

Wait, Jenkins can do  
THAT?!

Stefan Rotsch, AOE GmbH

# Jenkins

- Continuously **build** and **test** software projects
- **Monitor** execution of repeated jobs



# Jenkins

- Written in **Java**
- Supports various **SCMs**
- Executes Ant, Maven,  
**Shell**, Windows Batch
- Extendable
- Free Software



# Installing Jenkins

# Installation

```
# Add PGP key
$ wget -q -O - https://jenkins-ci.org/
debian/jenkins-ci.org.key | sudo apt-key
add -
```

# Installation

```
# Add PGP key
# Add Jenkins APT repository
$ sudo sh -c 'echo deb http://pkg.jenkins-  /
ci.org/debian binary/ > /etc/apt/
sources.list.d/jenkins.list'
```

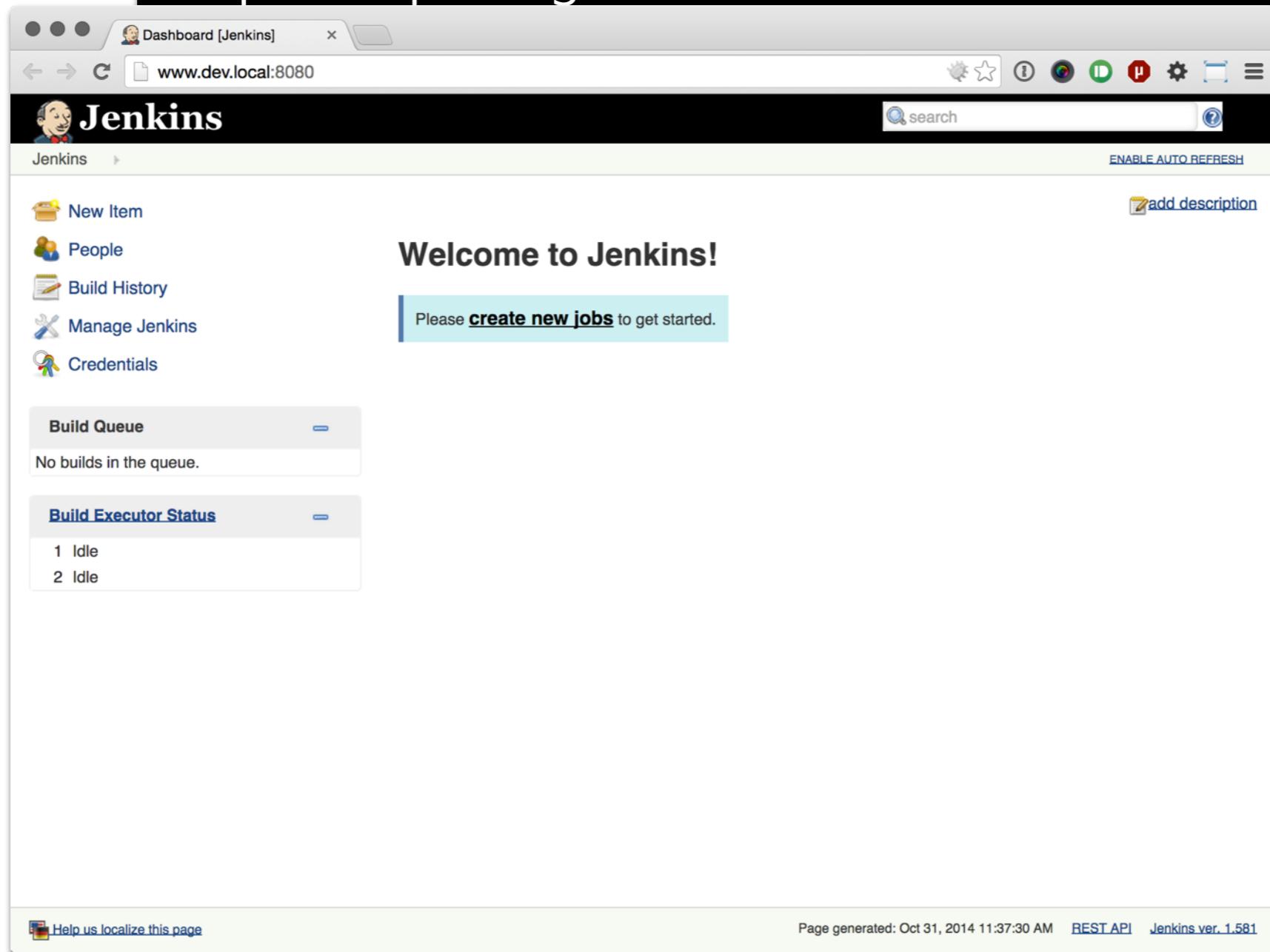
# Installation

```
# Add PGP key
# Add Jenkins APT repository
# Update package list and install
$ sudo apt-get update
$ sudo apt-get install jenkins
```

