline DSL
 - ◇ Zwei Stile: Scripted und Deklarativ (weniger «technisch»)

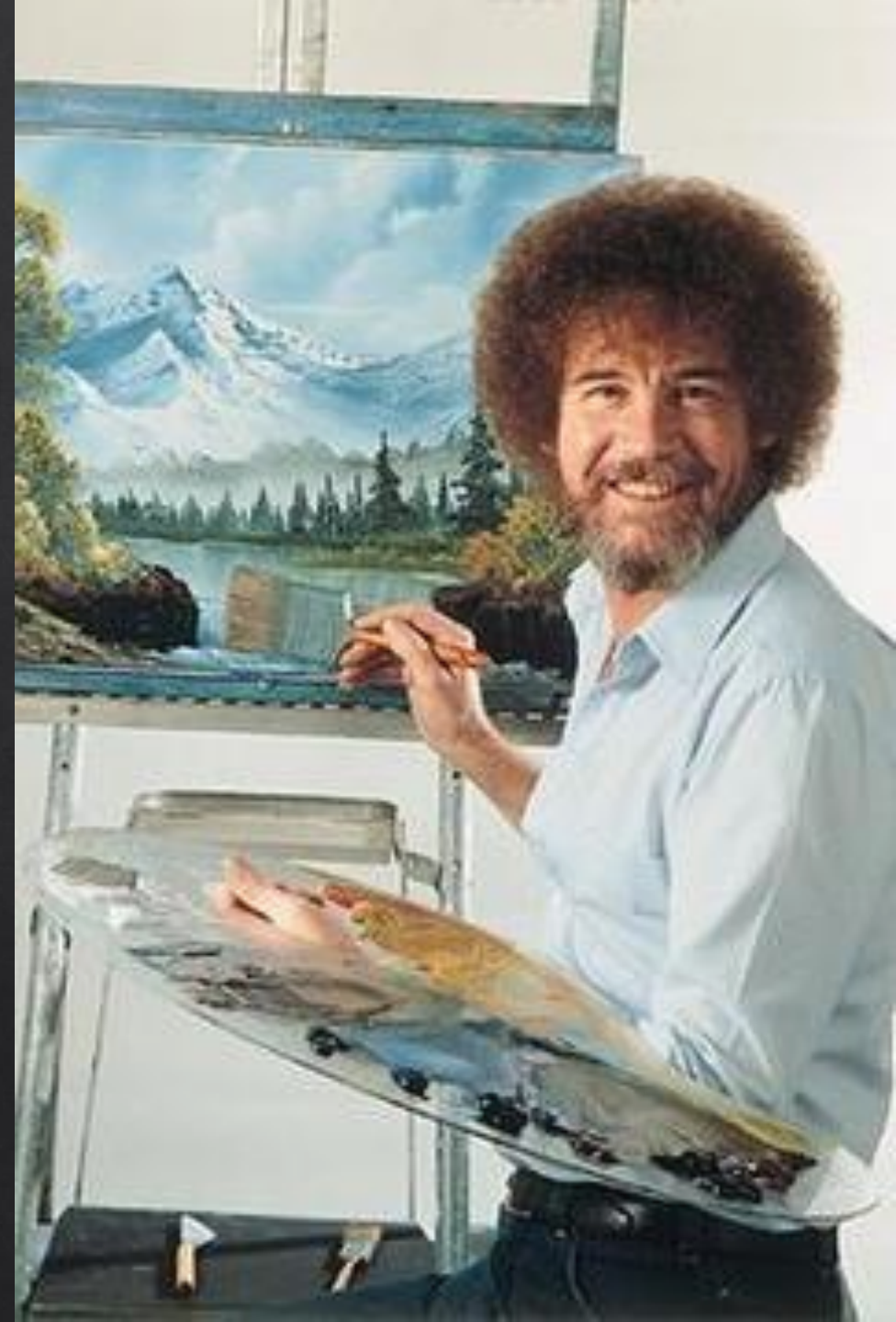


Zweiter Akt

Wie man betrunken wird

It's so easy...

- ◆ Ein Cron Trigger für alles – mit UI
- ◆ Orchestrator für Batch Jobs
- ◆ DevOps Koordinator
- ◆ Garbage Collector
- ◆ Produktionskritische Aufgaben



Der Dienst,
der das gesamte
Projekt &
die IT
zusammenhält

KLEBSTOFF SELBST GEMACHT



THE RIGHT STUFF

How the future began.



Der Stoff aus dem
die Helden sind

A BROSNET CLASSICS DOWN WINNER Production of a PHILIP SKLYMAN from "THE RIGHT STUFF"
CHARLES FRANK SCOTT GLENN ED HARRIS LANCE BENSCHEN SCOTT BULLIN DENNIS QUINN
SAM SHIPARD TERRY WARD KIM STANLEY BARBARA HERSHEN VERONICA CARWRIGHT
EMMA ARLED Music by BILL CONVE Director of Photography CALVIN FRECHMAN Based on the book by TIM WOLFE
Directed by JOHN WISLER and GREGORY CHADOFF Screenplay by James and James by HILF KAUSMAN

PG-13



Helden?

HULK
THE INCREDIBLE



Mit Grosser Verantwortung kommt grosse ...
Machtlosigkeit?

The Joy of Bugs



Offene Issues im Vergleich

Jenkins ~	TeamCity	Bamboo
800 (Core + Workflow) – 1500 (+ Pipeline) – 6000 (Total)	~750	~<10

Mein Name ist Bugs, Jenkins Bugs

- ◆ Gelöst durch zirkuläre Duplikationsreferenz
- ◆ Seit Ewigkeiten offen
- ◆ Bug fixed, aber eigentlich "postponed"
- ◆ Seit letzte Woche offen
- ◆ Beschreiben genau dein Problem, aber nur der für dich nicht relevante Teil wurde behoben



This Photo by Unknown Author is licensed under [CC BY-NC-ND](#)

