

Jenkins



Ruby

R. Tyler Croy

arrr tyler

# arrr tyler

- Jenkins community for almost 8 years
- Ruby user for 5 years
- Widely regarded as “tall”
- Jenkins Community Evangelist at CloudBees
- Not a pirate
- Seit fast acht Jahren im Jenkins Open-Source-Projekt aktiv
- Seit fünf Jahren Ruby-Benutzer
- Viele schauen zu ihm auf
- Arbeitet bei CloudBees als Jenkins Community Evangelist
- Arrrr, Wahrscheinlich kein Pirat

# Jenkins

- New Item
- People
- Build History
- Edit View
- Project Relationship
- Check File Fingerprint
- Manage Jenkins
- Support
- Credentials
- My Views
- Job Config History
- We Need Beer
- Scriptler

[add description](#)

- All
- All Disabled
- All Failed
- All Unstable
- Infrastructure
- Jenkins core**
- Libraries
- Other Projects
- Ruby

S	W	Name ↓	Last Success	Last Failure	Last Duration	LC
		<a href="#">jenkins_2.0</a>	6 days 0 hr - <a href="#">#10</a>	29 days - <a href="#">#3</a>	58 min	
		<a href="#">jenkins_2.0_pipeline</a>	N/A	N/A	N/A	n/a
		<a href="#">jenkins_its_branch</a>	6 hr 23 min - <a href="#">#202</a>	2 mo 29 days - <a href="#">#189</a>	33 min	
		<a href="#">jenkins_main_maven-3.1.0</a>	2 yr 6 mo - <a href="#">#7</a>	N/A	1 hr 11 min	
		<a href="#">jenkins_main_trunk</a>	2 days 1 hr - <a href="#">#4435</a>	N/A	57 min	
		<a href="#">jenkins_pom</a>	1 yr 4 mo - <a href="#">#264</a>	2 days 18 hr - <a href="#">#337</a>	47 sec	
		<a href="#">jenkins_rc_branch</a>	11 mo - <a href="#">#517</a>	1 yr 8 mo - <a href="#">#424</a>	25 min	
		<a href="#">remoting</a>	2 yr 7 mo - <a href="#">#4</a>	2 yr 7 mo - <a href="#">#3</a>	6 min 19 sec	
		<a href="#">tmp-jenkins-20-jenkinsfile</a>	13 days - <a href="#">#33</a>	13 days - <a href="#">#38</a>	3 min 39 sec	

**Build Queue** -

No builds in the queue.

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

# Jenkins

- Open source automation server
  - Packaged for practically every platform: Mac OS X, Windows, Linux, FreeBSD, Docker
  - System requirements: Java
  - Extensible: 1000+ plugins to integrate with your development stack
- Open-Source Automatisierungserver
  - Pakete für verschiedene Betriebssysteme: Mac OS X, Windows, Linux, FreeBSD, Docker
  - Systemanforderung - Java
  - Erweiterbar: Über 1000 Plugins zur Integration mit praktisch allen Entwicklungswerkzeugen

why

(i hope)

you care

# benefits of CI/CD

- running tests, linting, style checks, etc after every single commit (if possible)
- consistent feedback mechanism to developers and others via email, IRC, HipChat
- deployment hub for publishing code to Rubygems, Heroku, etc
- Nach jedem Commit wird die Software automatisch getestet und analysiert (wenn möglich)
- Entwickler erhalten ständig Feedback via Email, IRC, HipChat, usw.
- Zentral gesteuertes Deployment zu Rubygems, Heroku, usw.



words Jenkins uses

# Jenkins' Dictionary

- **Job:** entity representing a series of actions. Generally a Job will map to at least one source repository
- **Build:** a single execution of all the actions defined in a Job
- **Build Trigger:** criteria defined to start a new build (e.g. a new commit in a repository)
- **Build Step:** unit of work defined in a Job (e.g. running a shell script)
- **Post-build Action:** action used to process results once the Build Steps have finished execution (e.g. send a failure email)
- **Job:** repräsentiert eine Sequenz von Schritten. Üblicherweise nutzt ein Job mindestens ein Quellcode-Repository.
- **Build:** aller im Job definierter Schritte
- **Build-Auslöser:** definieren Kriterien, nach denen ein Build gestartet wird (wie z.B. ein neuer Commit im SCM)
- **Build-Schritt:** In einem Job definierte Arbeitseinheit (z.B. Ausführung eines Shell-Scripts)
- **Post-Build-Aktion:** Wird verwendet, um Buildergebnisse nach Fertigstellung aller Build-Schritte zu verarbeiten (z.B. Fehlerbenachrichtigung via Email senden)

let's make a job

first we need a Jenkins



# Jenkins

An extendable open source automation server

Fork me on GitHub

ZING!

POW!



- BLOG
- PARTICIPATE
- USE-CASES
- RESOURCES
- CONDUCT
- PRESS
- DONATE



## Meet Jenkins

Find out what Jenkins is and get started.