# Installation

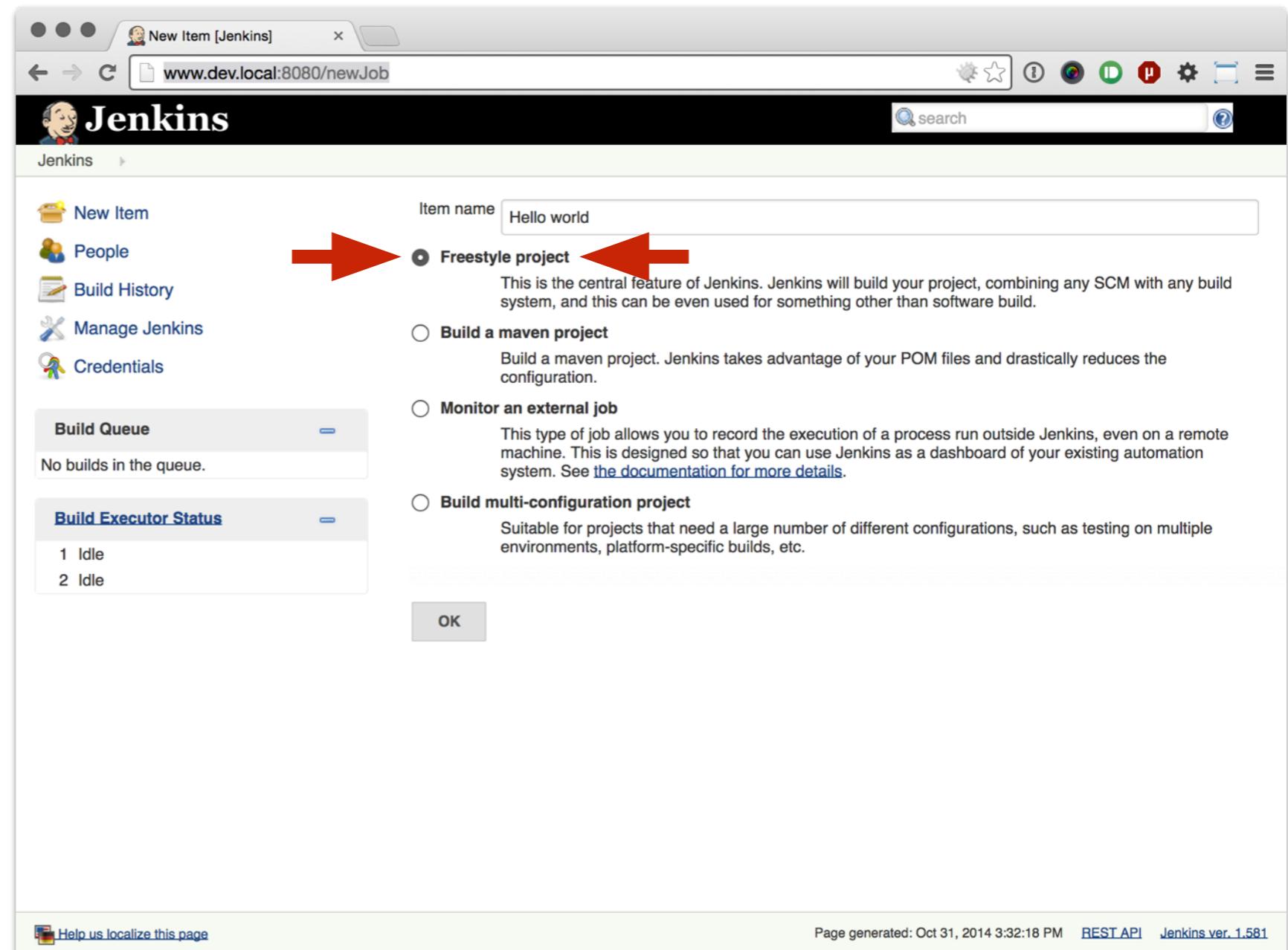
```
# Add PGP key  
# Add Jenkins APT repository  
# Update package list and install
```