Ein paar Beispiele

- ◇ Parameters null on first build
 - ◇ <https://issues.jenkins-ci.org/browse/JENKINS-40574>
- ◇ Duplicated by Declarative pipeline parameters default values are not loaded at first job run
 - ◇ <https://issues.jenkins-ci.org/browse/JENKINS-42922>
- ◇ Duplicates Build Parameters on first build
 - ◇ <https://issues.jenkins-ci.org/browse/JENKINS-41929>
- ◇ Field access in child classes (shared libs)
 - ◇ <https://issues.jenkins-ci.org/browse/JENKINS-47355>
- ◇ Groovy Traits are not supported
 - ◇ <https://issues.jenkins-ci.org/browse/JENKINS-46145>
- ◇ Active Choice and Shared Library
 - ◇ <https://issues.jenkins-ci.org/browse/JENKINS-46394>
- ◇ Global Pipeline Libraries triggers the 'poll SCM' of jobs
 - ◇ <https://issues.jenkins-ci.org/browse/JENKINS-39615>

Solution Obfuscation by Duplication

Details

Type: **Bug** Status: **RESOLVED** (View Workflow)
Priority: **Critical** Resolution: Duplicate
Component/s: pipeline

Created: 2016-12-20 16:06
Updated: 2017-09-01 19:38
Resolved: 2017-09-01 19:38

duplicates

JENKINS-41929 Offer "Build with Parameters" on first build when declarative Jenkinsfile found **OPEN**

is duplicated by

JENKINS-42922 Declarative pipeline parameters default values are not loaded at first job run **Duplicate**

JENKINS-41929 Offer "Build with Parameters" on first build when declarative Jenkinsfile found **OPEN**






Type: **Bug** Status: **RESOLVED** (View Workflow)
Priority: **Minor** Resolution: Duplicate
Component/s: workflow-multibranch-plugin

duplicates

JENKINS-40574 Pipeline build parameters are null in first build **RESOLVED**



Critical Bug fixed by Not Implemented Feature

duplicates	
 JENKINS-40574 Pipeline build parameters are null in first build	Duplicate
is duplicated by	
 JENKINS-40574 Pipeline build parameters are null in first build	Duplicate
 JENKINS-46594 Pipelines with parameters always fail to pick up parameters on first run	Duplicate
 JENKINS-40241 Determine parameters before running job	Duplicate
 JENKINS-45454 Pipeline from SCM jobs should have a fetch option	Duplicate

Type:	 New Feature	Status:	OPEN (View Workflow)
Priority:	 Major	Resolution:	Unresolved
Component/s:	pipeline-model-definition-plugin		

Created: 2017-02-10 15:24



Wa Tse Fak*



* Chinesischer Chef Chili Schlucker**

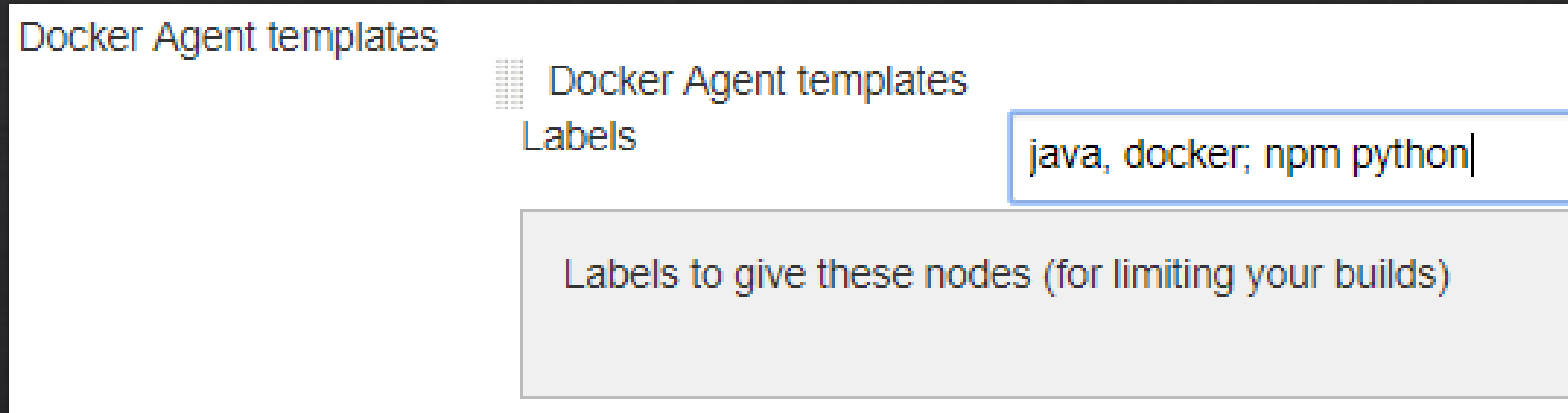
** Fake News

*Schrei nach Liebe

User Interface & User Experience

- ◇ ... als hätte es ein betrunkenener Säugling entwickelt
- ◇ Unübersichtliche Administration
- ◇ Navigation Unübersichtlich
- ◇ NaNy NaNd
- ◇ Broken Layouts
- ◇ Blau statt grün für «Success» !?
- ◇ Unklare Dokumentation der Elemente

