Continuous Integration im Rechenzentrum

Michael Prokop

OSDC.de
OPEN SOURCE DATA CENTER CONFERENCE
17.- 18. APRIL 2013 | NUREMBERG

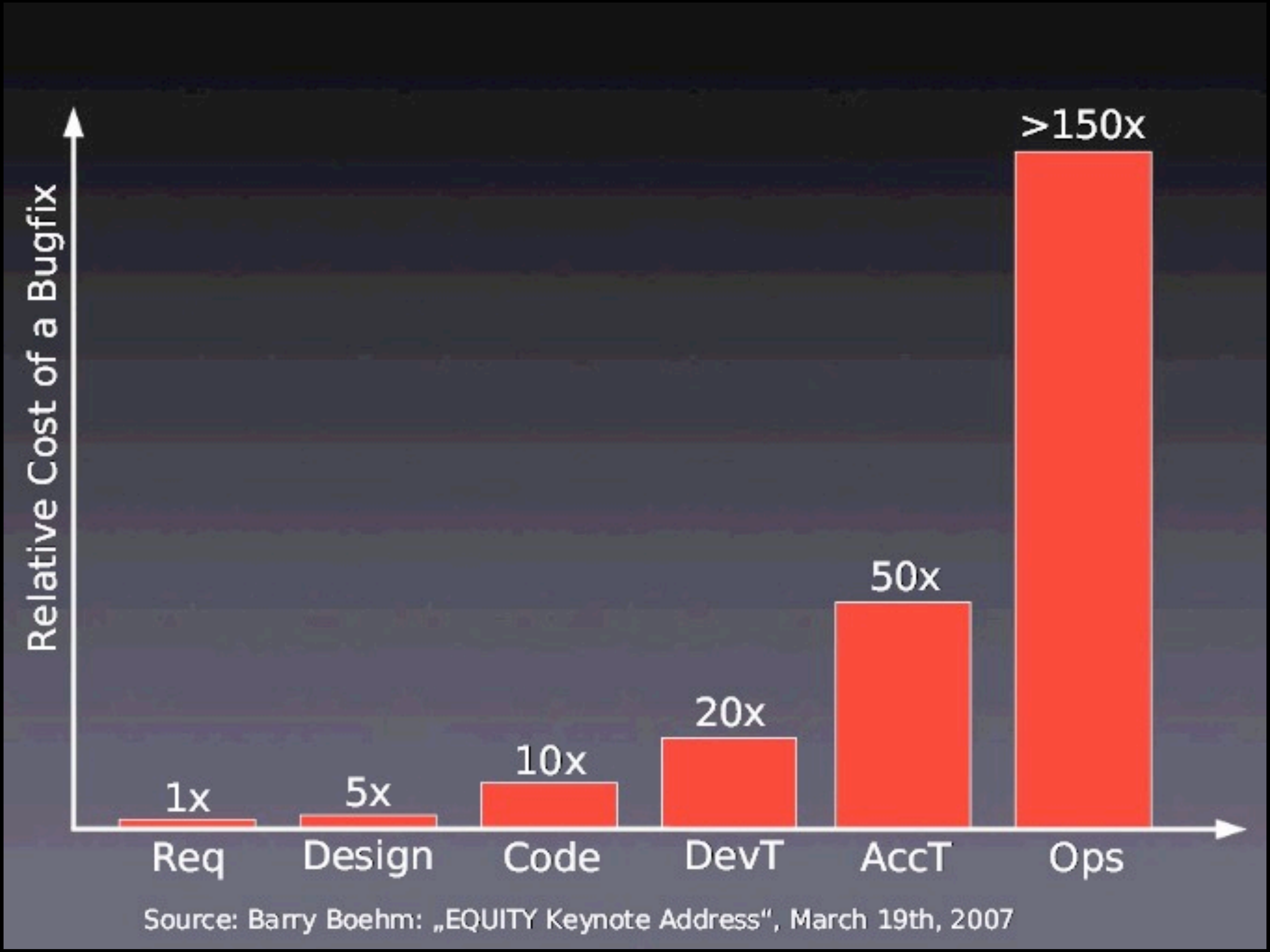
Roadmap

- Begriffsklärung + Gründe für CI
- CI-Server Jenkins
- CI mit Debian-Paketen
- Weitere Beispiele für Einsatz von CI/CD im Rechenzentrum
- Best Practices

Begriffsklärung

- Continuous Integration
- Continuous Deployment
- Continuous Delivery

Warum CI?

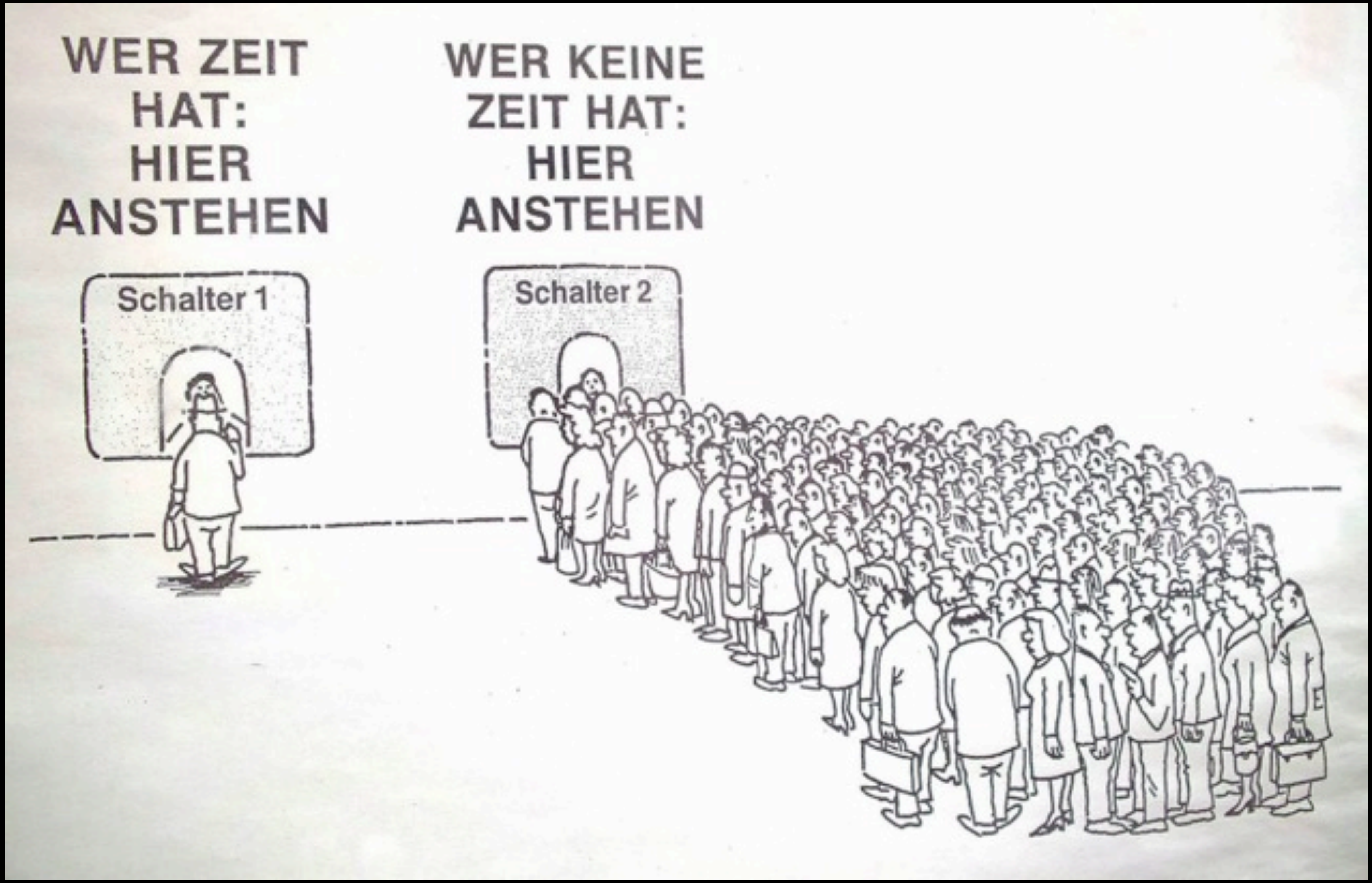


Relative Cost of a Bugfix
1x
5x
10x
20x
50x
>150x
Req
Design
Code
DevT
AccT
Ops
Source: Barry Boehm: „EQUITY Keynote Address", March 19th, 2007

Unabhängigkeit

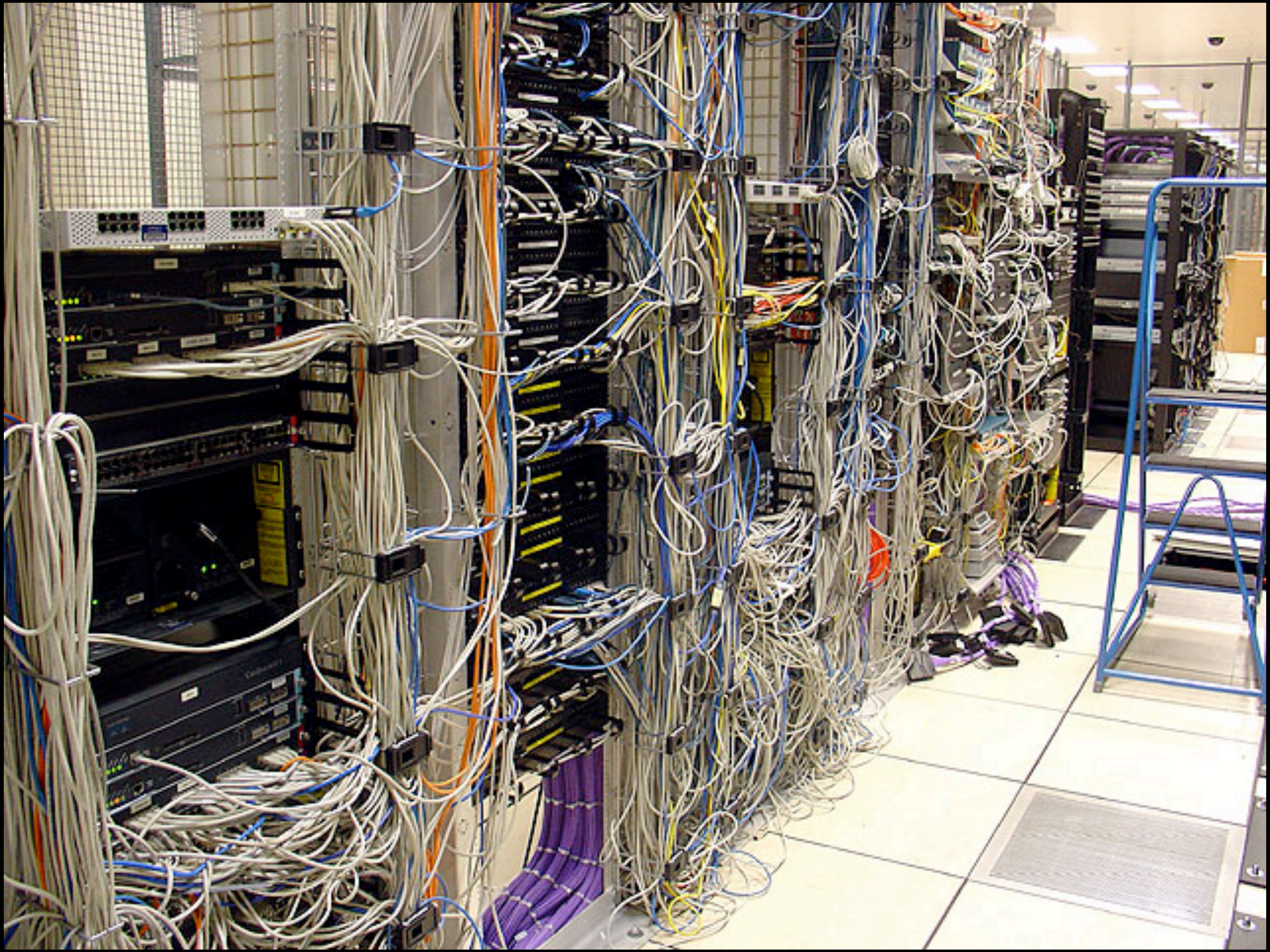
Quelle: http://decarabia.soup.io/post/241926962/Image

Skalierbarkeit


WER ZEIT HAT: HIER ANSTEHEN
WER KEINE ZEIT HAT: HIER ANSTEHEN
Schalter 1
Schalter 2

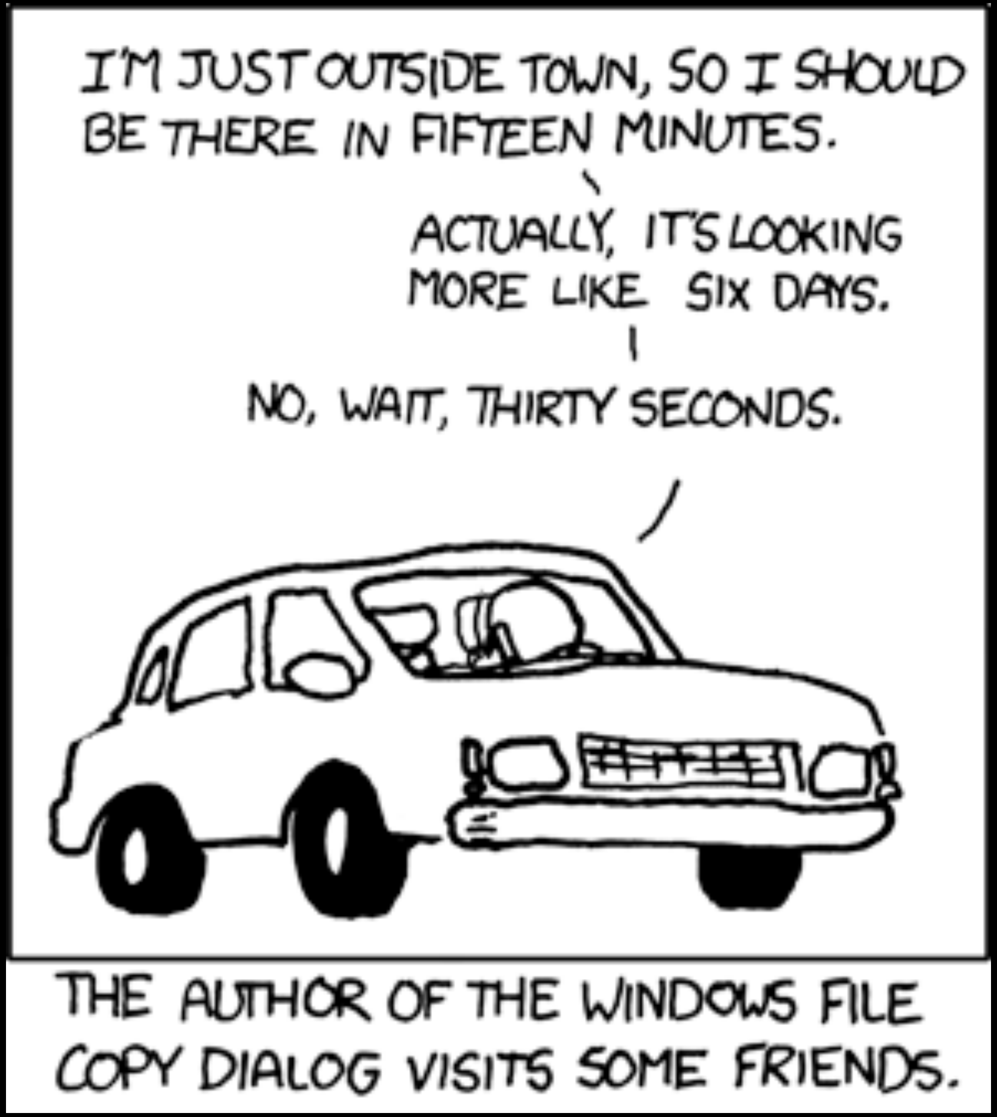
Quelle: http://up.arab-x.com/May12/M9b65492.jpg