1S- ✓

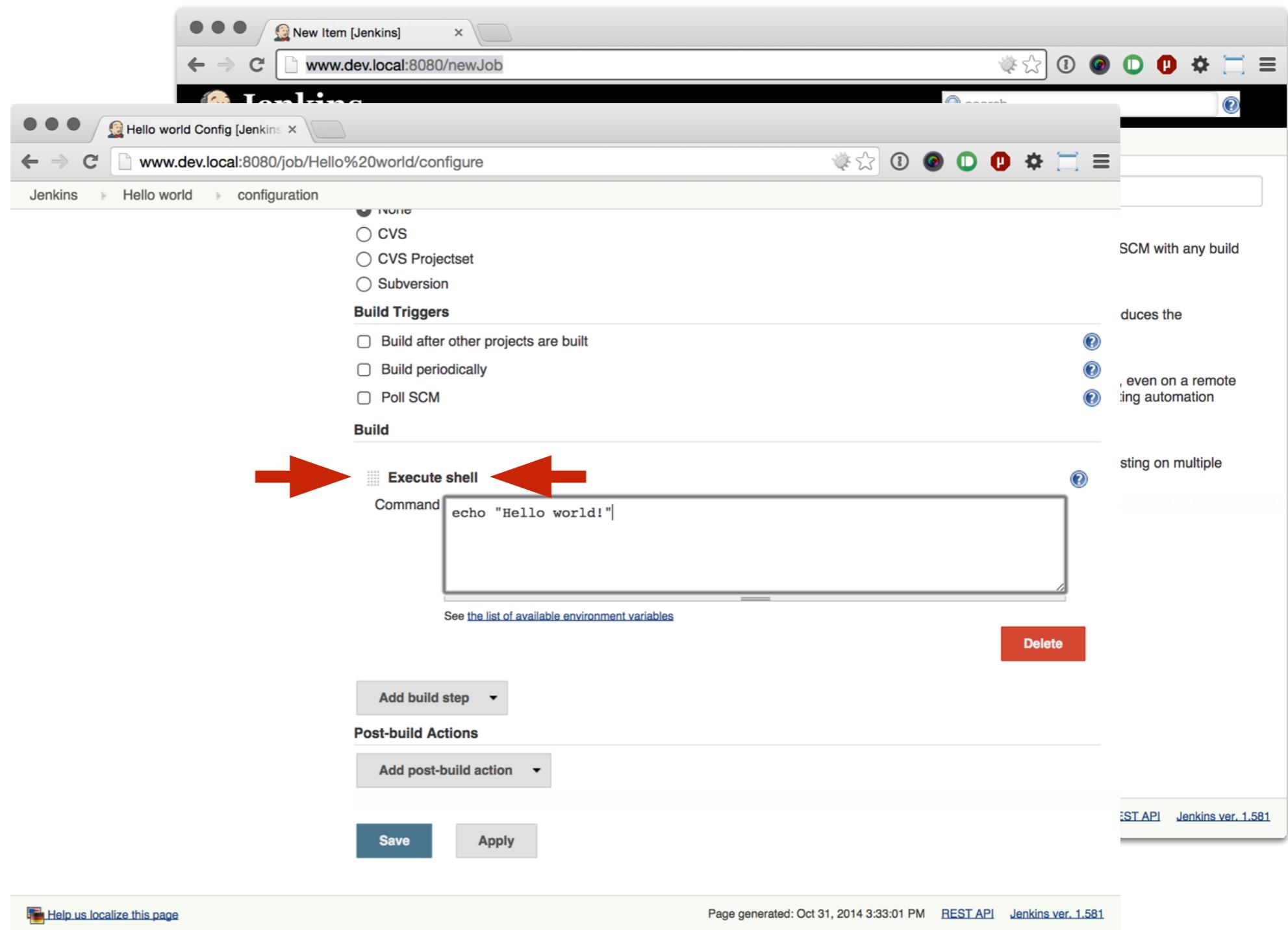


# Execute shell commands

# Shell commands



# Shell commands



The screenshot shows the Jenkins configuration page for a job named 'Hello world'. The 'Build' section is expanded, showing an 'Execute shell' step. A red arrow points to the 'Execute shell' label, and another red arrow points to the command input field, which contains the text 'echo "Hello world!"'. The Jenkins interface includes a top navigation bar with tabs for 'New Item [Jenkins]' and 'Hello world Config [Jenkins]'. The main content area shows various configuration options like SCM and build triggers, with descriptive text and help icons. At the bottom, there are 'Save' and 'Apply' buttons, and a footer with links to 'Help us localize this page', 'Page generated: Oct 31, 2014 3:33:01 PM', 'REST API', and 'Jenkins ver. 1.581'.