## Use Jenkins

See how to get more out of your Jenkins

## What is Jenkins?

In a nutshell, Jenkins is the leading open source automation server. Built on Java, it provides **hundreds** of plugins for building, testing, deploying and automating virtually any project

## Download

- Release
- Long-Term Support Release

### Java Web Archive (.war)

**Older but stable: 1.642.1**

[changelog](#) | [past releases](#) | [RC](#)

[What is LTS?](#)

```
java -jar jenkins.war
```

then we need plugins

Dashboard [Jenki x]

localhost:8080

# Jenkins

search

Jenkins

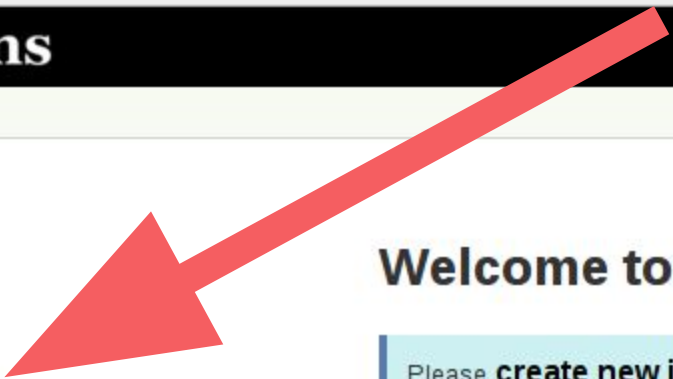
- New Item
- People
- Build History
- Manage Jenkins
- Credentials

## Welcome to Jenkins!

Please **create new jobs** to get started.

**Build Queue**

No builds in the queue.



CLICK!



New Item

People

Build History

**Manage Jenkins**

Credentials

### Build Queue

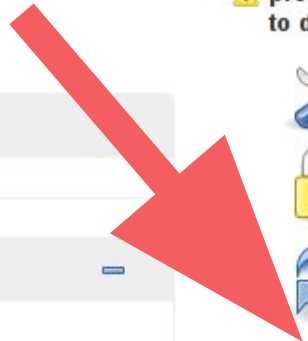
No builds in the queue.

### Build Executor Status

- 1 Idle
- 2 Idle

## Manage Jenkins

# DING!



**New version of Jenkins (1.647) is available for [download \(changelog\)](#). Unsecured Jenkins allows anyone on the network to launch processes on your behalf. Consider at least enabling authentication to discourage misuse.**

[Or Upgrade Automatically](#)

[Setup Security](#) [Dismiss](#)



[Configure System](#)  
Configure global settings and paths.



[Configure Global Security](#)  
Secure Jenkins; define who is allowed to access/use the system.



[Reload Configuration from Disk](#)  
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.



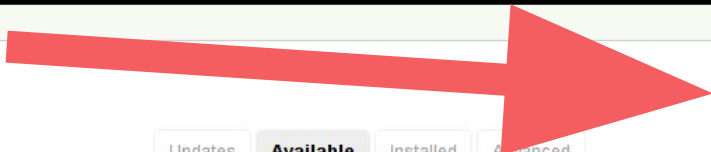
[Manage Plugins](#)  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins. **(updates available)**



[System Information](#)  
Displays various environmental information to assist trouble-shooting.

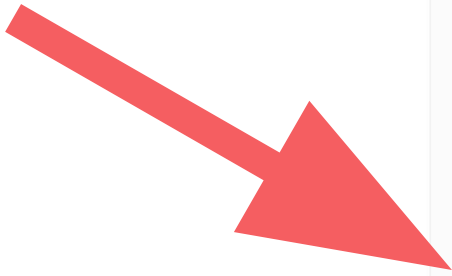


QUERY!



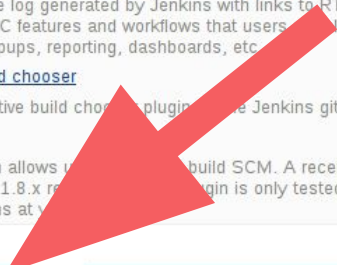
Filter:

MASH!



Install	Name	Version
<input type="checkbox"/>	<a href="#">Git Parameter Plug-In</a> This plugin allows you to choose between Git tags or sha1 of your SCM repository so Git Plugin installed is required.	0.4.0
<input type="checkbox"/>	<a href="#">/userContent in Git plugin</a> This plugin exposes \$JENKINS_HOME/userContent as Git repository.	1.4
<input type="checkbox"/>	<a href="#">Tracking Git Plugin</a> Lets one project track the Git revisions that are built for another project.	1.0
<input type="checkbox"/>	<a href="#">Team Concert Git Plugin</a> Integrates Jenkins with <a href="#">Rational Team Concert</a> for Jenkins Builds which use Git as source control. This plugin will create traceability links from Jenkins build to Rational Team Concert <a href="#">Work Items</a> and <a href="#">build</a> results. This plugin adds traceability links from a Jenkins build to an RTC build result. It also publishes links to work items and annotates the change log generated by Jenkins with links to RTC Work Items; it leverages the current RTC features and workflows that users are already familiar with such as, emails, toaster popups, reporting, dashboards, etc.	1.0.10
<input type="checkbox"/>	<a href="#">Alternative build chooser</a> An alternative build chooser plugin for the Jenkins git plugin.	1.1
<input type="checkbox"/>	<a href="#">Git plugin</a> This plugin allows you to build SCM. A recent Git runtime is required (1.7.9 minimum, 1.8.x recommended). This plugin is only tested on official <a href="#">git client</a> . Use exotic installations at your own risk.	2.4.2

POP!