Reproduzierbar



Quelle: http://www.flickr.com/photos/route79/13120127/

Berechenbar


I'M JUST OUTSIDE TOWN, SO I SHOULD BE THERE IN FIFTEEN MINUTES.
ACTUALLY, IT'S LOOKING MORE LIKE SIX DAYS.
NO, WAIT, THIRTY SECONDS.
THE AUTHOR OF THE WINDOWS FILE COPY DIALOG VISITS SOME FRIENDS.

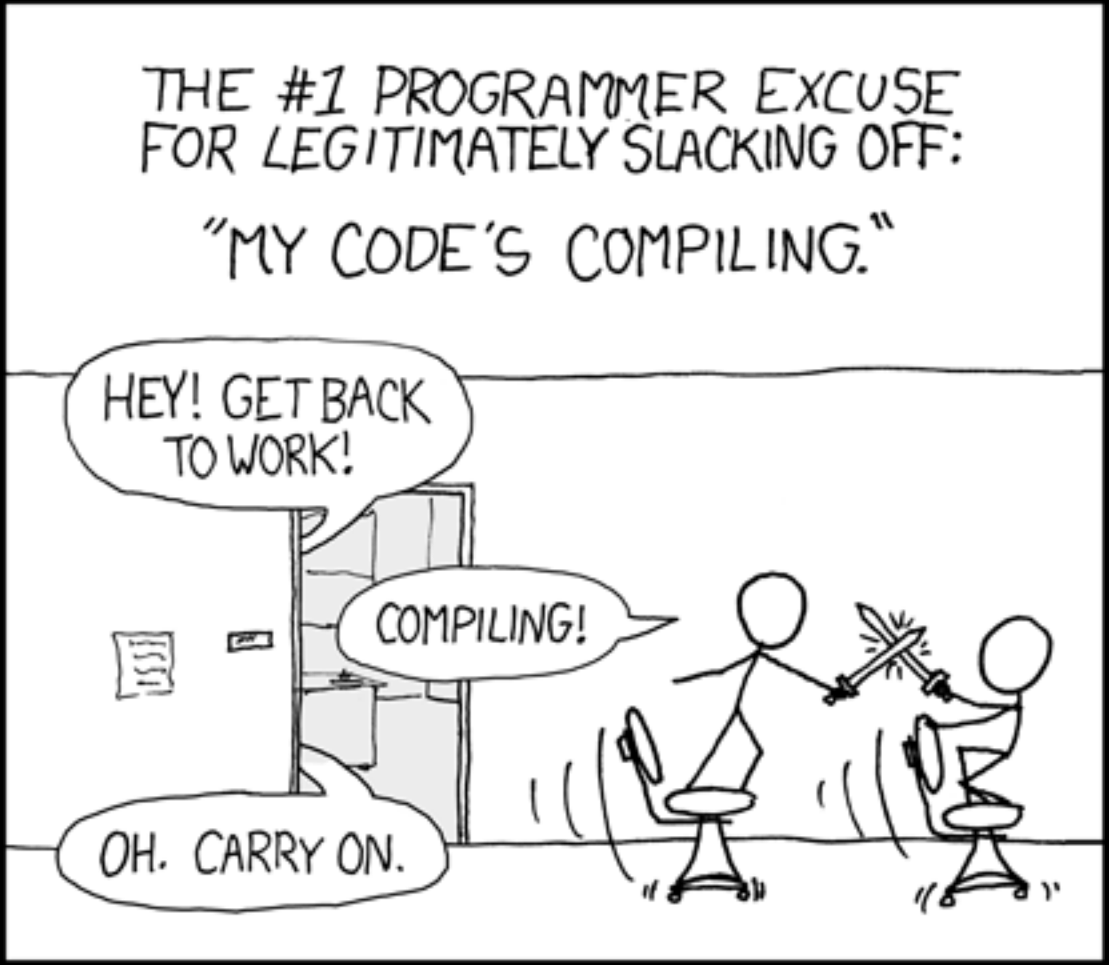
Quelle: http://xkcd.com/612/

Versionskontrolle

- Nur was unter Versionskontrolle ist zählt
- Distributed VCS ftw!

% make

alleine ist NICHT genug



Quelle: http://xkcd.com/303/

Jenkins

das “Wordpress der CI-Server”

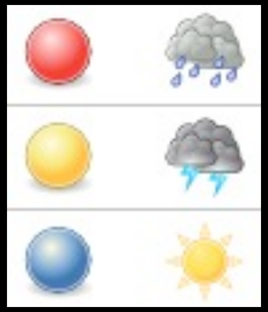
Jenkins

- Open Source (MIT Lizenz)
- wöchentliche && LTS-Releases
- >60k Installationen (Stand Ende März)
- >700 Plugins (Stand Mitte April)
- Community

FAQ #1 - Java?!

- ja, RAM hilft
- nein, man braucht keinen Javacode anzugreifen
- nein, unterstützt nicht nur Java-Projekte

FAQ #2: Blau?! Gelb?!



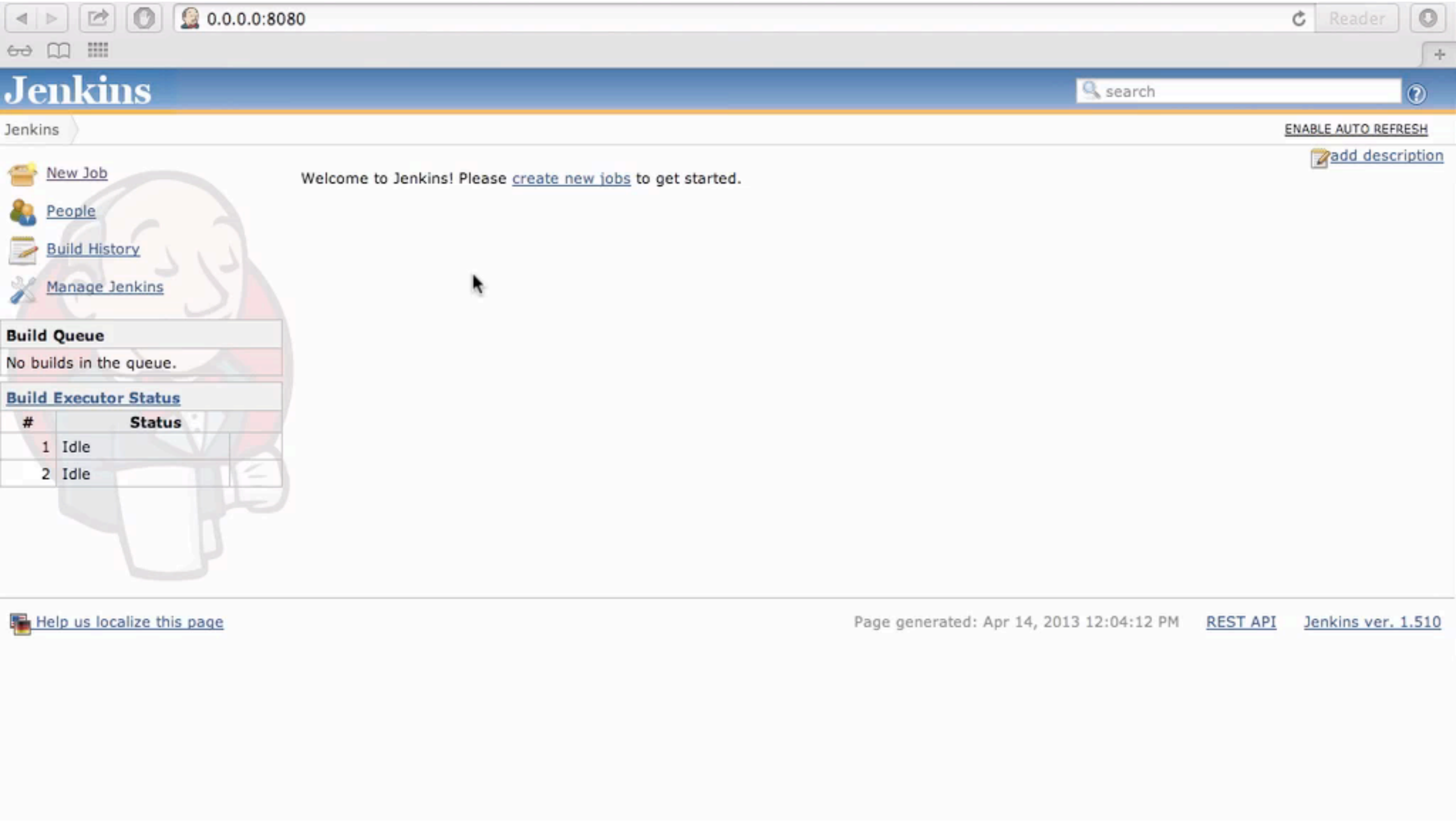
- http://jenkins-ci.org/content/why-does-jenkins-have-blue-balls

Getting Started

```
% curl -L -o jenkins.war \
http://mirrors.jenkins-ci.org/war/latest/jenkins.war
```

```
% java -jar jenkins.war
% $BROWSER http://0.0.0.0:8080
```

Disclaimer: bitte die (Upstream-)Software-Pakete nutzen



Jenkins

Welcome to Jenkins! Please create new jobs to get started.

Build Queue

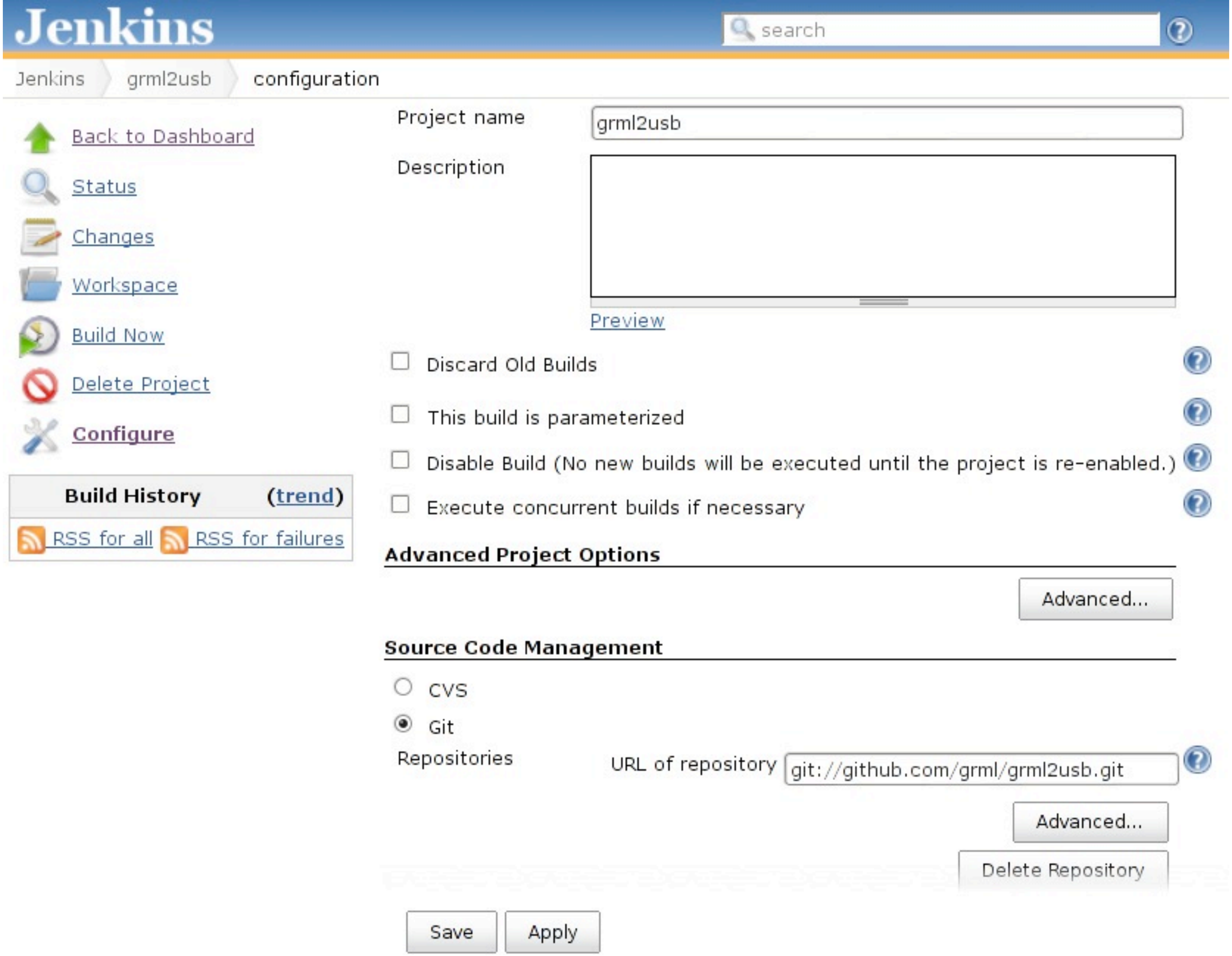
No builds in the queue.