Documentation like bad JavaDocs



ProTip: It's a space separated list

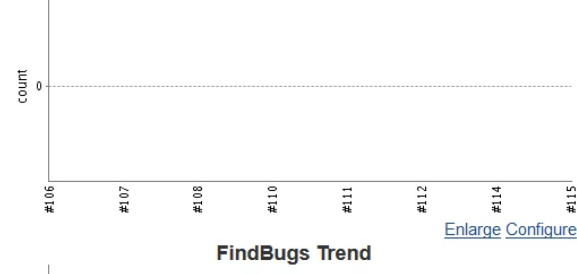
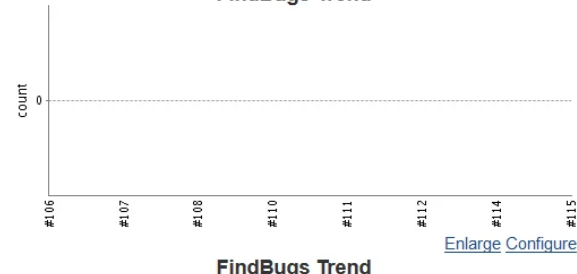
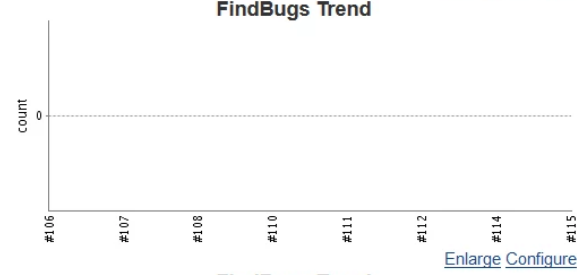
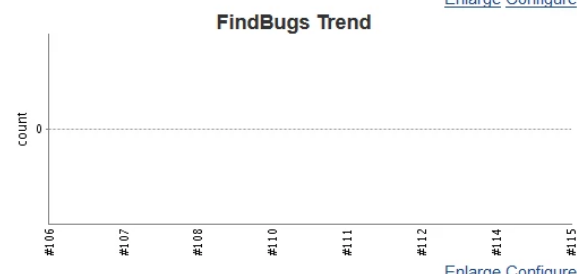
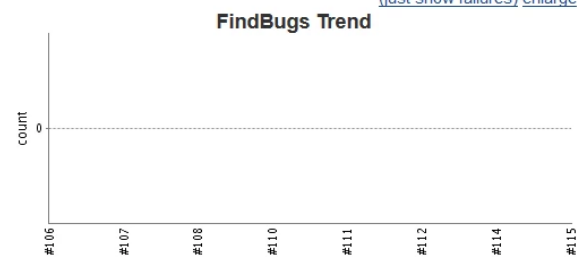


Advanced oder Deprecated?

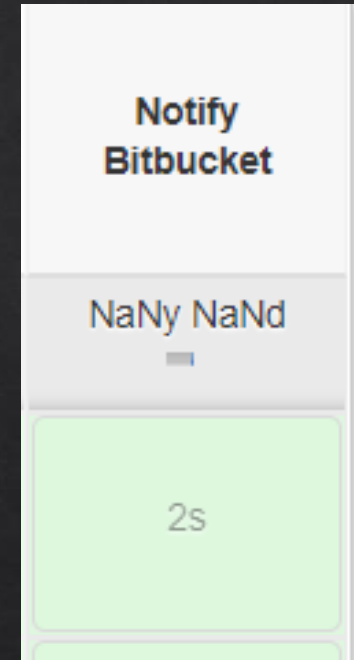
```
writeMavenPom: Write a maven project file.  
writeYaml: Write a yaml from an object.  
ws: Allocate workspace  
zip: Create Zip file  
— Advanced/Deprecated —  
ArtifactoryGradleBuild: run Artifactory gradle  
MavenDescriptorStep: Get Artifactory Maven descriptor  
addInteractivePromotion: Add Interactive promotion  
archive: Archive artifacts  
artifactoryDistributeBuild: Distribute build  
artifactoryDownload: Download artifacts  
artifactoryMavenBuild: run Artifactory maven  
artifactoryPromoteBuild: Promote build  
artifactoryUpload: Upload artifacts  
catchError: Catch error and set build result  
collectEnv: Collect environment variables and system properties  
conanAddRemote: Add new repo to Conan config  
conanAddUser: Add new user to Conan config  
deployArtifacts: Deploy artifacts  
dockerFingerprintFrom: Record trace of a Docker image used in FROM  
dockerFingerprintRun: Record trace of a Docker image run in a container  
dockerPullStep: Artifactory docker pull  
dockerPushStep: Artifactory docker push
```



Geht kaum trendiger



NaNy NaNd?



System Administration

The screenshot displays the Jenkins administration interface. At the top, the Jenkins logo and navigation links are visible. The main content area is divided into several sections:

- Left Sidebar:** Contains navigation links such as 'Element anlegen', 'Benutzer', 'Build-Verlauf', and 'Meine Ansichten'. It also features a 'Build-Warteschlange' (Build Queue) section showing 'Keine Builds geplant' and a 'Build-Prozessor-Status' (Build Executor Status) section showing four 'master' nodes in a 'Ruhend' (Idle) state.
- Main Content Area:**
 - Jenkins Home-Verzeichnis:** A text area for the welcome message, currently containing '/data/jenkins'.
 - Maven-Projekt-Konfiguration:** Includes settings for 'Globale MAVEN_OPTS', 'Lokales Maven-Repository' (set to 'Default (~/.m2/repository)'), 'Anzahl der Build-Prozessoren' (4), 'Auslastung' (set to 'Diesen Knoten so viel wie möglich verwenden'), 'Ruheperiode' (0), and 'Anzahl der SCM-Checkout Wiederholungen' (2).
 - Globale Eigenschaften:** Includes checkboxes for 'Namenskonventionen für Jobnamen erzwingen', 'Umgebungsvariablen', and 'Verzeichnisse von Hilfsprogrammen anpassen'.
 - SonarQube servers:** Features a checkbox for 'Enable injection of SonarQube server configuration as build environment variables'. Below this, there are input fields for 'Name', 'Server URL', and 'Server authentication token'. A note indicates that if checked, administrators can inject SonarQube configuration as environment variables. A default URL of 'http://localhost:9000' is shown.

- Element anlegen
- Benutzer
- Build-Verlauf
- Ansicht bearbeiten
- Projektbeziehungen
- Fingerabdruck überprüfen
- Jenkins verwalten
- Meine Ansichten
- Scriptler
- Zugangsdaten
- Ansicht anlegen

Build-Warteschlange →

Keine Builds geplant

Build-Prozessor-Status →

- master
- 1 Ruhend
- 2 Ruhend
- 3 Ruhend
- 4 Ruhend

Jenkins Home-Verzeichnis /data/jenkins
 Willkommensmeldung

[Nur Text] [Vorschau](#)

Maven-Projekt-Konfiguration

Globale MAVEN_OPTS

Lokales Maven-Repository

Anzahl der Build-Prozessoren

Labels

Auslastung

Ruheperiode

Anzahl der SCM-Checkout Wiederholungen

Standardansicht

Namenskonventionen für Jobnamen erzwingen

Globale Eigenschaften

- Umgebungsvariablen
- Verzeichnisse von Hilfsprogrammen anpassen

SonarQube servers

Environment variables Enable injection of SonarQube server configuration as build environment variables
If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

SonarQube Installationen

Name	SonarQube
Server URL	.. /
Server authentication token	Voreinstellung ist http://localhost:9000

SonarQube authentication token. Mandatory when anonymous access is disabled.