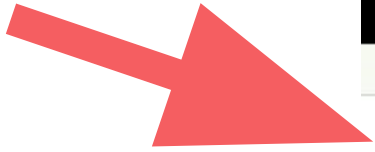
Update information obtained: 1 min

okay good, where were we

oh right

let's make a job

BAM!



Dashboard [Jenki x]

localhost:8080







# Jenkins

Jenkins

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

**Build Queue**

No builds in the queue.

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

**Build Queue**

No builds in the queue.

**Build Executor Status**

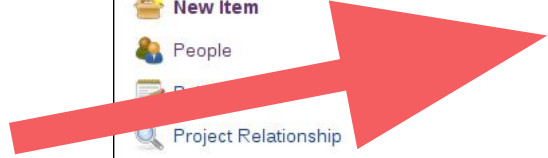
1	Idle
2	Idle
3	Idle
4	Idle

Item name

- Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
- Maven project**  
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
- Pipeline**  
Orchestrates long-running activities that can span multiple build slaves. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- External Job**  
This type of job allows you to record the execution of a process run outside Jenkins, even on a remote machine. This is designed so that you can use Jenkins as a dashboard of your existing automation system. See [the documentation for more details](#).
- Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
- Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
- Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches in one SCM repository.
- Copy existing Item**  
Copy from

OK

**BOINK!**



rugby Config [Jen x] localhost:8080/job/rugby/configure

# Jenkins

rugby > configuration

- Back to Dashboard
- Status
- Changes
- Workspace
- Build Now
- Delete Project
- Configure**
- Move

**Build History** [trend](#)

find

[RSS for all](#) [RSS for failures](#)

Project name:

Description:

[Plain text] [Preview](#)

- Discard Old Builds
- Docker Container
- GitHub project
- Restrict build execution causes
- This build is parameterized
- Throttle build
- Queue Build (No new builds will be executed until the project is re-enabled.)
- Execute concurrent builds (necessary for distributed SCM clients)
- Restrict where this project can be run

Label Expression:

Advanced Prescriptions: [Advanced...](#)

Source Code Management

- None
- CVS
- CVS Projectset
- Git
- Subversion

**Build Triggers**

- Build after other projects are built
- Build periodically
- Build when a change is pushed to GitHub
- GitHub Pull Request Builder
- Poll SCM

**Build Environment**

- Send files or execute commands over SSH before the build starts
- Send files or execute commands over SSH after the build runs

SYSTEM OVERLOAD!



easy mode

## Source Code Management

- None
- CVS
- CVS Projects
- Git



# KNOCK!

Repositories

Repository URL  [?](#)

Credentials   [?](#)

Branches to build

Branch Specifier (blank for 'any')  [?](#)

Repository browser  [?](#)

Additional Behaviours



# PRESTO!

Build when a change is pushed to GitHub

GitHub Pull Request Builder

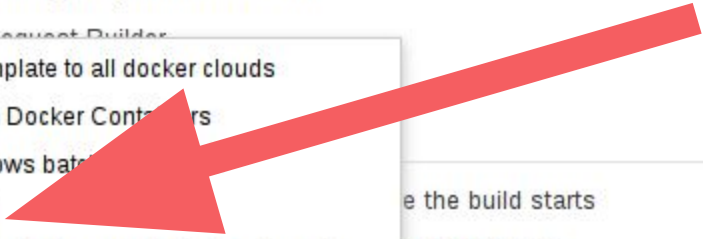
- Add a new template to all docker clouds
- Build / Publish Docker Containers
- Execute Windows batch
- Execute shell
- Execute shell script on remote host using ssh
- Invoke Ansible Ad-Hoc Command
- Invoke Ansible Playbook
- Invoke Ant
- Invoke top-level Maven targets
- Send files or execute commands over SSH
- Set build status to "pending" on GitHub commit
- Start/Stop Docker Containers
- Update build name

Add build step

... the build starts

... the build runs

... and message (Must configure upstream job using



# UAH!

Execute shell

```
Command #!/bin/bash -xe  
  
# Load our RVM installation for maximum fun  
source ~/.rvm/scripts/rvm  
  
# use a custom gemset for my job  
rvm use ruby-2.3.0@{JOB_NAME} --create  
  
# ensure we have bundler  
gem install bundler --no-ri --no-rdoc  
  
# shazam  
bundle install  
  
# kerpow  
bundle exec rake -T
```

[See the list of available environment variables](#)

Delete

Add build step

Post-build Actions









Add post-build action

Save

Apply

pew pew!