None

CVS

CVS Projectset

Subversion

Build Triggers

Build after other projects are built

Build periodically

Poll SCM

Build

Execute shell

Command

```
echo "Hello world!"
```

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

Delete

Add build step

Post-build Actions

Add post-build action

Save

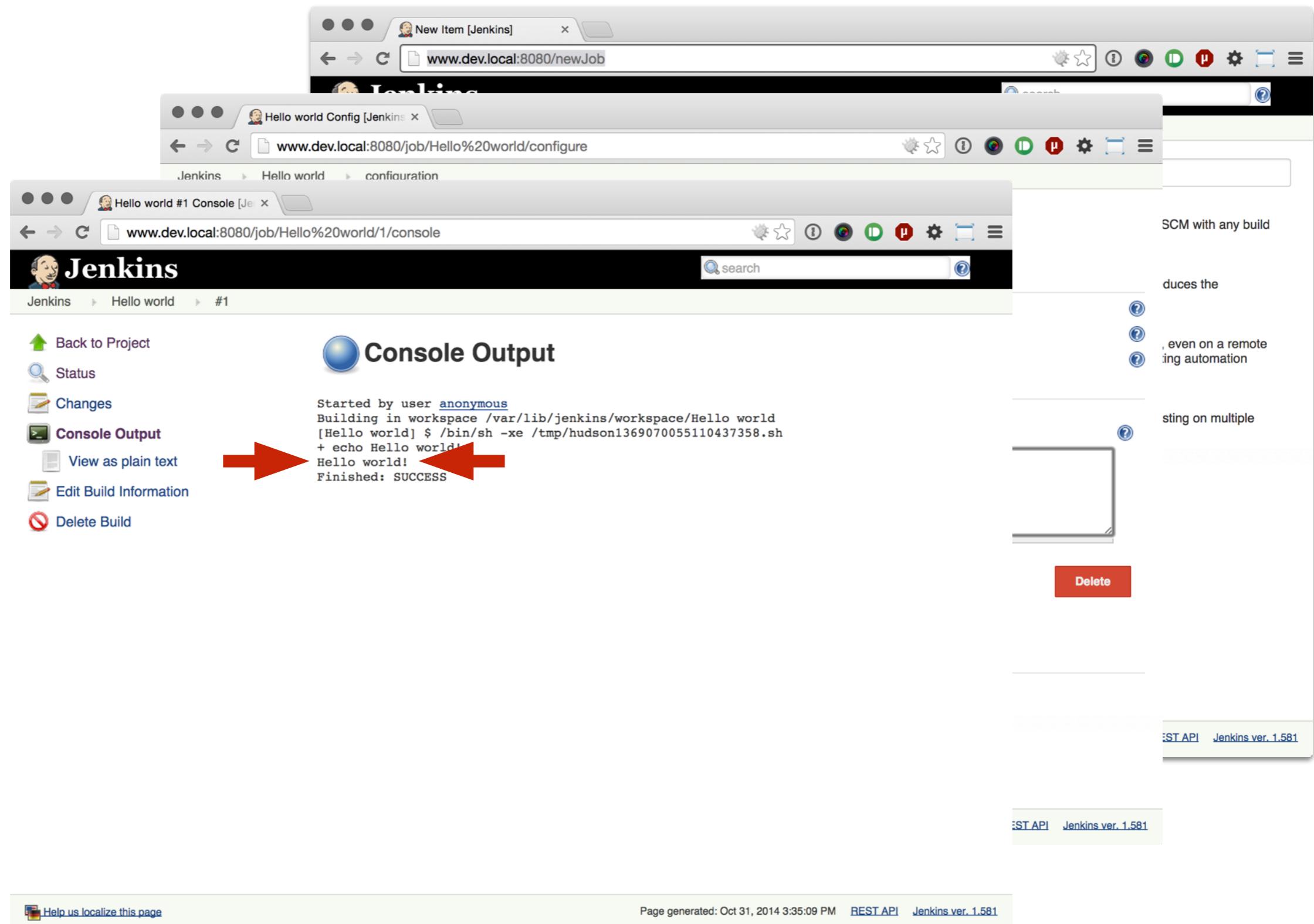
Apply

Help us localize this page

Page generated: Oct 31, 2014 3:33:01 PM

REST API Jenkins ver. 1.581

# Shell commands



The screenshot shows a web browser with three tabs open, all pointing to `www.dev.local:8080`:

- New Item [Jenkins]**: Shows the Jenkins dashboard with a 'Hello world' job.
- Hello world Config [Jenkins]**: Shows the configuration page for the 'Hello world' job.
- Hello world #1 Console [Jenkins]**: Shows the build console output for the 'Hello world' job.