Liste der SonarQube Installationen

Kerberos Single Sign-On

Enable Single Sign-On plugin

Redirect if domain is not present in URL

Domain to redirect to

Kerberos properties

Service Account

Password

Location of krb5.conf

Location of login.conf

Login Server Module

Login Client Module

Allow anonymous access

Allow Localhost

Allow Basic

Allow Delegation

Allow Unsecure Basic

Prompt NTLM

SCM Sync configuration

SCM None Subversion Git

Never bother me with commit messages

Display SCM Sync Status

Commit message pattern

Manual synchronization includes

Reload config from SCM

Metrics

Access keys

Pipeline Speed/Durability Settings

Pipeline Default Speed/Durability Level

Ivy Project Configuration

Global ANT_OPTS

Usage Statistics

Helfen Sie Jenkins zu verbessern, indem Sie anonyme Nutzungsstatistiken und Absturzprotokolle an das Jenkins-Projekt senden.

Bitbucket Endpoints

Bitbucket Cloud

Manage hooks

Credentials

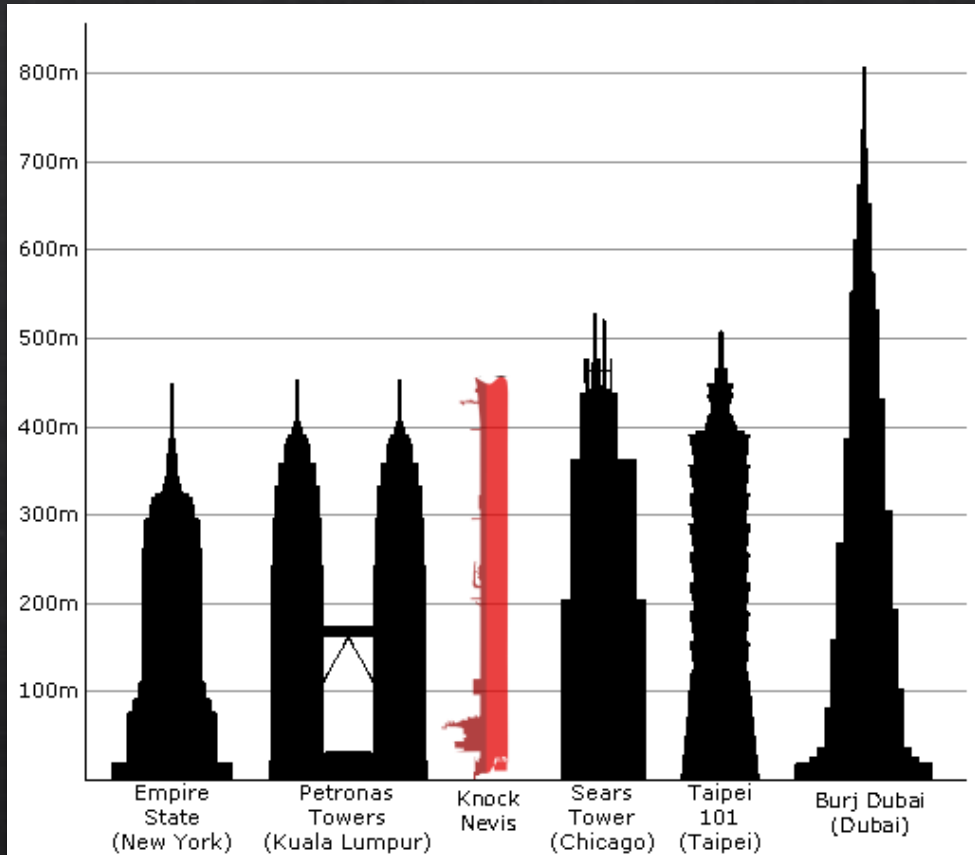


Table with multiple columns and rows, likely a data table or spreadsheet. The content is mostly illegible due to the small font size and high resolution. It appears to contain various data points and possibly a list of items.