WUT!

-  Back to Dashboard
-  **Status**
-  Changes
-  Workspace
-  Build Now
-  Delete Project
-  Configure
-  Move

## Project rugby

 [Workspace](#)  
 [Recent Changes](#)

**SHAZAM!**

## Permalinks

 **Build History** [trend](#) 

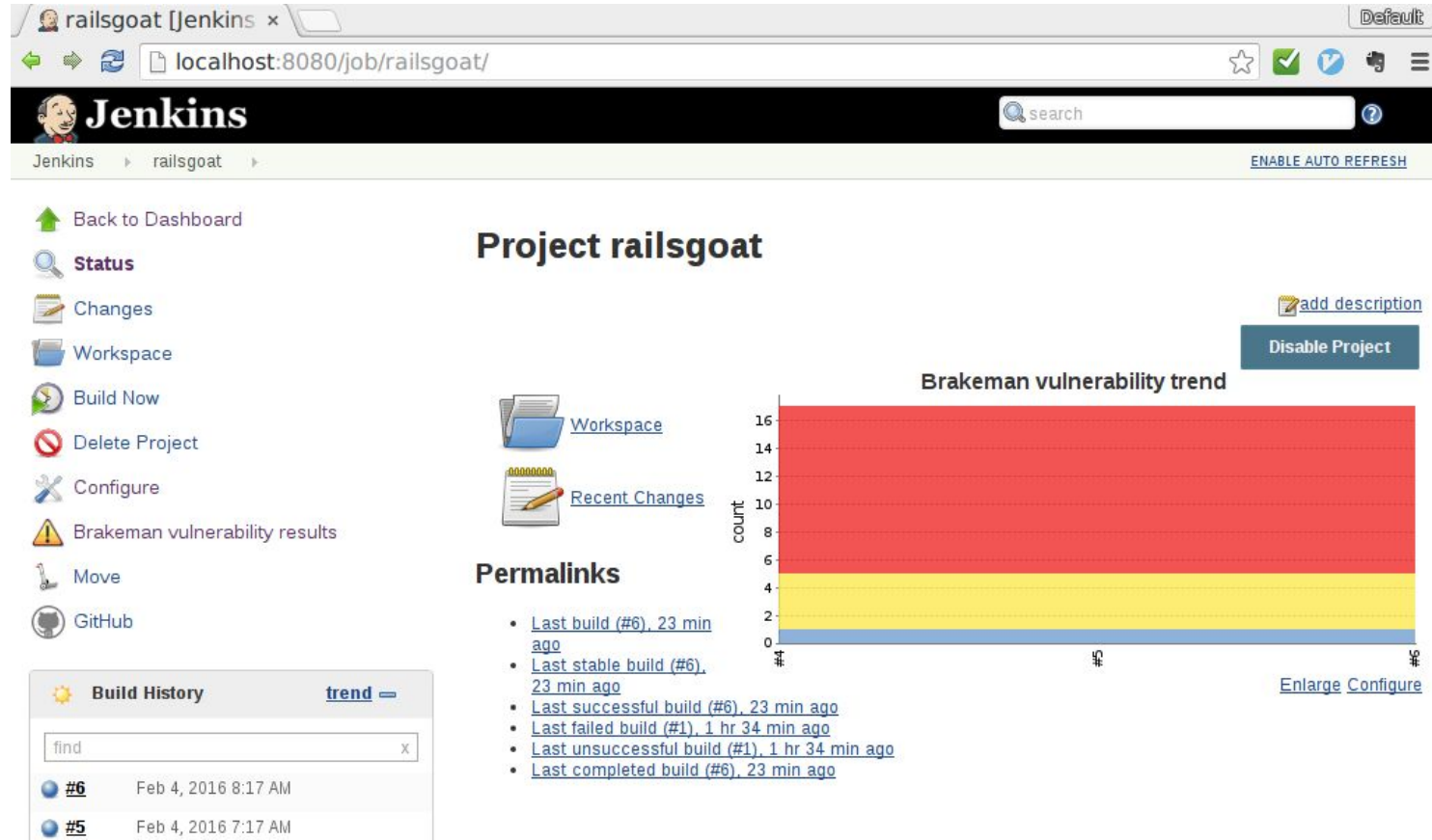


 [RSS for all](#)  [RSS for failures](#)