Build Executor Status

#	Status
1	Idle
2	Idle

Help us localize this page

Page generated: Apr 14, 2013 12:04:12 PM REST API Jenkins ver. 1.510



Project name: grml2usb

Description

Preview

☐ Discard Old Builds

☐ This build is parameterized

☐ Disable Build (No new builds will be executed until the project is re-enabled.)

☐ Execute concurrent builds if necessary

Advanced Project Options

Advanced...

Source Code Management

○ CVS

● Git

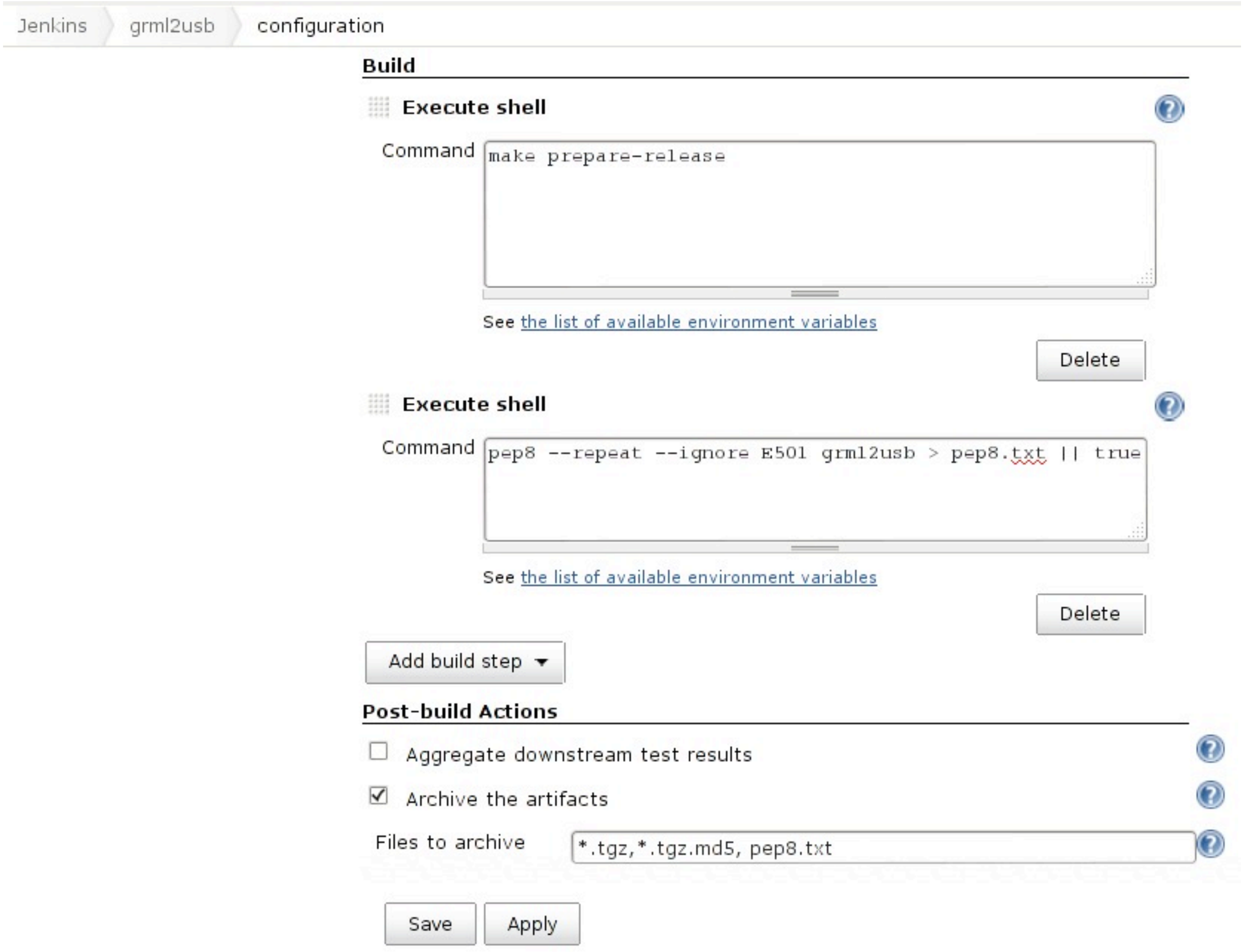
Repositories

URL of repository: git://github.com/grml/grml2usb.git

Advanced...

Delete Repository

Save Apply



Jenkins > grml2usb > configuration

Build

Execute shell

Command

```
make prepare-release
```

See the list of available environment variables

Execute shell

Command

```
pep8 --repeat --ignore E501 grml2usb > pep8.txt || true
```

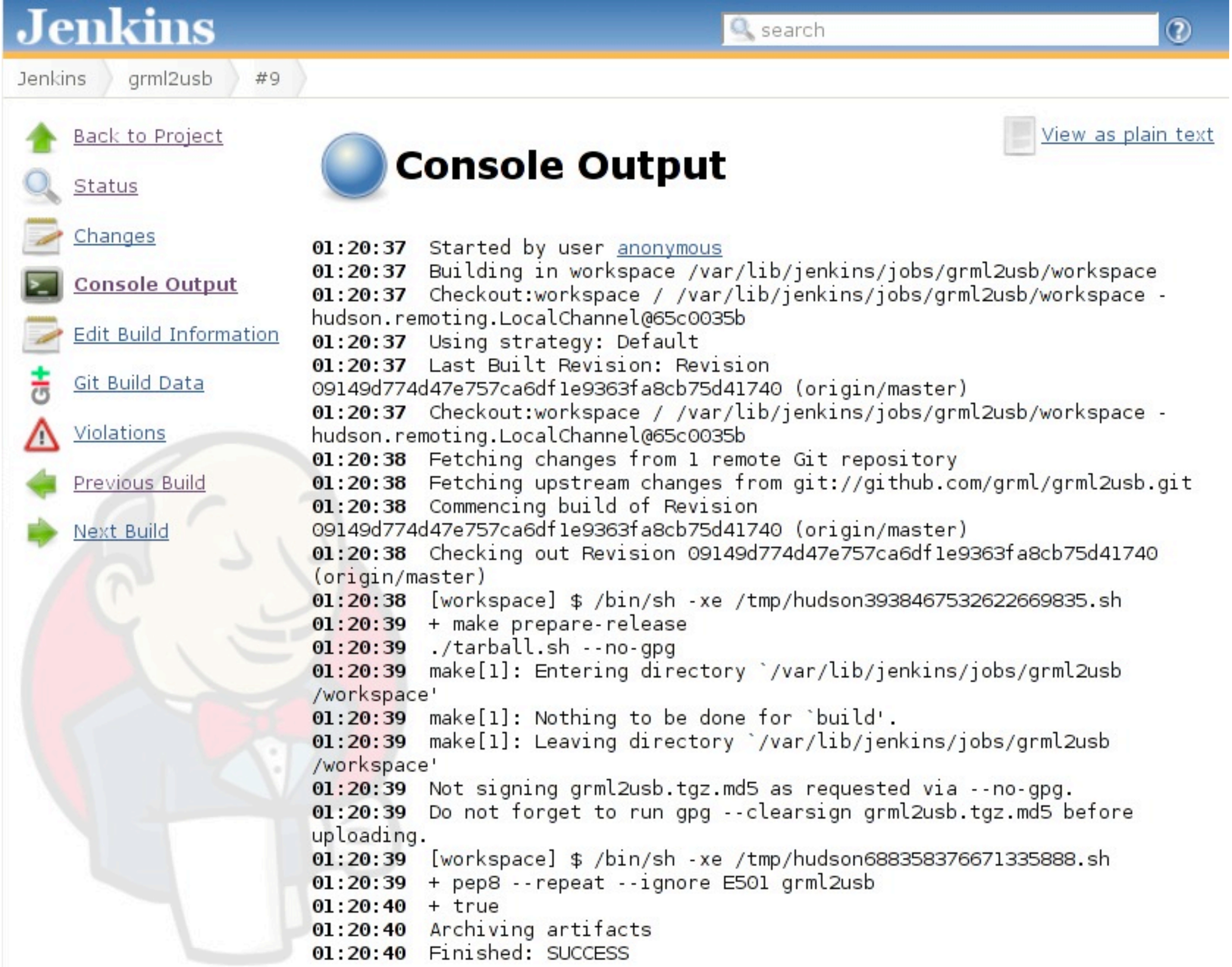
See the list of available environment variables

Post-build Actions

☐ Aggregate downstream test results

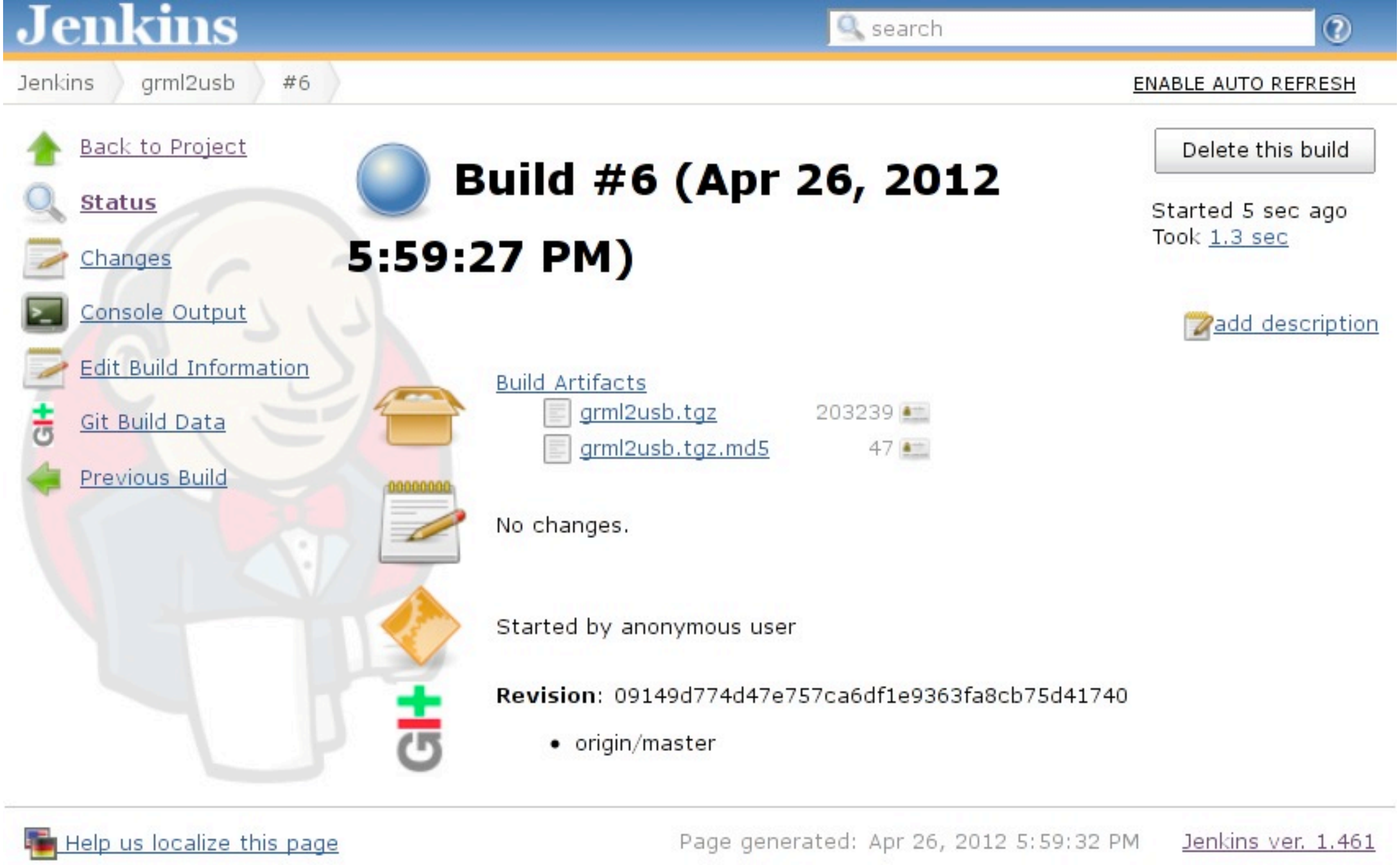
☑ Archive the artifacts

Files to archive: *.tgz,*.tgz.md5, pep8.txt



Console Output

```
01:20:37  Started by user anonymous
01:20:37  Building in workspace /var/lib/jenkins/jobs/grml2usb/workspace
01:20:37  Checkout:workspace / /var/lib/jenkins/jobs/grml2usb/workspace -
hudson.remoting.LocalChannel@65c0035b
01:20:37  Using strategy: Default
01:20:37  Last Built Revision: Revision
09149d774d47e757ca6df1e9363fa8cb75d41740 (origin/master)
01:20:37  Checkout:workspace / /var/lib/jenkins/jobs/grml2usb/workspace -
hudson.remoting.LocalChannel@65c0035b
01:20:38  Fetching changes from 1 remote Git repository
01:20:38  Fetching upstream changes from git://github.com/grml/grml2usb.git
01:20:38  Commencing build of Revision
09149d774d47e757ca6df1e9363fa8cb75d41740 (origin/master)
01:20:38  Checking out Revision 09149d774d47e757ca6df1e9363fa8cb75d41740
(origin/master)
01:20:38  [workspace] $ /bin/sh -xe /tmp/hudson3938467532622669835.sh
01:20:39  + make prepare-release
01:20:39  ./tarball.sh --no-gpg
01:20:39  make[1]: Entering directory `/var/lib/jenkins/jobs/grml2usb
/workspace'
01:20:39  make[1]: Nothing to be done for `build'.
01:20:39  make[1]: Leaving directory `/var/lib/jenkins/jobs/grml2usb
/workspace'
01:20:39  Not signing grml2usb.tgz.md5 as requested via --no-gpg.
01:20:39  Do not forget to run gpg --clearsign grml2usb.tgz.md5 before
uploading.
01:20:39  [workspace] $ /bin/sh -xe /tmp/hudson6883583766713358888.sh
01:20:39  + pep8 --repeat --ignore E501 grml2usb
01:20:40  + true
01:20:40  Archiving artifacts
01:20:40  Finished: SUCCESS
```



Jenkins

search

Jenkins › grml2usb › #6

ENABLE AUTO REFRESH

Back to Project

Status

Changes

Console Output

Edit Build Information

Git Build Data

Previous Build

Build #6 (Apr 26, 2012 5:59:27 PM)

Delete this build

Started 5 sec ago
Took 1.3 sec

add description

Build Artifacts

grml2usb.tgz	203239
grml2usb.tgz.md5	47

No changes.