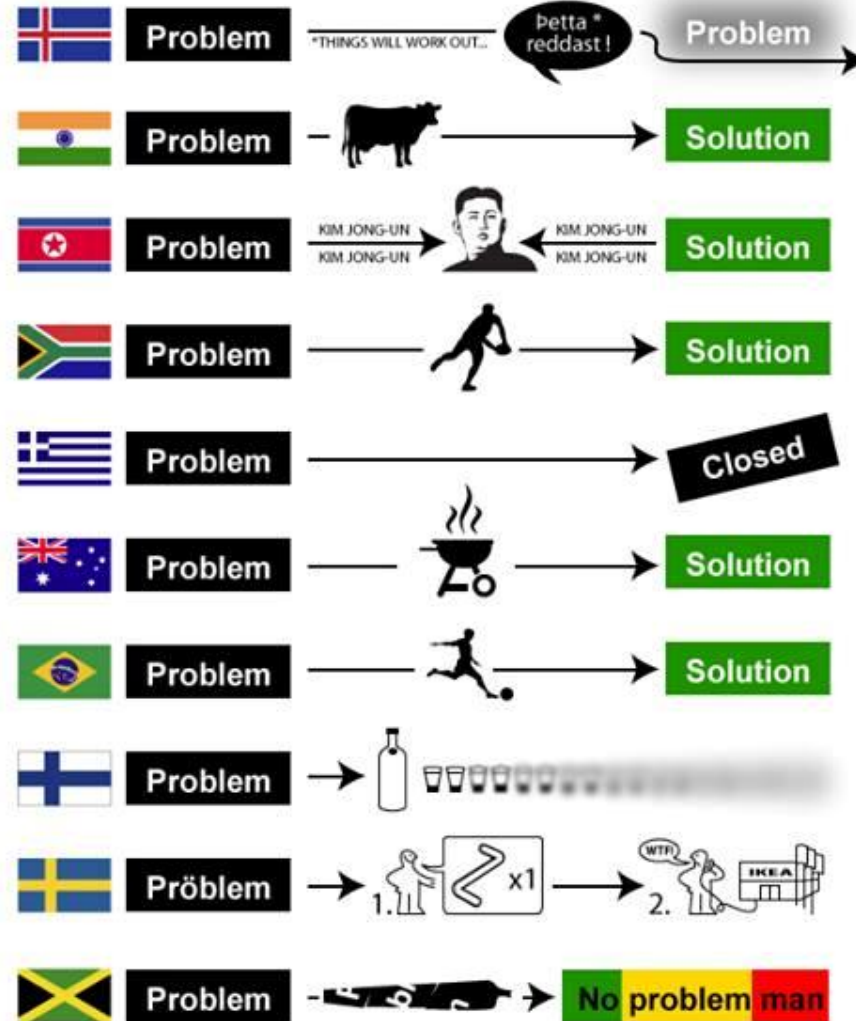
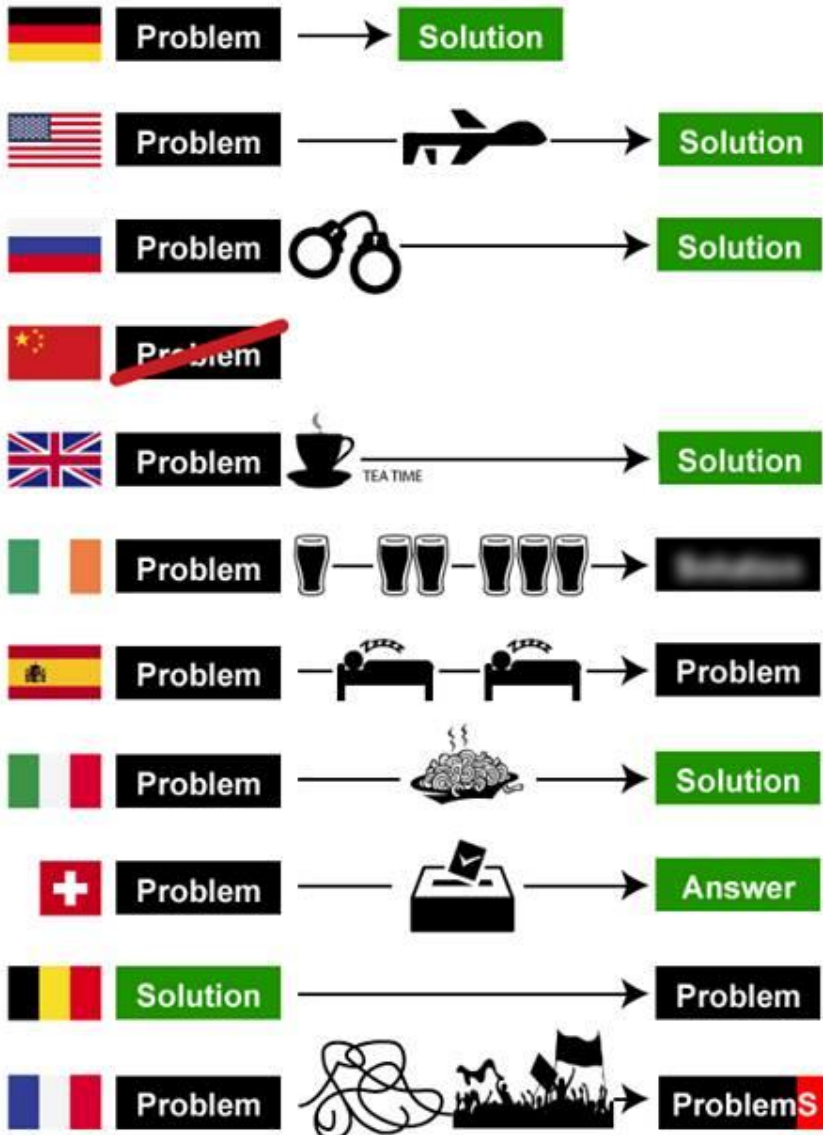


Problem → Solution

INTERNATIONAL GUIDELINES FOR PROBLEM SOLVING

Send me your comments or ideas:
facebook/jeromevadon or @jeromevadon

3.0
UPDATED
by popular request



Send me your comments or ideas facebook/jeromevadon or @jeromevadon

Problem → Plugin



- Pipeline API Plugin
- Pipeline AWS Plugin
- Pipeline Bamboo Plugin
- Pipeline Basic Steps Plugin
- Pipeline Build Step Plugin
- Pipeline Classpath Step Plugin
- Pipeline Dependency Walker...
- Pipeline Editor Plugin
- Pipeline GitHub Library Plugin
- Pipeline Githubnotify Step Pl...
- Pipeline Github Plugin
- Pipeline Gra...
- Pipeline Gro...
- Pipeline Inp...
- Pipeline Job...
- Pipeline Ma...
- Pipeline Mil...
- Pipeline Mo...
- Pipeline Mu...
- Pipeline Mu...
- Pipeline No...
- Pipeline NP...
- Pipeline PLU...

Dingding[钉钉]
钉钉Jenkins通知器

Girls Plugin

Erstellt von Alan Harder, zuletzt geändert von Mid...

This plugin violates community guidelines

⚠️ Deprecatd: This plugin has been deprecated.
Violation of community guidelines

See also:

Emotional Jenkins Plugin

Created by masanobuimai, last modified by Nobuhiro Ban on Jul 06, 2013

Plugin Information
View emotional-jenkins-plugin on the plugin site for more information.

This funny plugin changes the expression of Mr. Jenkins in the background when your builds fail.

The plugin is maintained by Masanobu Imai.

Once installed, you need to enable this on per-job basis.

Configuration

To enable this feature for a job, go to the job configura...

Screenshot



Labelled Pipeline Steps
TODO

BruceSchneier Plugin

Created by Michael Rooney, last modified by Baptiste Mathus on Nov 02, 2015

beer
This plugin dispenses

ChuckNorris Plugin

Created by Clifano Subagio, last modified by Baptiste Mathus on Jan 24, 2016

Plugin Information
View ChuckNorris on the plugin site for more information.