'kay

advanced mode

# Brakeman plugin

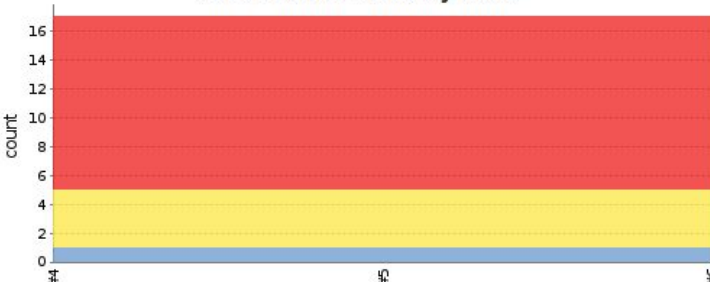


The screenshot shows the Jenkins web interface for the 'railsgoat' project. The browser address bar shows 'localhost:8080/job/railsgoat/'. The Jenkins header includes the logo, a search bar, and navigation links. The left sidebar contains a menu with options like 'Back to Dashboard', 'Status', 'Changes', 'Workspace', 'Build Now', 'Delete Project', 'Configure', 'Brakeman vulnerability results', 'Move', and 'GitHub'. The main content area is titled 'Project railsgoat' and features a 'Brakeman vulnerability trend' chart. The chart shows a count of vulnerabilities over time, with a red area representing high severity (count 11-16), a yellow area representing medium severity (count 5), and a blue area representing low severity (count 1). Below the chart is a 'Permalinks' section with a list of build links. A 'Build History' panel is visible at the bottom left, showing build #6 from Feb 4, 2016 8:17 AM and build #5 from Feb 4, 2016 7:17 AM.

## Project railsgoat

[add description](#) [Disable Project](#)

### Brakeman vulnerability trend



Workspace [Recent Changes](#)

### Permalinks

- [Last build \(#6\), 23 min ago](#)
- [Last stable build \(#6\), 23 min ago](#)
- [Last successful build \(#6\), 23 min ago](#)
- [Last failed build \(#1\), 1 hr 34 min ago](#)
- [Last unsuccessful build \(#1\), 1 hr 34 min ago](#)
- [Last completed build \(#6\), 23 min ago](#)

### Build History

Build	Time
#6	Feb 4, 2016 8:17 AM
#5	Feb 4, 2016 7:17 AM



# More Brakeman

## Post-build Actions

### Publish Brakeman warnings

Brakeman Output File

Health thresholds

 100%

 0%

Configure the thresholds for the build health. If left empty then no health report is created. If the actual number of warnings is between the provided thresholds then the build health is interpolated.

Health priorities

Only priority high
  Priorities high and normal
  All priorities

Determines which warning priorities should be considered when evaluating the build health.

Status thresholds (Totals)

	All priorities	Priority high	Priority normal	Priority low
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

If the number of total warnings is greater than one of these thresholds then a build is considered as unstable or failed, respectively. I.e., a value of 0 means that the build status is changed if there is at least one warning found. Leave this field empty if the state of the build should not depend on the number of warnings.

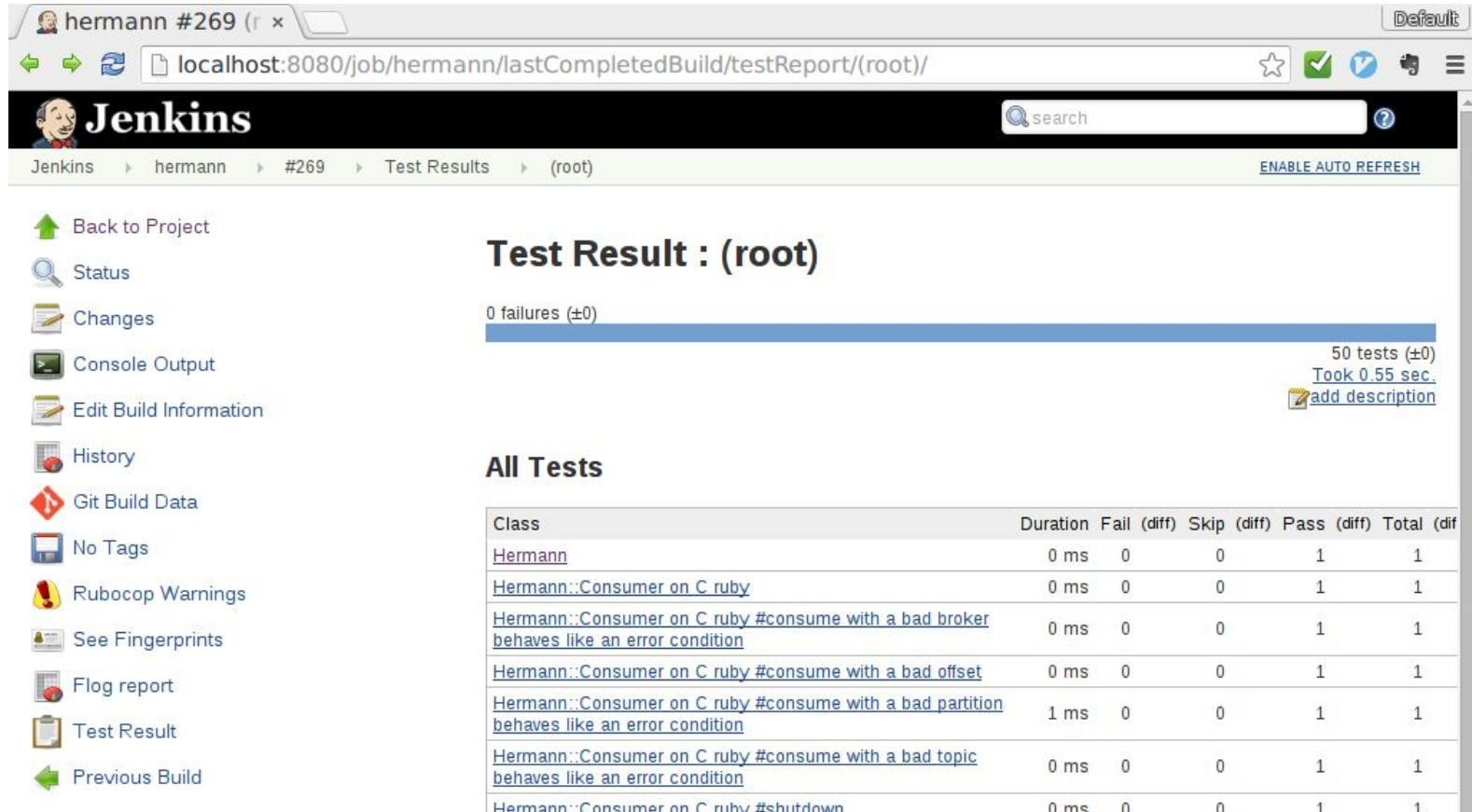
Compute new warnings (based on the last successful build unless another reference build is chosen below)