Started by anonymous user

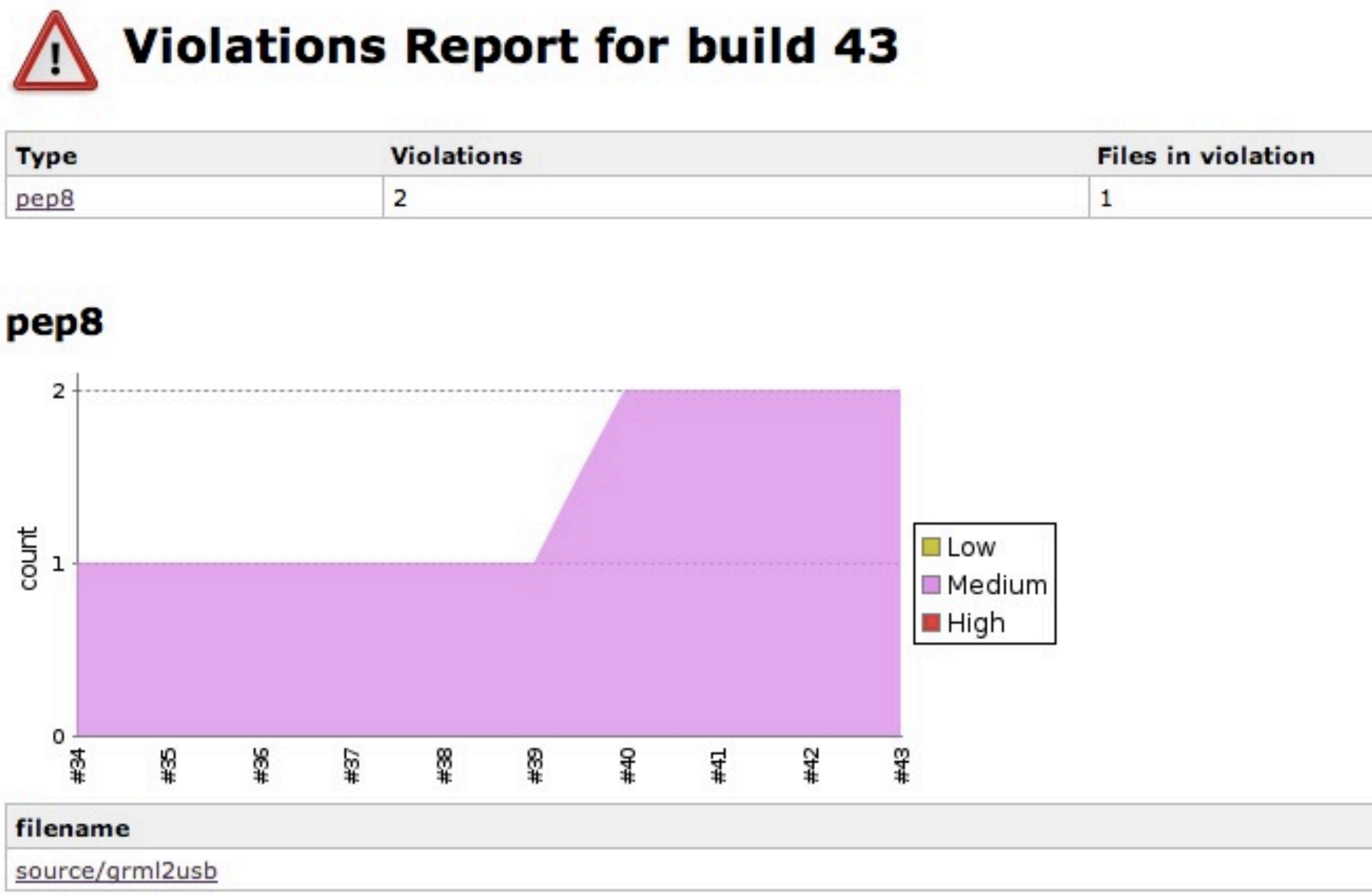
Revision: 09149d774d47e757ca6df1e9363fa8cb75d41740

- origin/master

Help us localize this page

Page generated: Apr 26, 2012 5:59:32 PM

Jenkins ver. 1.461

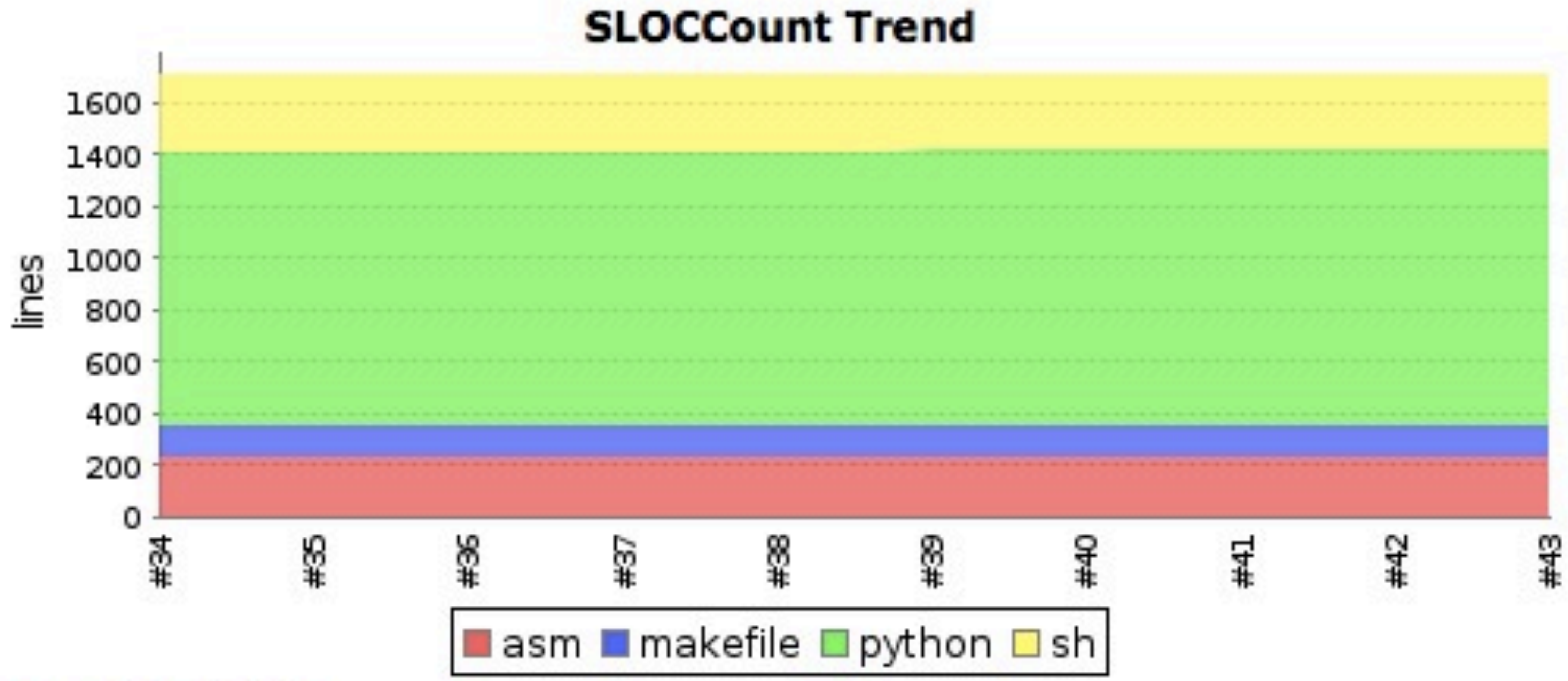


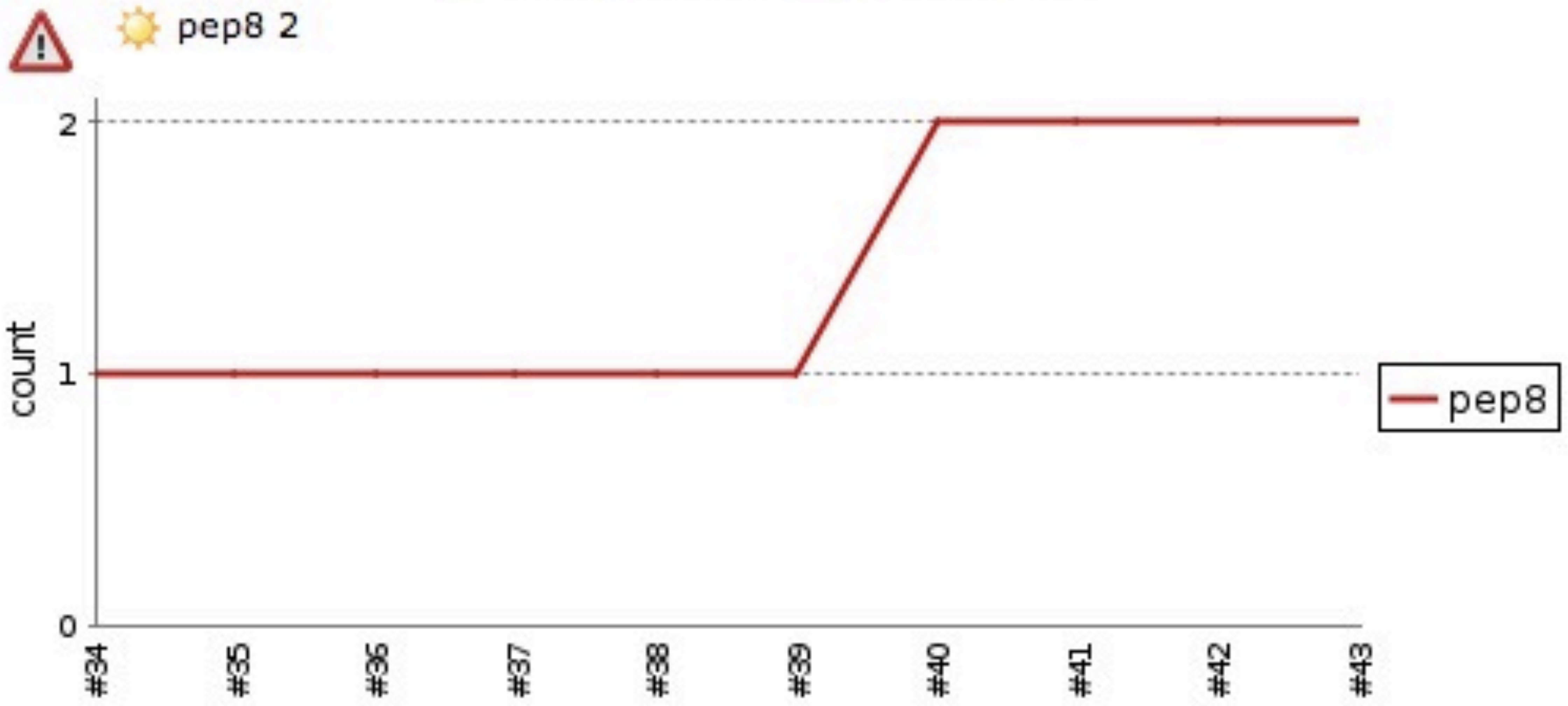
Violations Report for build 43

Type	Violations	Files in violation
pep8	2	1

pep8

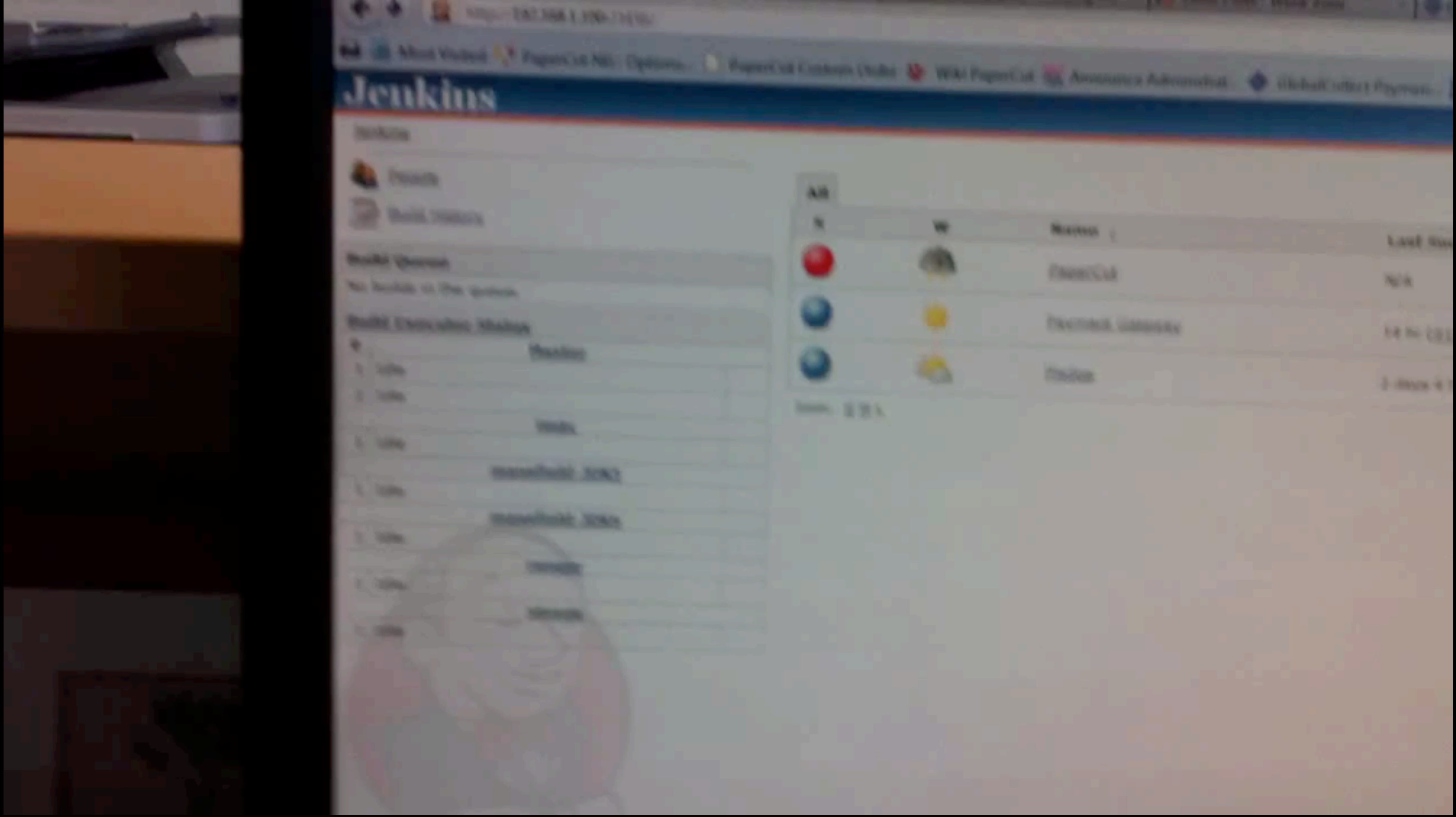
filename
source/grml2usb


SLOCCount Trend
lines
1600
1400
1200
1000
800
600
400
200
0
#34
#35
#36
#37
#38
#39
#40
#41
#42
#43
asm
makefile
python
sh


pep8 2
count
2
1
0
#34
#35
#36
#37
#38
#39
#40
#41
#42
#43
pep8

Bestandteile einer Buildpipeline, u.a.