Displays a picture of Chuck Norris (instead of Jenkins the butler) and a random Chu...

Download & Installation

You can preferably install this plugin directly from the Plugins Update Center, or if ne...

pry
This plugin is a sample to explain how to write a Jenkins plugin.

3. Run a build. After it's completed, Chuck Norris image (and the random quote)

Screenshots

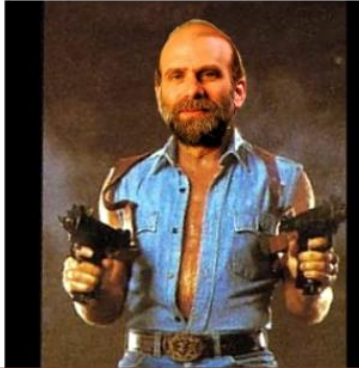
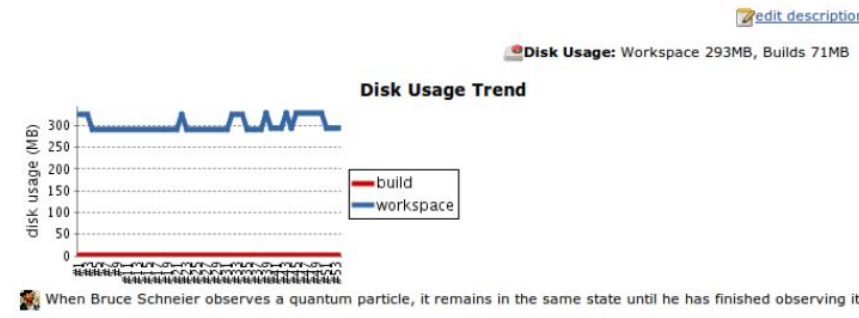
Build #15 (Sep 10, 2007 12:02:44 AM) Started 18 min ago Took 10 sec

- No changes.
- Started by user anonymous
- Chuck Norris's beard can type 140 wpm.

' is available. It may have been removed from distribution.

to help improve this plugin? Click here to learn more!

of Jenkins the butler) and a random Bruce Schneier fact on each build page.



360 FireLine

火线是一款支持Android和Java的静态代码扫描工具，支持安全和资源泄露规则的检查。最新版本已经兼容阿里巴巴Java代码规范（P3C）。

Plugin → Problem

- ◇ Ständig neue Updates
 - ◇ Updates können
 - ◇ Build Brechen
 - ◇ Build verändern
 - ◇ Zu anderen Plugins Inkompatibilitäten hervorbringen
- ◇ Qualität der Plugins sehr wechselhaft
 - ◇ Zu viele werden nur dürftig gewartet
- ◇ Dokumentation teilweise sehr dürftig

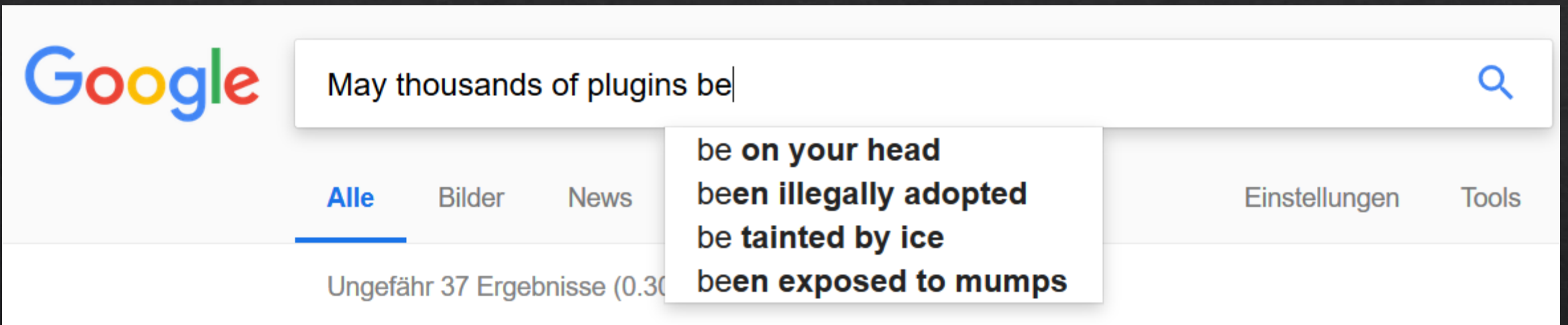
Zu Viele Plugins



This block contains a large, white, rectangular area that is heavily distorted and tilted. It appears to be a screenshot of a list of plugin names or code, but the text is completely illegible due to the extreme perspective and compression. The list is organized into columns, suggesting a structured data set.



Fluchen für Fortgeschrittene



The image shows a screenshot of a Google search page. The Google logo is on the left. The search bar contains the text "May thousands of plugins be". Below the search bar, there are tabs for "Alle", "Bilder", and "News", with "Alle" selected. To the right of the search bar are "Einstellungen" and "Tools". A dropdown menu is open below the search bar, displaying four suggestions: "be on your head", "been illegally adopted", "be tainted by ice", and "been exposed to mumps". Below the search bar, it says "Ungefähr 37 Ergebnisse (0.30".