Trend graph

[You can define the default values for the trend graph in a separate view.](#)

Delete

# JUnit plugin + ci\_reporter



The screenshot shows the Jenkins web interface for a build named 'hermann #269'. The browser address bar shows 'localhost:8080/job/hermann/lastCompletedBuild/testReport/(root)/'. The Jenkins header includes a search bar and a navigation breadcrumb: 'Jenkins > hermann > #269 > Test Results > (root)'. On the right side of the header, there is a link for 'ENABLE AUTO REFRESH'.

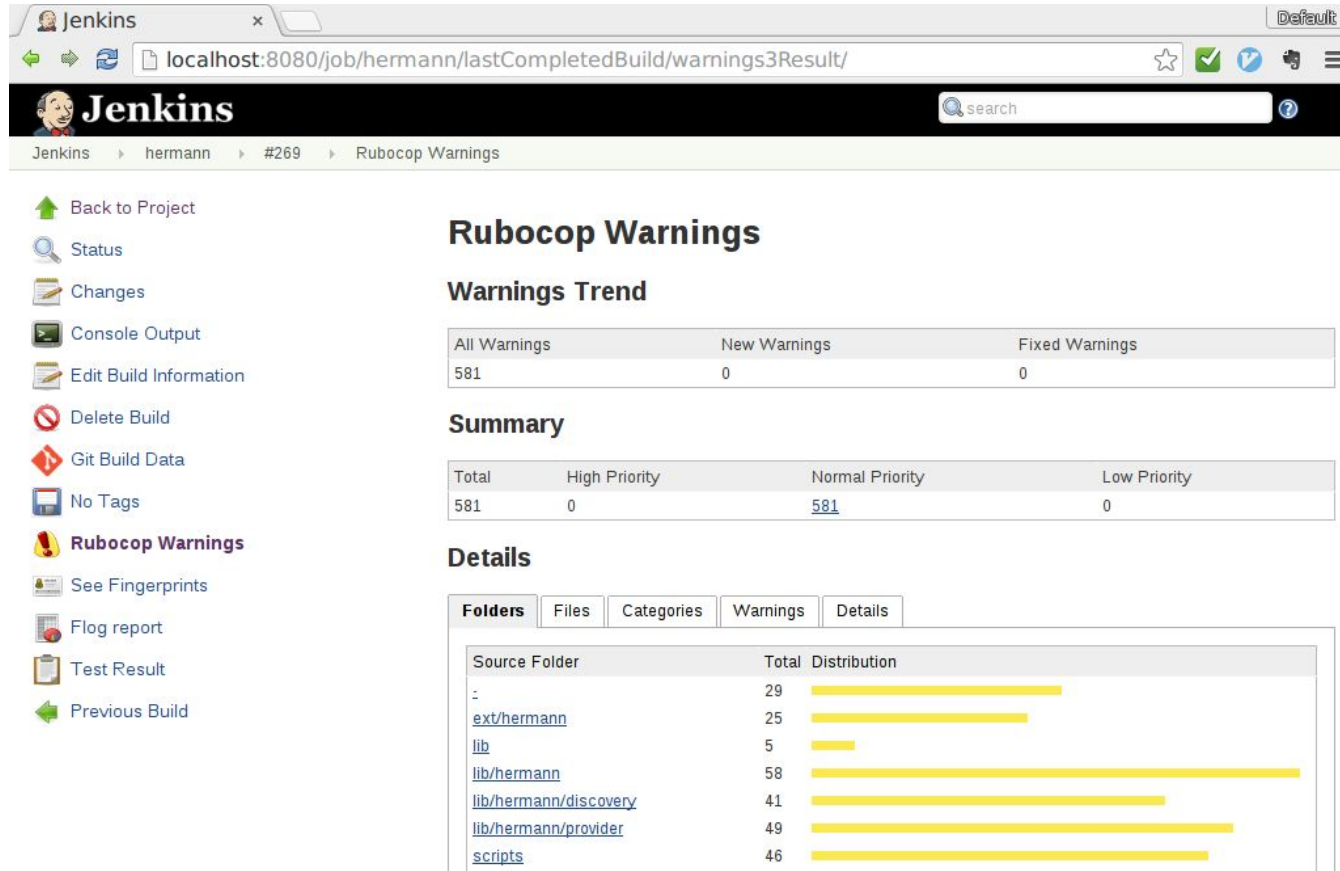
The main content area is titled 'Test Result : (root)'. Below the title, it indicates '0 failures (±0)' and shows a blue progress bar representing 100% success. Summary statistics on the right show '50 tests (±0)', 'Took 0.55 sec.', and a link to 'add description'.