- Build Artifacts (*.jar, *.deb, *.rpm,...)
- Stages (development, testing, production,...)
- Q/A-Tests (unit/component/system/...)
- Notifications



https://www.youtube.com/watch?v=1EGk2rvZe8A

CI mit Debian-Paketen

jenkins-debian-glue

Debian Packaging

- dpkg [v3] + debhelper [v8]
- dh-make, dh-make-perl, dh-make-php, dh-make-ruby/gem2deb
- fpm (https://github.com/jordansissel/fpm)
- {cvs,svn,git,...}-buildpackage
- cowbuilder/pbuilder/sbuild/...
- reprepro/dak/freight/...

jenkins-debian-glue.org

- Debian-Pakete kontrolliert bauen
- Auch für Nicht-Debian-Entwickler benutzbar (reprepro/freight/cowbuilder/...)
- Unterstützt Subversion + Git ootb
- Vorwiegend Shell, ein wenig Ruby/Python/ Perl (je nach Einsatz) -> leicht adaptierbar

jenkins-debian-glue im Praxiseinsatz

- Grml (http://jenkins.grml.org/)
  - hostet u.a. dpkg, FAI, initramfs-tools
- PostgreSQL (https://wiki.postgresl.org/wiki/Apt)
- Icinga (http://icingabuild.dus.dg-i.net)
- LLVM Debian/Ubuntu (http://llvm.org/apt/)

Source-Pakete

- (Upstream-)Source (orig.tar.gz)
- Debian-Änderungen (debian.tar.gz) [opt.]
- Control-Datei (.dsc)

Wichtig: nur einmal pro Paket Erstellen

Binary-Pakete

- *_all.deb/*_amd64.deb/*_i386.deb
- *.changes, *.dsc, *.tar.gz

Wichtig: pro Architektur einmal Bauen (Ausnahme für “Architecture: all”)

Repository

- reprepro und freight Handling ootb
  - http://mirrorer.alioth.debian.org/
  - https://github.com/rcrowley/freight/
- standardmässig ein Repository pro Projekt
- sog. Release-Repository + trunk-release-Repository einfach aktivierbar

-> kein manuelles Setup/Management notwendig

Q/A-Tests

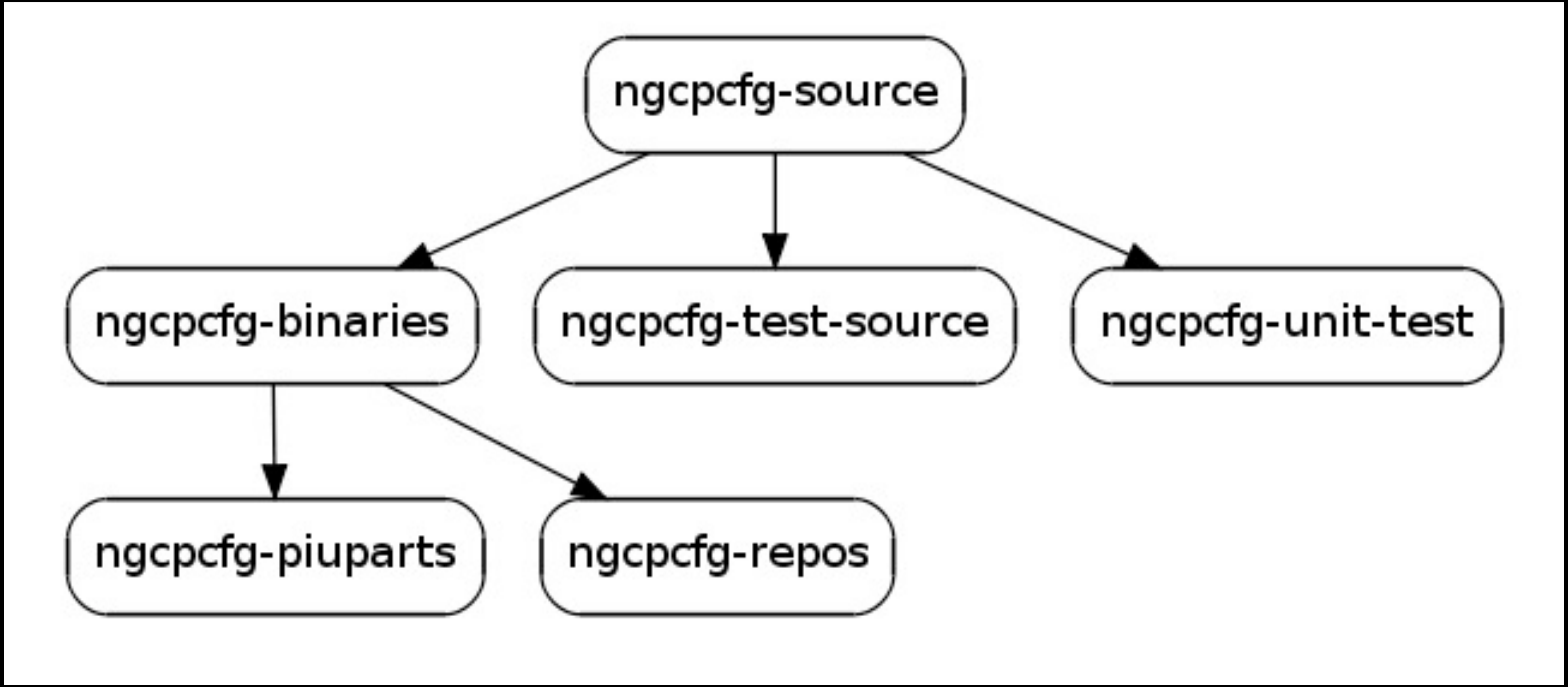
- lintian: Paketqualität
- autopkgtest: Paket-Tests in definierter Umgebung
- piuparts: Install/Deinstall/Upgrade-Tests
- perlcritics/checkbashism/...: Code-Policies

Resultat als TAP/jUnit/...-Report in Jenkins

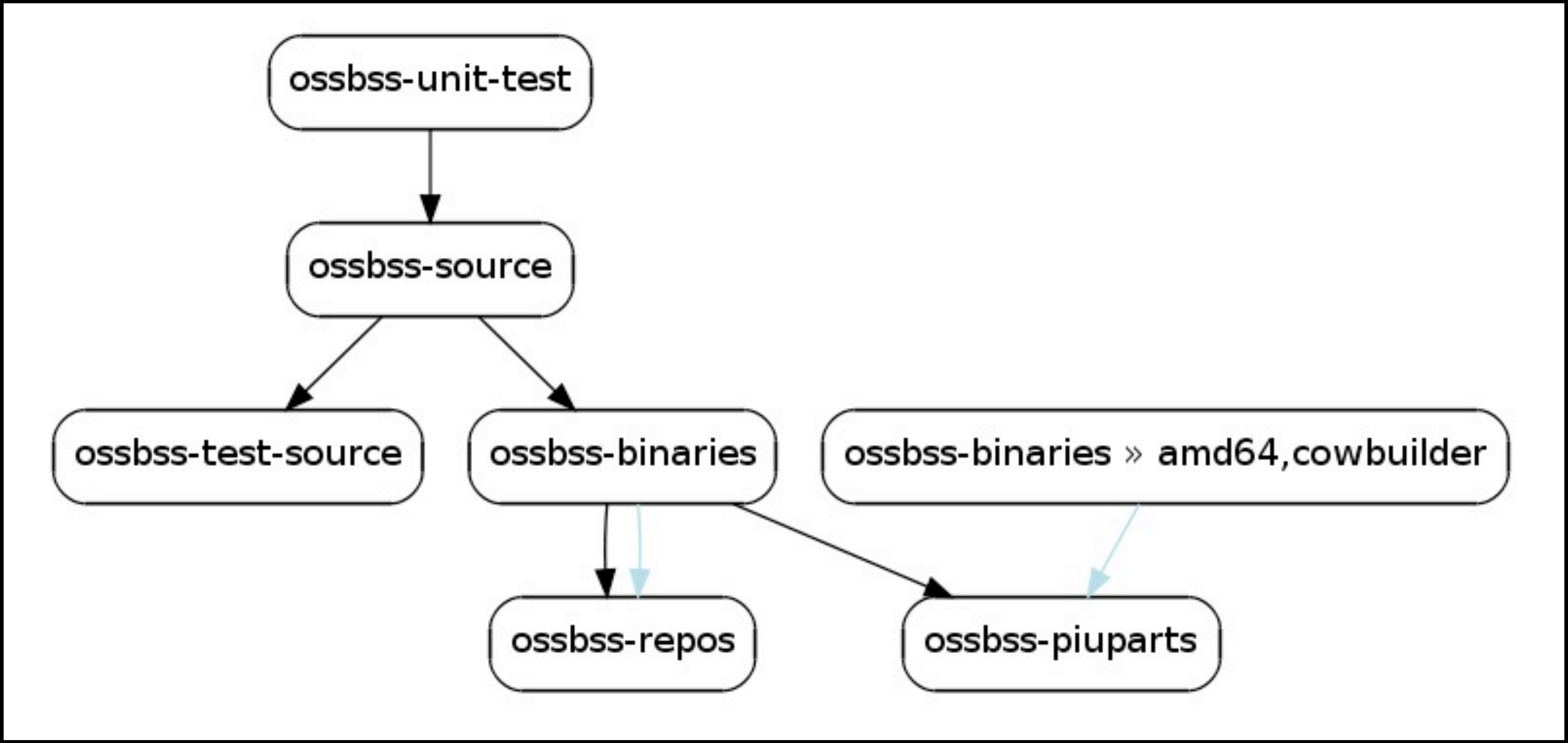
jenkins-debian-glue

- *-source -> Source-Paket
- *-binaries -> Binary-Paket(e)
- *-repos -> Repository-Handling (optional)
- *-piuparts -> Install/Deinstall/Upgrade-Testing (optional)

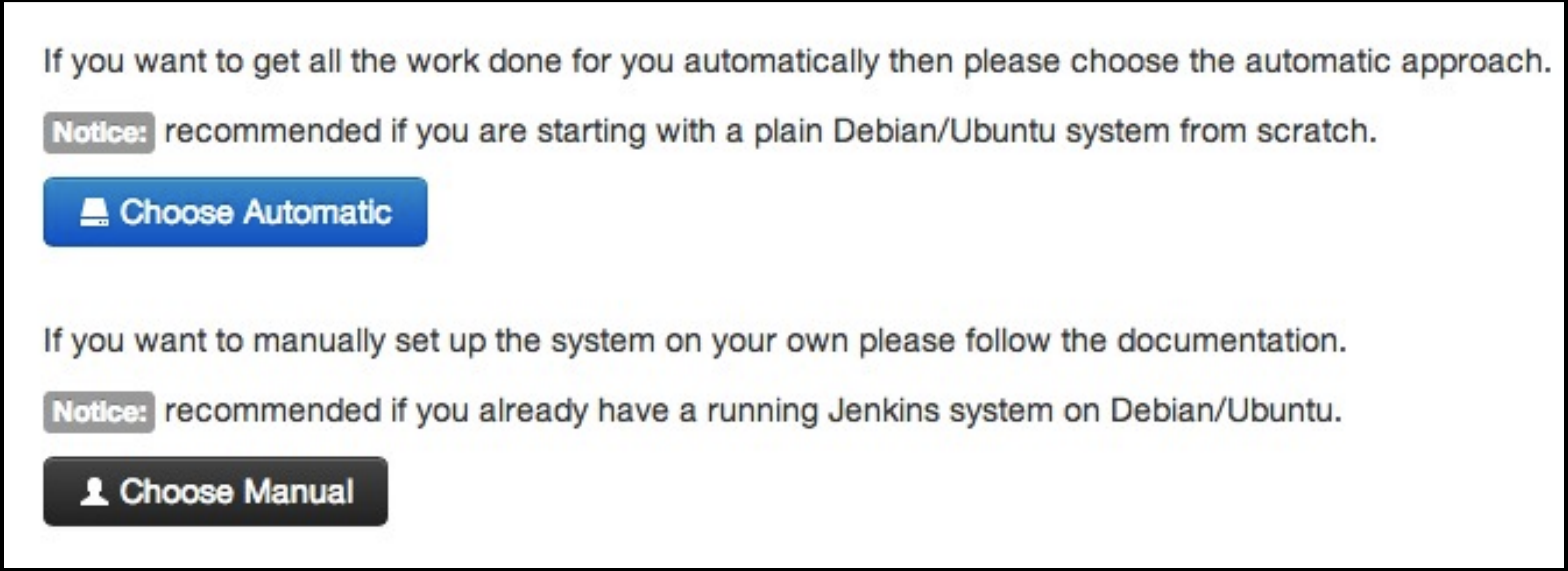
Bsp. für Dependencies


$$\begin{tikzcd}
 & \text{ngcpcfg-source} \arrow[dl] \arrow[d] \arrow[dr] & \\
\text{ngcpcfg-binaries} \arrow[d] \arrow[dr] & \text{ngcpcfg-test-source} & \text{ngcpcfg-unit-test} \\
\text{ngcpcfg-piuparts} & \text{ngcpcfg-repos} &
\end{tikzcd}$$