Google

May thousands of plugins be

Alle Bilder News

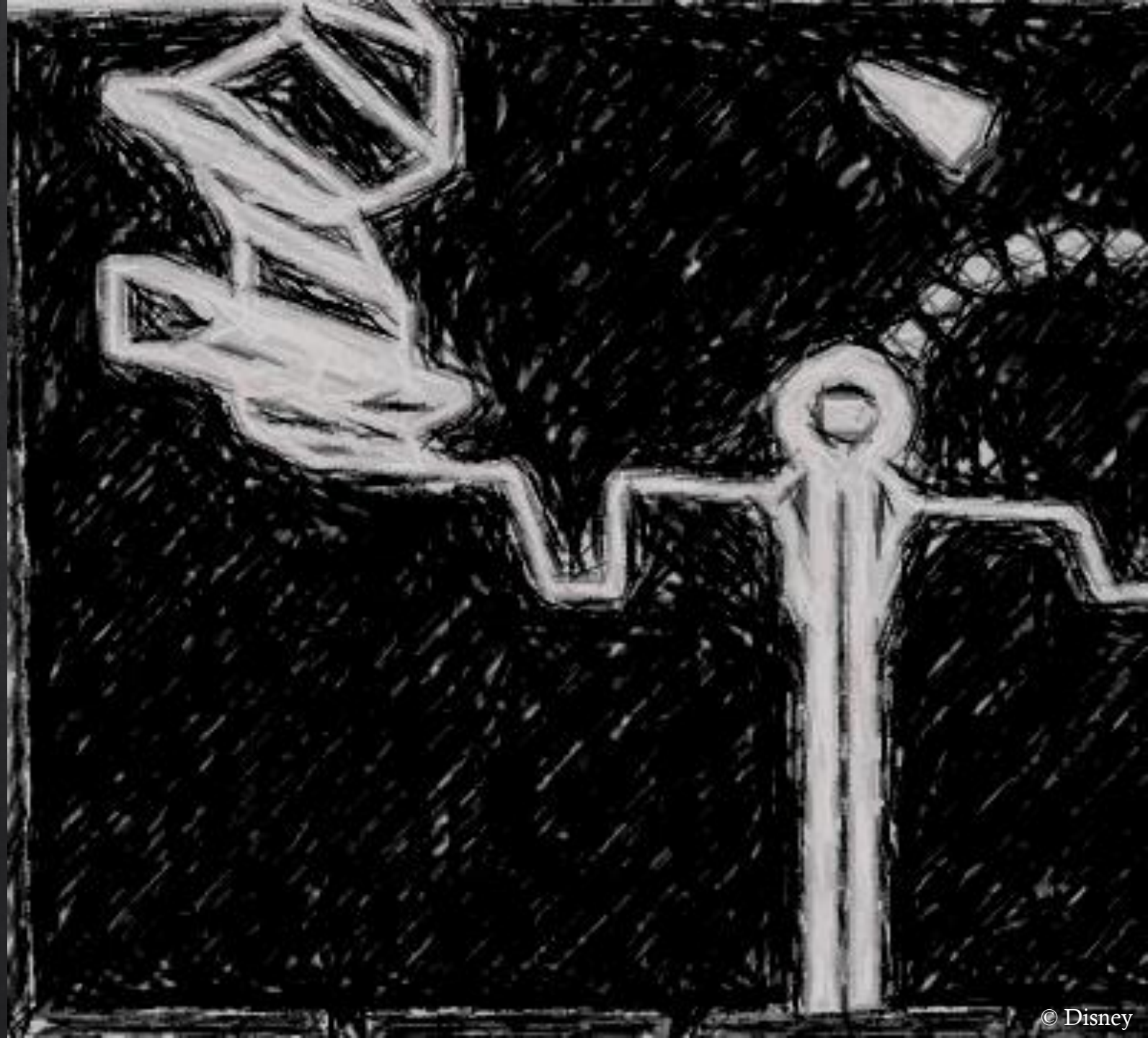
Einstellungen Tools

Ungefähr 37 Ergebnisse (0.30

be on your head
been illegally adopted
be tainted by ice
been exposed to mumps

Stabilität

- ◇ Isolation Build Prozesse
- ◇ Core Dumps in Jobs
- ◇ Out-of-Memory
- ◇ Nur begrenzt Möglichkeit Speicher zu limitieren
- ◇ Plugins



Pipeline Entwicklung

- ◇ Sehr zeitaufwändig
 - ◇ Viel Trial&Error
 - ◇ Kein Lokaler Test möglich
 - ◇ Feedbackzyklen bei kleinen Änderungen von >5min nicht ungewöhnlich
 - ◇ Verhältnis Entwicklung:Test einer Pipeline: ca 1:3
- ◇ Dürftige Dokumentation der Plugins → Source Code

Schwaches «Job/Pipeline» Konzept

- ◇ Pipelines
 - ◇ Lose Aneinanderreihung von Jobs/Steps
 - ◇ Keine Feste Struktur
 - ◇ Stages Verändern sich unter Umständen bei Ausführung (z.B. Abbruch)
 - ◇ Pipelines sind keine 1st-Class Citizen
 - ◇ Manuelle Inputs blockieren Executor und Pipeline oder Timeouts
- ◇ Jobs
 - ◇ Können einfach alles
 - ◇ Werden einfach für alles misbraucht

Pipeline : grundlegend kaputt

- ◇ Muss ausgeführt werden, dass Jenkins sie interpretieren kann
- ◇ Pipeline Konfiguration vs. Job Konfiguration
 - ◇ Job Konfig – sofort sichtbar, wird aber von Pipeline überschrieben
 - ◇ Pipeline Konfig – erst nach Ausführung sichtbar
- ◇ Zwei zueinander teilweise inkompatible Dialekte
 - ◇ Scripted & Deklarativ – wozu eigentlich?
 - ◇ Beispiele und Dokumentation verwirrend
 - ◇ Und: es gibt noch eine Job DSL

INKONSISTENZEN

ÜBERALL INKONSISTENZEN

Inkonsistenzen

ActiveChoice Plugin

```
return[  
  'host1' : 'ENV1',  
  'host2' : 'ENV2',  
]
```

Build with Parameters? → Uses Keys
Scheduled Build? → Uses Values



Spot the difference

Git Parameter

Name

Description

Parameter Type

Branch

Branch Filter

Tag Filter

Sort Mode

RegEx

Wildcard



CPS Groovy != Groovy

- ◇ “Pipeline should run any valid groovy code”
 - ◇ Enums are not auto-converted `.toString()` → messes up all string interpolation, sometime silently
 - ◇ no Traits (for classes in Shared Library)
 - ◇ `new File()` does not work
 - ◇ Primitive Typen funktionieren nicht
- ◇ CPS = Continuation Passing Style = Continuous Pain Supply

Code Quiz 1

- ◆ Gegeben in Shared Lib:
- ◆ Frage: Welcher Branch wird durchlaufen, wenn ...
 - ◆ `param.DryRun == true`
 - ◆ `param.DryRun == false`

```
class MyClass {
    void myMethod(script) {
        if (isDryRun(script)) {
            //IF-TRUE: skip step
        } else {
            //IF-FALSE execute step
        }
    }

    boolean isDryRun(script) {
        // what is returned here?
        return script.dryRun.isEnabled()
    }
}

//dryRun.groovy
boolean isEnabled(){
    def value = params.DryRun == null
                                     ? true
                                     : params.DryRun

    return value as boolean
}
```

Lösung:

- ◆ Immer: IF-TRUE
- ◆ Fix: use Boolean Object type!