A sidebar on the left contains various navigation options: 'Back to Project', 'Status', 'Changes', 'Console Output', 'Edit Build Information', 'History', 'Git Build Data', 'No Tags', 'Rubocop Warnings', 'See Fingerprints', 'Flog report', 'Test Result', and 'Previous Build'.

The 'All Tests' section contains a table with the following data:

Class	Duration	Fail (diff)	Skip (diff)	Pass (diff)	Total (diff)
<a href="#">Hermann</a>	0 ms	0	0	1	1
<a href="#">Hermann::Consumer on C ruby</a>	0 ms	0	0	1	1
<a href="#">Hermann::Consumer on C ruby #consume with a bad broker behaves like an error condition</a>	0 ms	0	0	1	1
<a href="#">Hermann::Consumer on C ruby #consume with a bad offset</a>	0 ms	0	0	1	1
<a href="#">Hermann::Consumer on C ruby #consume with a bad partition behaves like an error condition</a>	1 ms	0	0	1	1
<a href="#">Hermann::Consumer on C ruby #consume with a bad topic behaves like an error condition</a>	0 ms	0	0	1	1
<a href="#">Hermann::Consumer on C ruby #shutdown</a>	0 ms	0	0	1	1

# Warnings plugin + Rubocop



The screenshot shows the Jenkins interface for a project named 'hermann' at build #269. The page title is 'Rubocop Warnings'. On the left, there is a navigation sidebar with options like 'Back to Project', 'Status', 'Changes', 'Console Output', 'Edit Build Information', 'Delete Build', 'Git Build Data', 'No Tags', 'Rubocop Warnings' (highlighted), 'See Fingerprints', 'Flog report', 'Test Result', and 'Previous Build'. The main content area is titled 'Rubocop Warnings' and contains a 'Warnings Trend' table, a 'Summary' table, and a 'Details' section with a table of source folders and their warning counts.

## Rubocop Warnings








### Warnings Trend

All Warnings	New Warnings	Fixed Warnings
581	0	0

### Summary

Total	High Priority	Normal Priority	Low Priority
581	0	<a href="#">581</a>	0

### Details

Source Folder	Total	Distribution
-	29	
<a href="#">ext/hermann</a>	25	
<a href="#">lib</a>	5	
<a href="#">lib/hermann</a>	58	
<a href="#">lib/hermann/discovery</a>	41	
<a href="#">lib/hermann/provider</a>	49	
<a href="#">scripts</a>	46	

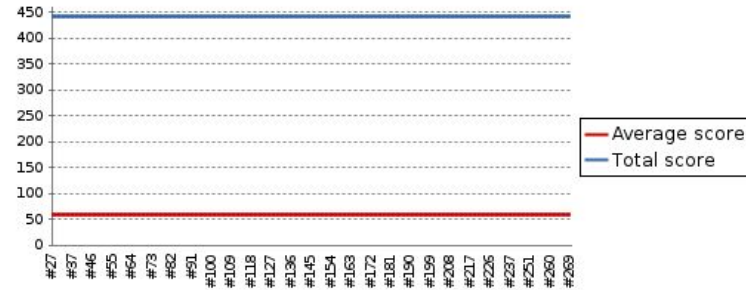
# RubyMetrics plugin + flog

Jenkins localhost:8080/job/hermann/270/flog/ Default

Jenkins hermann #270 Flog report ENABLE AUTO REFRESH

- [Back to Project](#)
- [Status](#)
- [Changes](#)
- [Console Output](#)
- [Edit Build Information](#)
- [Delete Build](#)
- [Git Build Data](#)
- [No Tags](#)
- [Rubocop Warnings](#)
- [See Fingerprints](#)
- [Flog report](#)**
- [Test Result](#)
- [Previous Build](#)