Bsp. für Dependencies



Deployment von j-d-g

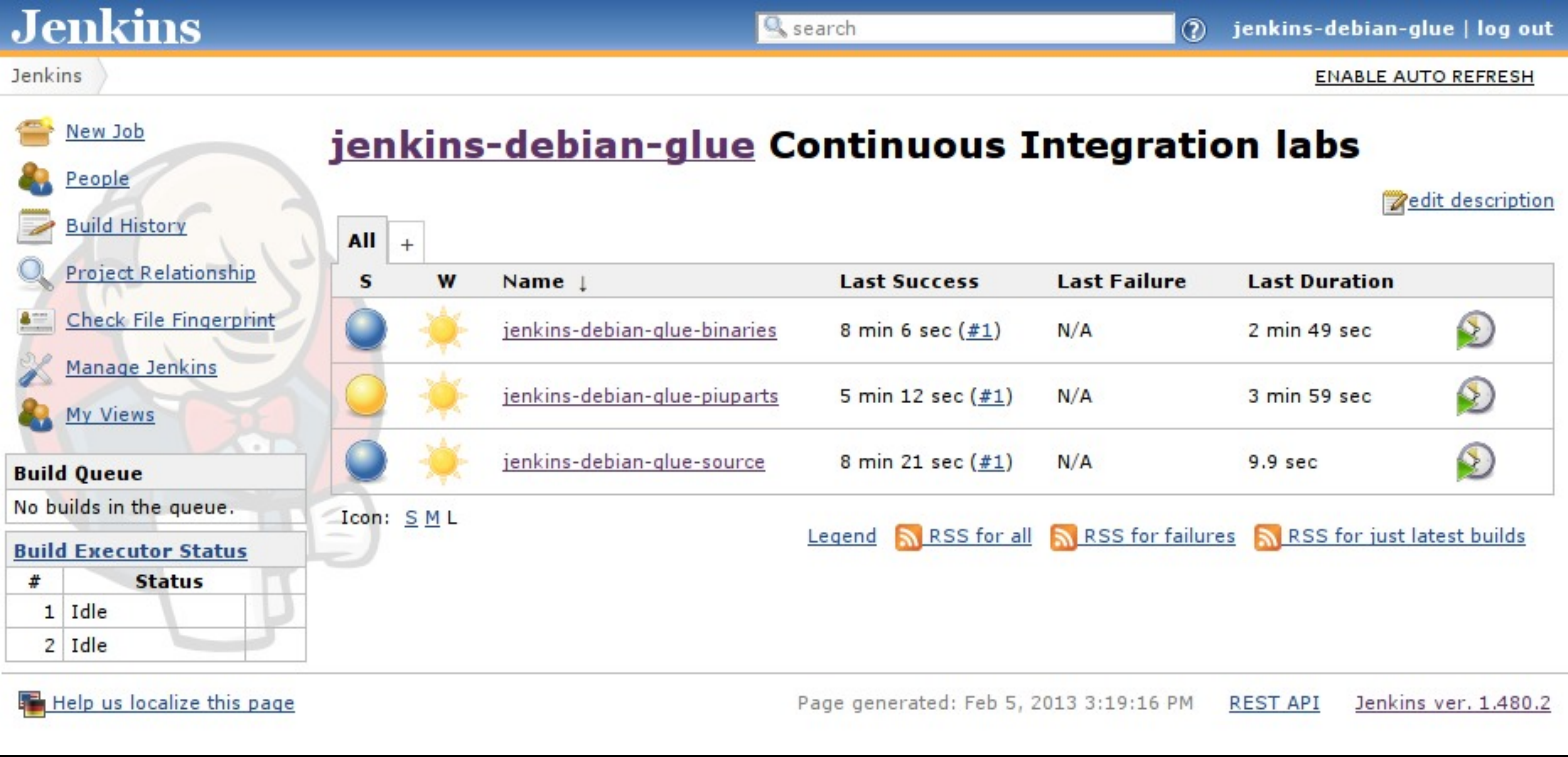

If you want to get all the work done for you automatically then please choose the automatic approach.

Notice: recommended if you are starting with a plain Debian/Ubuntu system from scratch.

If you want to manually set up the system on your own please follow the documentation.

Notice: recommended if you already have a running Jenkins system on Debian/Ubuntu.

- siehe http://jenkins-debian-glue.org/
- in <15 Minuten (auch auf EC2) mit minimalem Aufwand testbar


Jenkins | search | jenkins-debian-glue | log out

Jenkins | ENABLE AUTO REFRESH

New Job
People
Build History
Project Relationship
Check File Fingerprint
Manage Jenkins
My Views

jenkins-debian-glue Continuous Integration labs

edit description

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration
●	☀	jenkins-debian-glue-binaries	8 min 6 sec (#1)	N/A	2 min 49 sec
●	☀	jenkins-debian-glue-piuparts	5 min 12 sec (#1)	N/A	3 min 59 sec
●	☀	jenkins-debian-glue-source	8 min 21 sec (#1)	N/A	9.9 sec

Icon: S M L

Legend RSS for all RSS for failures RSS for just latest builds

Build Queue

No builds in the queue.

Build Executor Status

#	Status
1	Idle
2	Idle

Help us localize this page

Page generated: Feb 5, 2013 3:19:16 PM REST API Jenkins ver. 1.480.2

Default Deployment von jenkins-debian-glue

Weitere Einsatzbeispiele im Rechenzentrum

Puppet, Custom ISOs, Dokumentation,...

puppet-lint

- https://github.com/rodjek/puppet-lint
- Integration in VCS pre-commit-Hook (z.B. auch gemeinsam mit Syntax-Check)
  - https://gitorious.org/puppet-helpers/puppet-helpers

Puppet Environments

- z.B.:
  - development
  - staging
  - production
- https://puppetlabs.com/blog/git-workflow-and-puppet-environments/

Puppet Testing

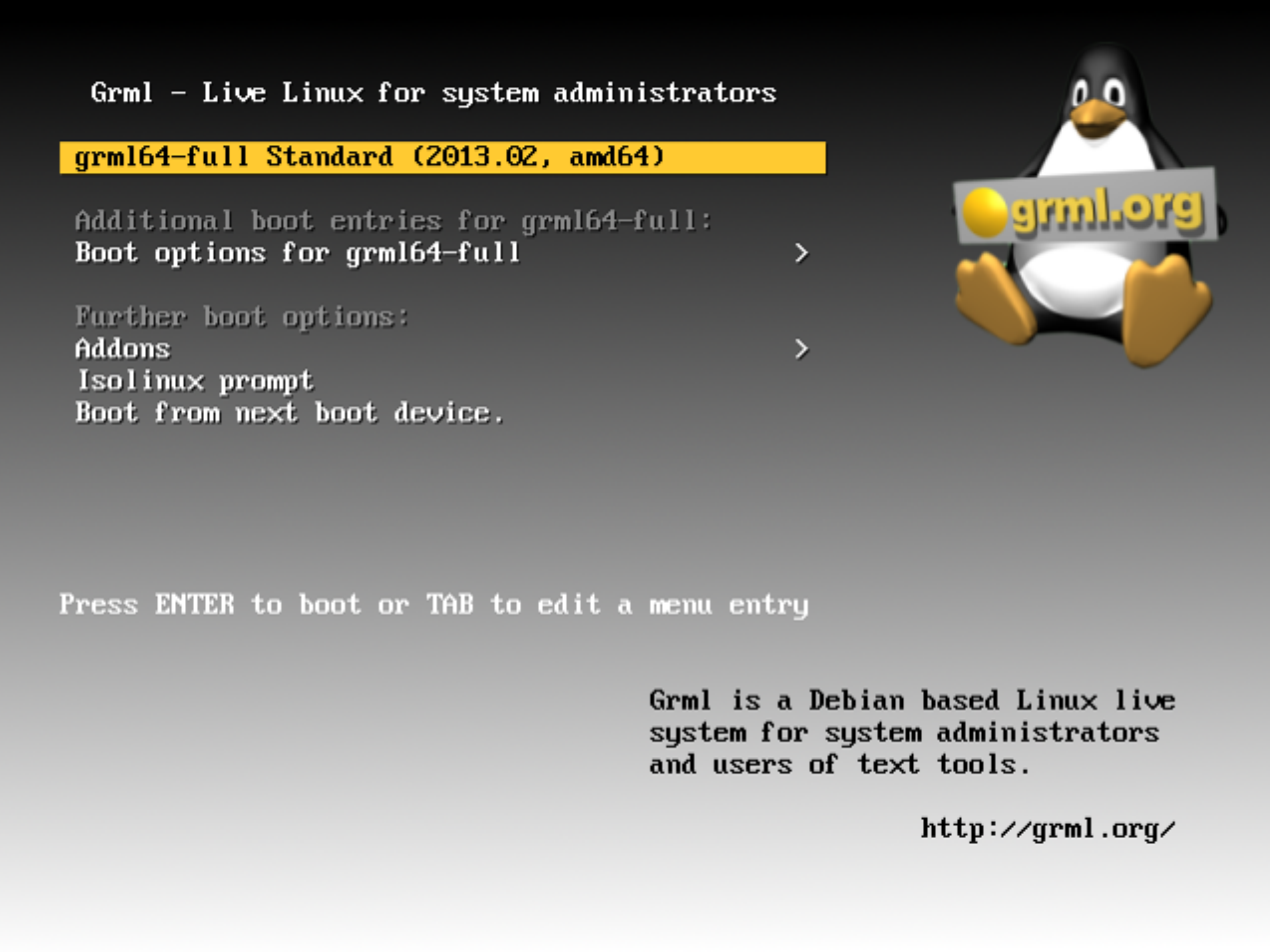
- RSpec-Puppet (http://rspec-puppet.com/)
- https://puppetlabs.com/blog/the-next-generation-of-puppet-module-testing/
- https://github.com/camptocamp/puppet-spec

Verifzieren vom System

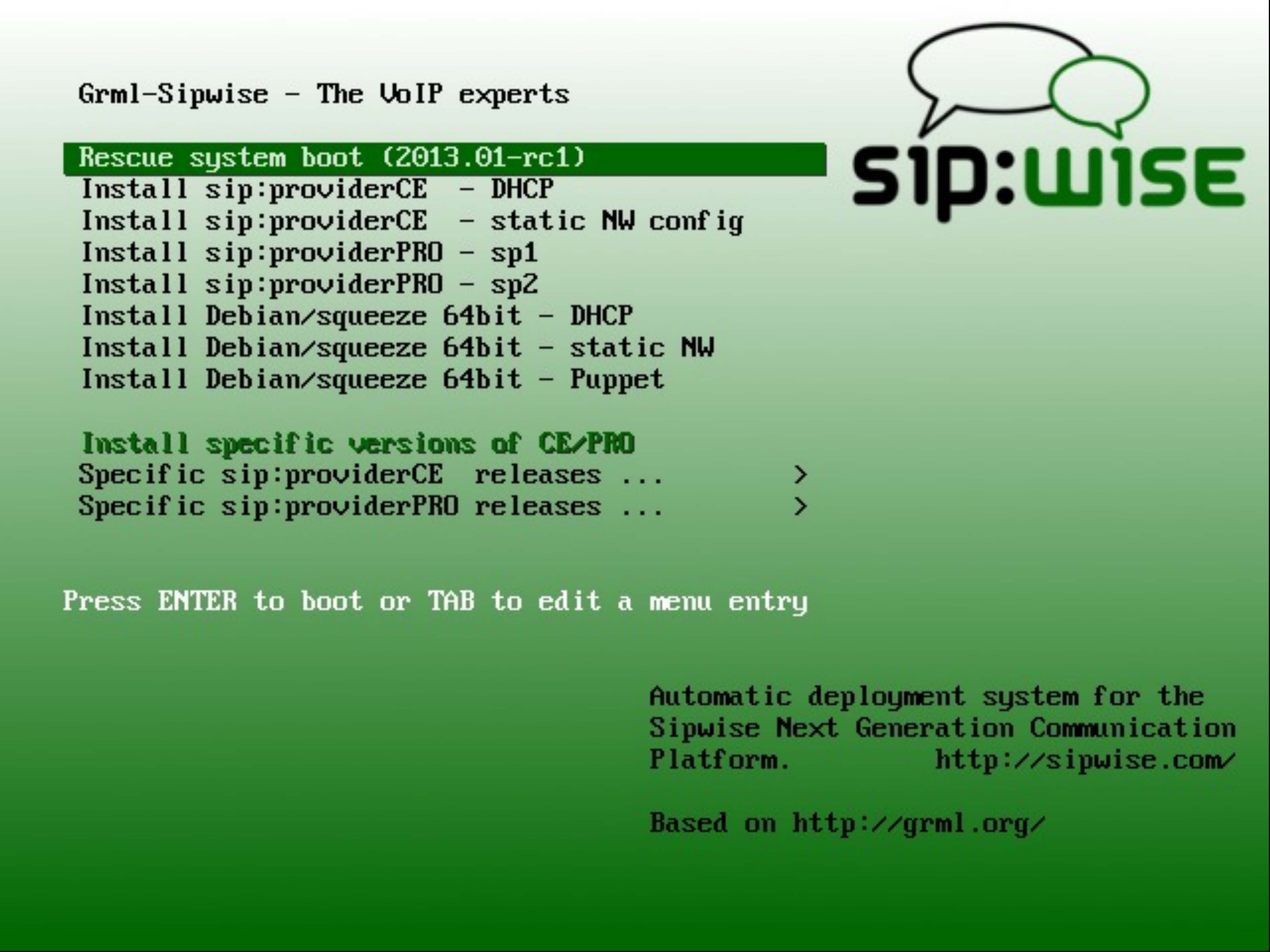
- RSpec tests
  - http://serverspec.org/
- mspectator
  - https://github.com/raphink/mspectator
- Tests::Server
  - http://search.cpan.org/dist/Test-Server/