Code Quiz 2

◆ Gegeben:

```
class MyClass {  
  
    void myMethod(script){  
  
        String className = getClass().getSimpleName()  
        script.println className //line1  
  
        String thisClassName = this.getClass().getSimpleName()  
        script.println thisClassName //line2  
    }  
}
```

◆ Frage: Was ist der output von

◆ line1 ?

◆ line2 ?

• Lösung: "CpsClosure2"

• Lösung: "MyClass"



Code Quiz 3

◆ Gegeben:

```
sh "ls | grep myfile"
```

◆ Frage: Was macht Jenkins?

1. ls
2. grep myfile



Code Quiz 4

Im Jenkinsfile folgender Aufruf

```
new MyClass().myMethod(this)
```

Frage:

Was ist das Ergebnis?

```
class MyClass extends BaseClass {  
    ...  
}  
  
abstract class BaseClass {  
  
    void myMethod(script){  
  
        String aString = ...  
        File aFile = ...  
  
        otherMethod(aString, aFile)  
  
    }  
  
    private void otherMethod(arg1, arg2){  
        //...  
    }  
}
```

Ergebnis

```
hudson.remoting.ProxyException: groovy.lang.MissingMethodException:
```

```
No signature of method: example.MyClass.otherMethod() is applicable for argument  
types:
```

```
(java.lang.String, java.io.File) values: [MyClass, /some/path/to/file]
```



Lösung

- ◆ Private Keyword entfernen

```
class MyClass extends BaseClass {
    ...
}
abstract class BaseClass {

    void myMethod(script){

        String aString = ...
        File aFile = ...

        otherMethod(aString, aFile)

    }

    private void otherMethod(arg1, arg2){
        //...
    }
}
```



Dritter Akt

Die Kunst nüchtern zu bleiben



This Photo by Unknown Author is licensed under [CC BY-SA](#)

Open Source ist kein Freibier

Keine Technologiewahl ohne Vollkostenrechnung

Verfügbarkeit von

- Know-How
(Dokumentation, Community, Source Code)
- Skills auf dem Arbeitsmarkt

Kosten

- Lizenzen
- Support
- Entwicklung
- Betrieb

Faktoren bei Technologiewahl

Zeit

- Entwicklung
- Betrieb
- Know-How Aufbau
- Troubleshooting

Funktionalität

- Basismerkmale
- Leistungsmerkmale
- Begeisterungsmerkmale

Open Source benötigt

- ◇ Contribution
 - ◇ Bug Reports & Reproducer
 - ◇ Code (Bugfixes & Features)
 - ◇ Dokumentationen & Beispiele
- ◇ Zahlende Kunden
 - ◇ Support Subscriptions (z.B. CloudBees)
 - ◇ Kommerzielle Editionen (z.B. Cloud Bees)
- ◇ Zeit & Geduld
 - ◇ Für Contribution
 - ◇ Workarounds zu finden
 - ◇ Maintenance

We choose to use Jenkins!

“We choose to use Jenkins in this decade and do the other things, not because they are easy, but because they are hard; because that goal will serve to organize and measure the best of our energies and skills, because that challenge is one that we are willing to accept, one we are unwilling to postpone, and one we intend to win, and the others, too”

John F. Kennedy



One Role to Tool them all

- ◇ Single-Role-Servers
 - ◇ Jenkins Server nur für eine Aufgabe benutzen
- ◇ Aufgabenbereich möglichst auf CI begrenzen
 - ◇ dafür wurde Jenkins entwickelt !
- ◇ Andere Aufgaben (Batch, Cron, DevOps)
 - ◇ mit anderen Tools abdecken
 - ◇ Wenn nicht möglich: Instanzen für Produktive Aufgaben trennen
- ◇ Bei grösseren CI/CD Umgebungen:
 - ◇ Staging Umgebung aufbauen um Plugins & Pipelines Testen zu können

Setup

- ◇ Anzahl Plugins auf das Nötigste reduzieren
- ◇ Plugins vorher testen / Rollback einplanen
- ◇ Server und Executors trennen
 - ◇ Separate Maschinen
 - ◇ Docker
- ◇ Kontinuierlich Updaten, Updaten, Updaten

Entwicklung von Pipeline & Jobs

- ◇ Für einen Stil entscheiden
 - ◇ Alles in Groovy ODER Alles in Bash & Groovy Pipelines nur als Orchestrator
 - ◇ Jenkinsfiles ODER Manuelle Build-Konfigurationen
 - ◇ Deklarative ODER Scripted Pipelines?
- ◇ Shared Library verwenden
 - ◇ Parametrierbare Pipeline Templates
 - ◇ Custom Stages
 - ◇ DSL Erweiterungen
 - ◇ Shared Libs versionieren (Tags oder Revision)
- ◇ Keep as Simple as Possible and Readable
- ◇ Genug Zeit für Entwicklung einplanen

Pipelines sind Software

- ◇ Think First!
- ◇ Software benötigt klare Anforderungen
 - ◇ Was muss die Pipeline leisten? Was sind die Ziele? Was der Kontext?
 - ◇ Prozess designen & visualisieren (Whiteboard)
- ◇ Dann
 - ◇ Pipeline implementieren
 - ◇ Pipeline testen → Pipeline Unit kann helfen
 - ◇ Pipeline fortlaufend optimieren



Danke!

Fragen?

Copyright

- ◇ Jedes verwendete Bild ohne gesonderten Urhebernachweis, insbesondere aus dem Stück «Dinner for One», vielfach auch im Internet zu finden, dient dem bildungsmässigen Wissenstransfer und nicht kommerziellen Interessen.
- ◇ Eventuelle Urheberrechtsverletzungen sind nicht beabsichtigt.
- ◇ Sie auch auch https://de.wikipedia.org/wiki/Fair_Use