## Flog report



Total score: 440.8      Average score: 58.799995

```

71.6: lib/hermann/provider/java_simple_consumer.rb
  20.9: Hermann::Provider::JavaSimpleConsumer#consume
      lib/hermann/provider/java_simple_consumer.rb:73
    • 1.7: next
    • 0.4: lit_fixnum
  
```

# HTMLPublisher plugin + Simplecov

localhost:8080/job/hermann/Coverage\_Report/

[Back to hermann](#) [index](#) [Zip](#) Generated 3 minutes ago

## All Files (78.3%)

All Files (78.3% covered at 1.96 hits/line)

19 files in total. 825 relevant lines. 646 lines covered and 179 lines missed

Search:

File	% covered	Lines	Relevant Lines	Lines covered	Lines missed	Avg. Hits / Line
<a href="#">lib/hermann/provider/java_simple_consumer.rb</a>	32.14 %	144	56	18	38	0.3
<a href="#">spec/integration/producer_spec.rb</a>	47.73 %	72	44	21	23	0.5
<a href="#">lib/hermann/provider/java_producer.rb</a>	50.0 %	96	30	15	15	0.5
<a href="#">lib/hermann.rb</a>	57.14 %	35	14	8	6	3.1
<a href="#">spec/providers/java_simple_consumer_spec.rb</a>	62.07 %	90	58	36	22	0.6
<a href="#">spec/providers/java_producer_spec.rb</a>	68.42 %	95	57	39	18	0.7
<a href="#">lib/hermann/timeout.rb</a>	70.0 %	37	20	14	6	0.8
<a href="#">lib/hermann/discovery/zookeeper.rb</a>	76.06 %	152	71	54	17	1.8
<a href="#">spec/hermann_lib/producer_spec.rb</a>	84.38 %	116	64	54	10	1.3
<a href="#">lib/hermann/consumer.rb</a>	86.21 %	71	29	25	4	2.8
<a href="#">lib/hermann/producer.rb</a>	88.71 %	154	62	55	7	6.9
<a href="#">spec/producer_spec.rb</a>	92.7 %	229	137	127	10	1.9
<a href="#">lib/hermann/result.rb</a>	92.86 %	74	28	26	2	6.7
<a href="#">spec/hermann_spec.rb</a>	95.24 %	36	21	20	1	1.4
<a href="#">lib/hermann/errors.rb</a>	100.0 %	32	11	11	0	1.9
<a href="#">spec/discovery/zookeeper_spec.rb</a>	100.0 %	101	56	56	0	1.7

protip

```
for f in scripts/ci/*.sh; do  
    bash $f;  
done
```

# rescue nil

- @agentdero
- rtyler on Freenode, GitHub, keybase.io
- [tyler@linux.com](mailto:tyler@linux.com)
- this presentation: <https://goo.gl/pTDdMS>
- <https://jenkins-ci.org/s/ruby>
- Plugins to check out:
  - Pipeline
  - JUnit
  - Brakeman
  - Cucumber Test Result
  - Docker
  - HTMLPublisher
  - Email-ext
  - RubyMetrics



bonus round

# Jenkins 2.0

```
/* Jenkinsfile */  
node('docker-capable') {  
    docker.image('rvm-image').inside {  
        checkout scm  
        sh 'bundle exec rake'  
    }  
}
```

## Getting Started

# Initial Jenkins Setup

Choose plugins. This will add features to your Jenkins environment.

### Start with recommended plugins

Install the set of plugins the community finds most useful

### Customize your plugins

Select from a community approved list of plugins

## Plugin Selection

- General
- Organization and Administration
- Build Tools
- Build Analysis and Reporting
- SCM
- Distributed Builds and Containers
- User Management and Security
- Notifications and Publishing

All | None | Recommended

Selected (8/19)

Note that the full list of plugins is not shown here. Additional plugins can be installed in the **Plugin Manager** once the initial setup is complete. [See the Wiki for more information.](#)

### General (1/2)

**External Monitor Job Type Plugin**  
Adds the ability to monitor the result of externally executed jobs.

**Translation Assistance plugin**  
This plugin adds an additional dialog box in every page, which enables people to contribute localizations for the messages they are seeing in the current page.

### Organization and Administration (1/1)

**OWASP Markup Formatter Plugin**  
Uses policy definitions to allow limited HTML markup in user-submitted text.

### Build Tools (0/2)

**Ant Plugin**

Back

Install



# rescue nil

- @agentdero
  - rtyler on Freenode, GitHub, keybase.io
  - [tyler@linux.com](mailto:tyler@linux.com)
  - this presentation: <https://goo.gl/pTDdMS>
- <https://jenkins-ci.org/s/ruby>
  - Plugins to check out:
    - Pipeline
    - JUnit
    - Brakeman
    - Cucumber Test Result
    - Docker
    - HTMLPublisher
    - Email-ext
    - RubyMetrics