Custom Grml ISOs

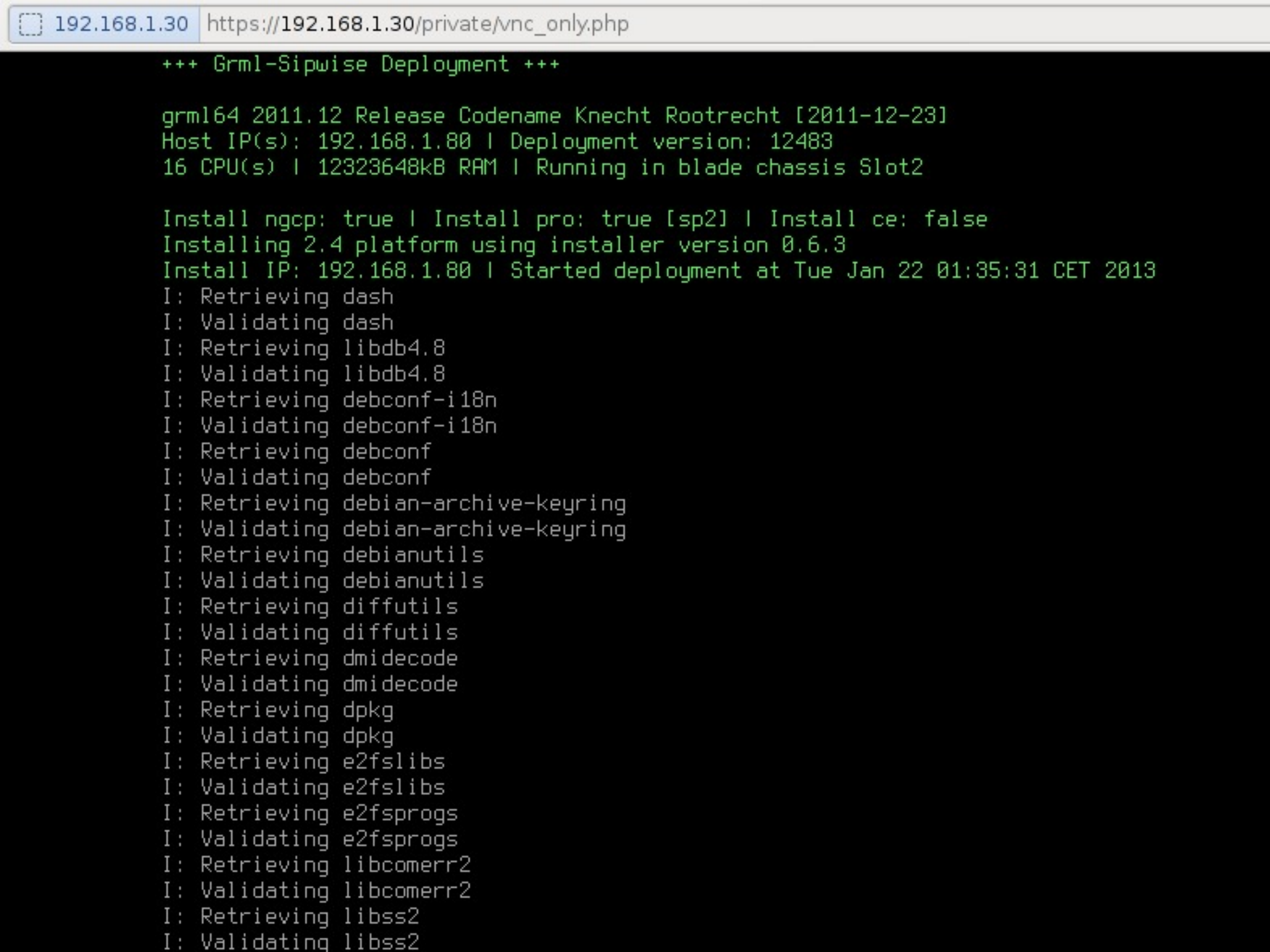
- grml-live.git (http://grml.org/grml-live/): templates/boot/isolinux/* anpassen
  - Bootoptionen (z.B. netscript=http://example.org/path/to/deployment.sh)
  - Bootsplash Layout (z.B. Firmenlogo)
- % sudo grml2iso -c templates -o custom.iso grml.iso
  - Teil von grml2usb (http://grml.org/grml2usb/)



```
Grml - Live Linux for system administrators

grml64-full Standard (2013.02, amd64)

Additional boot entries for grml64-full:
Boot options for grml64-full                >

Further boot options:
Addons                                      >
Isolinux prompt
Boot from next boot device.


Press ENTER to boot or TAB to edit a menu entry


                                   Grml is a Debian based Linux live
                                   system for system administrators
                                   and users of text tools.

                                                   http://grml.org/
```



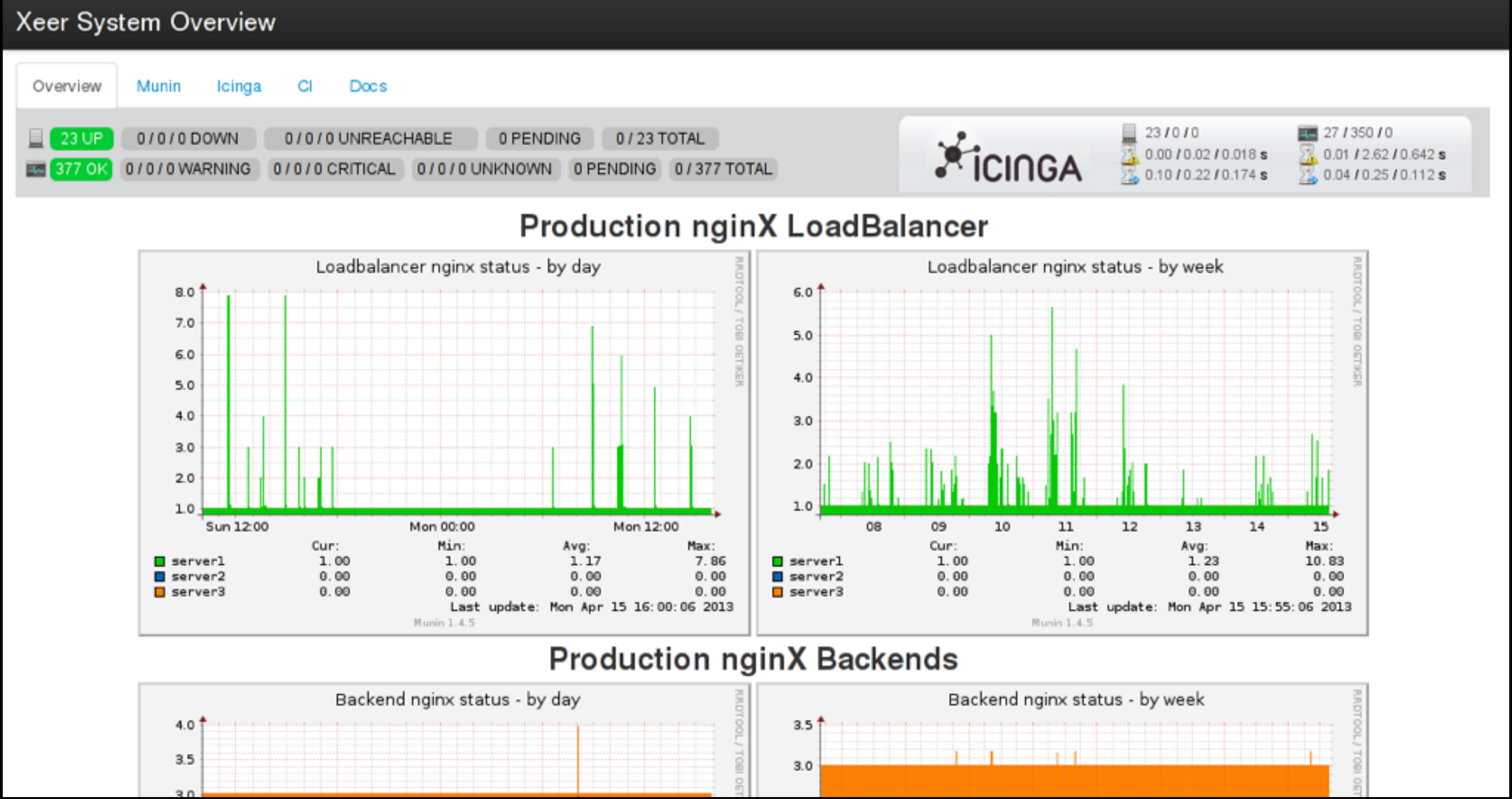
```
Grml-Sipwise - The VoIP experts

Rescue system boot (2013.01-rc1)
Install sip:providerCE  - DHCP
Install sip:providerCE  - static NW config
Install sip:providerPRO - sp1
Install sip:providerPRO - sp2
Install Debian/squeeze 64bit - DHCP
Install Debian/squeeze 64bit - static NW
Install Debian/squeeze 64bit - Puppet

Install specific versions of CE/PRO
Specific sip:providerCE  releases ...      >
Specific sip:providerPRO releases ...      >

Press ENTER to boot or TAB to edit a menu entry

                          Automatic deployment system for the
                          Sipwise Next Generation Communication
                          Platform.          http://sipwise.com/

                          Based on http://grml.org/
```



```
+++ Grml-Sipwise Deployment +++

grml64 2011.12 Release Codename Knecht Rootrecht [2011-12-23]
Host IP(s): 192.168.1.80 | Deployment version: 12483
16 CPU(s) | 12323648kB RAM | Running in blade chassis Slot2

Install ngcp: true | Install pro: true [sp2] | Install ce: false
Installing 2.4 platform using installer version 0.6.3
Install IP: 192.168.1.80 | Started deployment at Tue Jan 22 01:35:31 CET 2013
I: Retrieving dash
I: Validating dash
I: Retrieving libdb4.8
I: Validating libdb4.8
I: Retrieving debconf-i18n
I: Validating debconf-i18n
I: Retrieving debconf
I: Validating debconf
I: Retrieving debian-archive-keyring
I: Validating debian-archive-keyring
I: Retrieving debianutils
I: Validating debianutils
I: Retrieving diffutils
I: Validating diffutils
I: Retrieving dmidecode
I: Validating dmidecode
I: Retrieving dpkg
I: Validating dpkg
I: Retrieving e2fslibs
I: Validating e2fslibs
I: Retrieving e2fsprogs
I: Validating e2fsprogs
I: Retrieving libcomerr2
I: Validating libcomerr2
I: Retrieving libss2
I: Validating libss2
```

Admin-Doku

- z.B. mit Sphinx (reStructured TeXt, Such-Feature in HTML-Ausgabe inkludiert!)
- automatisch via Jenkins-Job bauen aus VCS
  - immer aktuelle Dokumentation für alle zugänglich (HTML, PDF,...)



23 UP | 0 / 0 / 0 DOWN | 0 / 0 / 0 UNREACHABLE | 0 PENDING | 0 / 23 TOTAL

377 OK | 0 / 0 / 0 WARNING | 0 / 0 / 0 CRITICAL | 0 / 0 / 0 UNKNOWN | 0 PENDING | 0 / 377 TOTAL

```
23 / 0 / 0                    27 / 350 / 0
0.00 / 0.02 / 0.018 s         0.01 / 2.62 / 0.642 s
0.10 / 0.22 / 0.174 s         0.04 / 0.25 / 0.112 s
```

Production nginX LoadBalancer

Loadbalancer nginx status - by day

```
              Cur:        Min:        Avg:        Max:
🟩 server1    1.00        1.00        1.17        7.86
🟦 server2    0.00        0.00        0.00        0.00
🟧 server3    0.00        0.00        0.00        0.00
                       Last update:  Mon Apr 15 16:00:06 2013
```

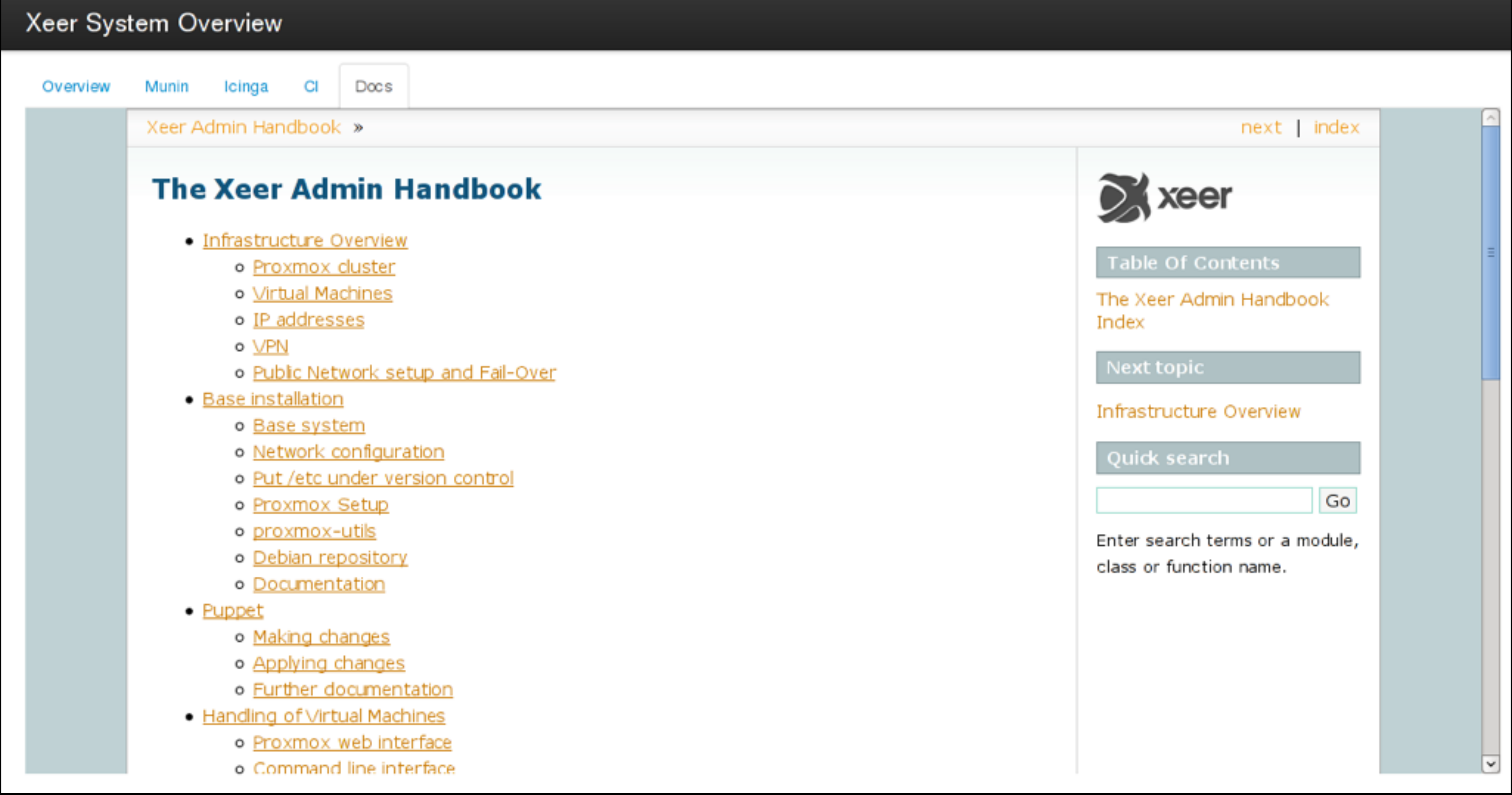
Loadbalancer nginx status - by week

```
              Cur:        Min:        Avg:        Max:
🟩 server1    1.00        1.00        1.23        10.83
🟦 server2    0.00        0.00        0.00        0.00
🟧 server3    0.00        0.00        0.00        0.00
                       Last update:  Mon Apr 15 15:55:06 2013
```

Production nginX Backends

Backend nginx status - by day

Backend nginx status - by week


Xeer System Overview

Overview | Munin | Icinga | CI | Docs

Xeer Admin Handbook » next | index

The Xeer Admin Handbook

xeer

Table Of Contents

The Xeer Admin Handbook
Index

Next topic

Infrastructure Overview

Quick search

Go

Enter search terms or a module, class or function name.

Vagrant/Veewee

- Vagrant base boxes + VMs mit Veewee automatisch bauen
  - https://github.com/jedi4ever/veewee
- Testen/Entwicklung mit Vagrant
  - http://www.vagrantup.com/
  - Entwickler können Puppet/Chef/...-Rezepte schreiben und testen!

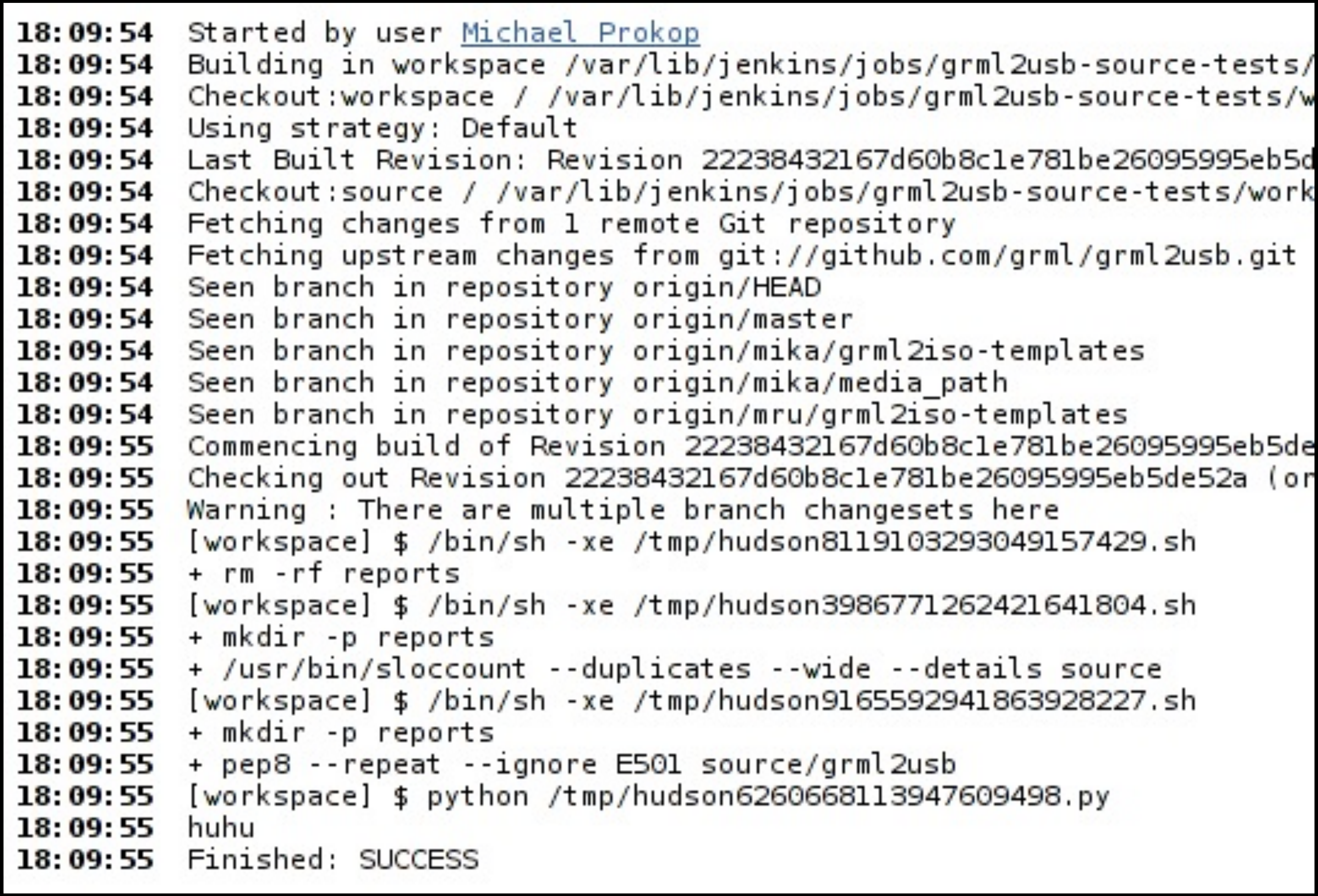
Best Practices

... oder nicht jeder muss die gleichen Schmerzen erleben

Automatisieren was weh tut

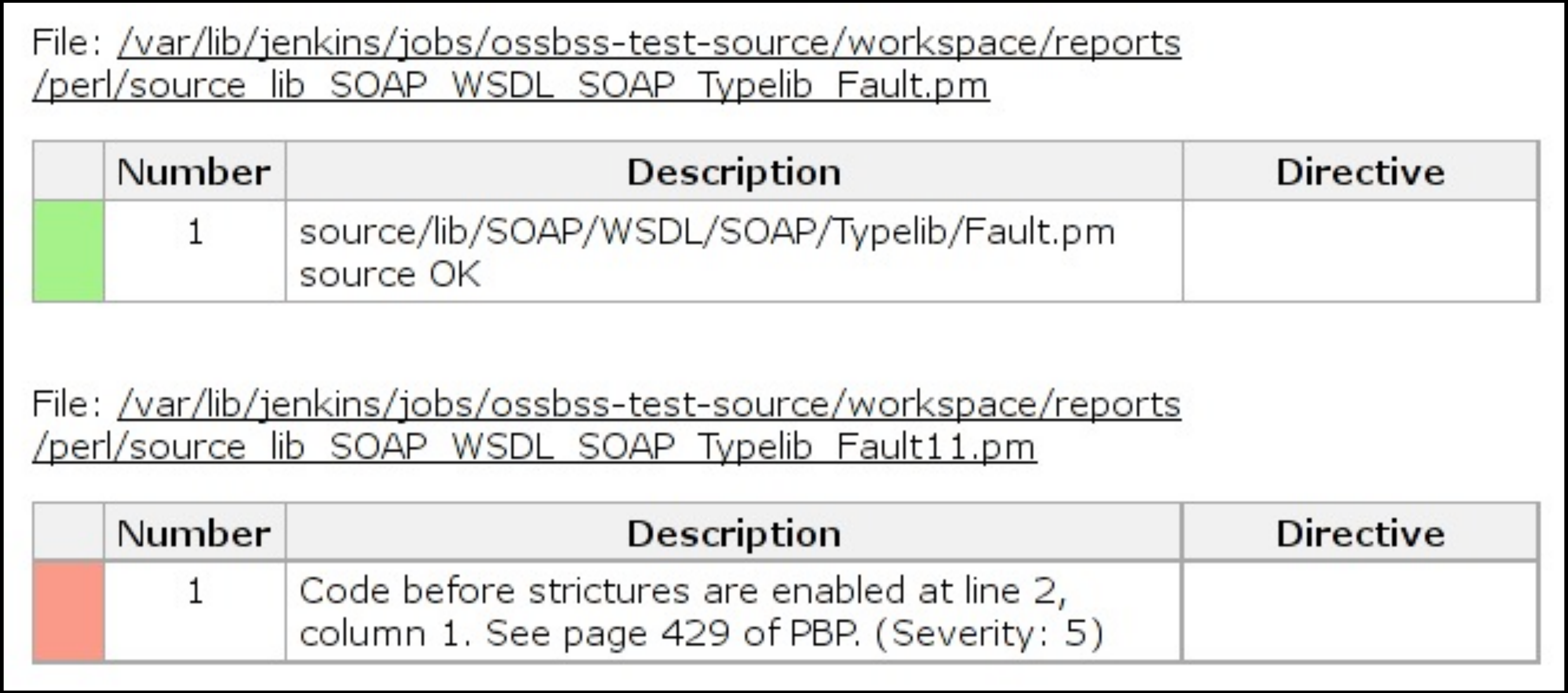
Quelle: http://www.flickr.com/photos/ctrl-l/3579073698/

Timestamper Plugin



```
18:09:54  Started by user Michael Prokop
18:09:54  Building in workspace /var/lib/jenkins/jobs/grml2usb-source-tests/
18:09:54  Checkout:workspace / /var/lib/jenkins/jobs/grml2usb-source-tests/w
18:09:54  Using strategy: Default
18:09:54  Last Built Revision: Revision 22238432167d60b8c1e781be26095995eb5d
18:09:54  Checkout:source / /var/lib/jenkins/jobs/grml2usb-source-tests/work
18:09:54  Fetching changes from 1 remote Git repository
18:09:54  Fetching upstream changes from git://github.com/grml/grml2usb.git
18:09:54  Seen branch in repository origin/HEAD
18:09:54  Seen branch in repository origin/master
18:09:54  Seen branch in repository origin/mika/grml2iso-templates
18:09:54  Seen branch in repository origin/mika/media_path
18:09:54  Seen branch in repository origin/mru/grml2iso-templates
18:09:55  Commencing build of Revision 22238432167d60b8c1e781be26095995eb5de
18:09:55  Checking out Revision 22238432167d60b8c1e781be26095995eb5de52a (or
18:09:55  Warning : There are multiple branch changesets here
18:09:55  [workspace] $ /bin/sh -xe /tmp/hudson8119103293049157429.sh
18:09:55  + rm -rf reports
18:09:55  [workspace] $ /bin/sh -xe /tmp/hudson3986771262421641804.sh
18:09:55  + mkdir -p reports
18:09:55  + /usr/bin/sloccount --duplicates --wide --details source
18:09:55  [workspace] $ /bin/sh -xe /tmp/hudson9165592941863928227.sh
18:09:55  + mkdir -p reports
18:09:55  + pep8 --repeat --ignore E501 source/grml2usb
18:09:55  [workspace] $ python /tmp/hudson6260668113947609498.py
18:09:55  huhu
18:09:55  Finished: SUCCESS
```

Test Anything Protocol + Plugin



File: /var/lib/jenkins/jobs/ossbss-test-source/workspace/reports/perl/source_lib_SOAP_WSDL_SOAP_Typelib_Fault.pm

	Number	Description	Directive
🟩	1	source/lib/SOAP/WSDL/SOAP/Typelib/Fault.pm source OK	

File: /var/lib/jenkins/jobs/ossbss-test-source/workspace/reports/perl/source_lib_SOAP_WSDL_SOAP_Typelib_Fault11.pm

	Number	Description	Directive
🟥	1	Code before strictures are enabled at line 2, column 1. See page 429 of PBP. (Severity: 5)	

Bruce Schneier Plugin

... knows Alice and Bob's shared secret.

“discard old builds”

Quelle: http://www.flickr.com/photos/epsos/5575089139/

schnelle
Hardware nutzen

Entwicklerzeit ist
teuer

Homogenität

Quelle: http://www.flickr.com/photos/baggis/226567107/

Builds triggern und *nicht* pollen

Jenkins Jobs Handling

- Erstellen von Jobs automatisieren
- Configs in VCS speichern
  - https://wiki.jenkins-ci.org/display/JENKINS/SCM+Sync+configuration+plugin
- jenkins-job-builder & CO
  - https://github.com/openstack-infra/jenkins-job-builder
  - ... viele weitere Tools: https://gist.github.com/mika/5237127

Externe Abhängigkeiten beseitigen

Beispiele was schiefgehen kann (BTDT):

- Github
- PyPI
- RubyGems
- Percona Repository
- ....

There are only two hard problems in Computer Science: cache invalidation, naming things and off-by-one errors.

Jenkins Master als Controlinstanz + Jenkins Slaves fürs Bauen

Dashboards

- View auf Repository
- View auf Build-Status
- Frontend für Bauen von Releases
- ...

Low-Hanging Fruits für Speedup

- tmpfs
- eatmydata
- lokaler Package-Mirror

Puppet mit mcollective

mcollective ftw!

```
% mco rpc package apt_update
% mco package update \
    jenkins-debian-glue \
    -W /jenkins-slave/
```

Achtung vor Catch-22

1) CI-Upgrade geht nicht wegen Bug,
Bugfix von Plugin hängt aber
von neuer CI-Version ab
2) Buildskripte die unter dem CI-System
stecken kommen vom CI-System selbst

....

Wartungsfenster auch für CI-Umgebung schaffen

Recap

- Keine Angst vor Jenkins
- Verfügbare Jenkins-Plugins anschauen
- Automatisierung (Paketmanagement, Configuration Management,...)
- Kein manuelles SSH (fabric, mcollective,....)
- Tests schreiben
- Dashboards

Fragen || Wünsche?

OSDC.de
OPEN SOURCE DATA CENTER CONFERENCE
17.- 18. APRIL 2013 | NUREMBERG

@mikagrml

mika @ github

michael-prokop.at/blog/

grml-solutions.com