The 'Hello world #1 Console' tab is active. The page title is 'Console Output'. The content of the console output is:

```
Started by user anonymous
Building in workspace /var/lib/jenkins/workspace/Hello world
[Hello world] $ /bin/sh -xe /tmp/hudson1369070055110437358.sh
+ echo Hello world!
Hello world!
Finished: SUCCESS
```

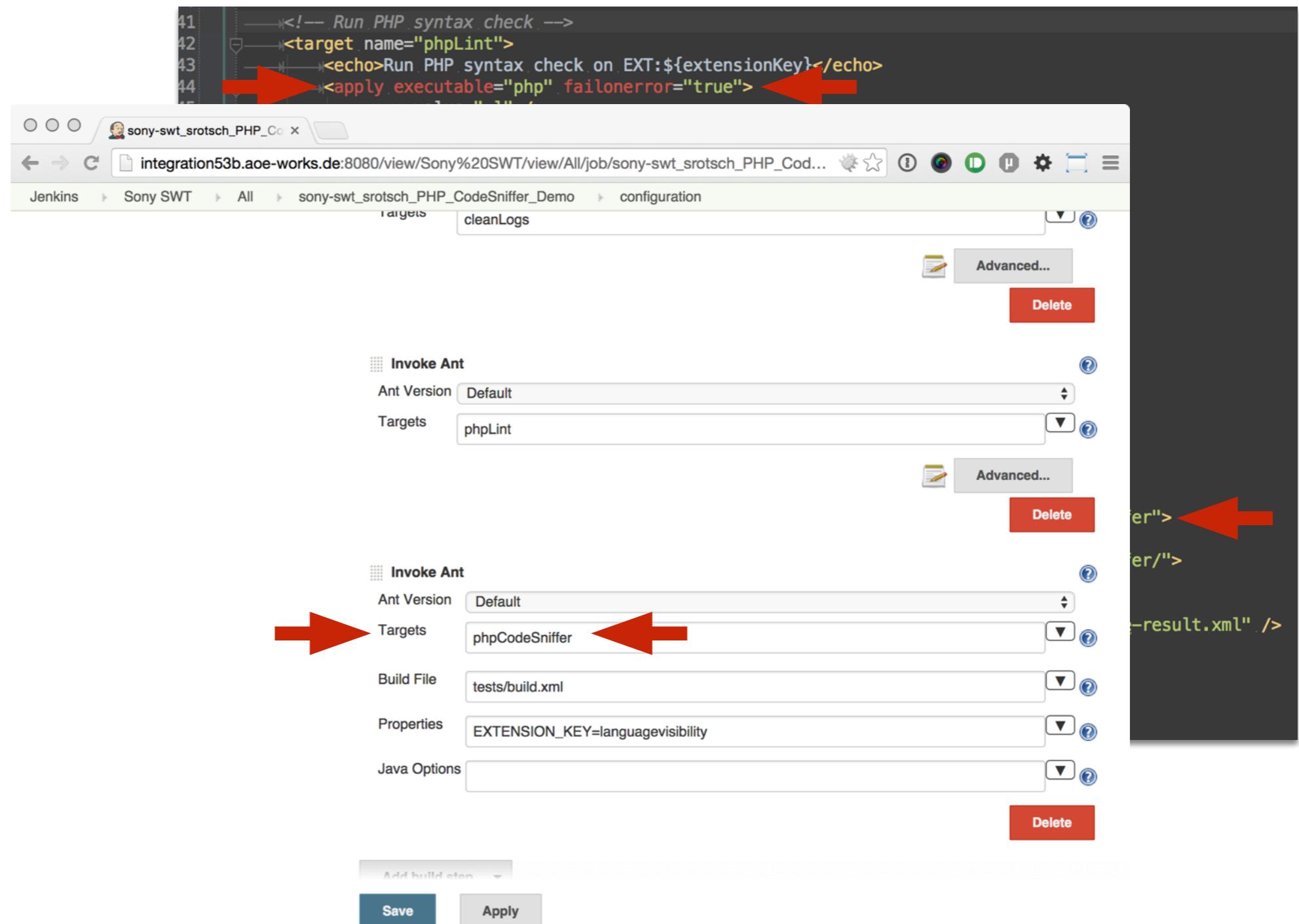
A large red arrow points to the line 'Hello world!'. The Jenkins logo is visible in the top left of the main content area.

Inspect code with  
PHP\_CodeSniffer

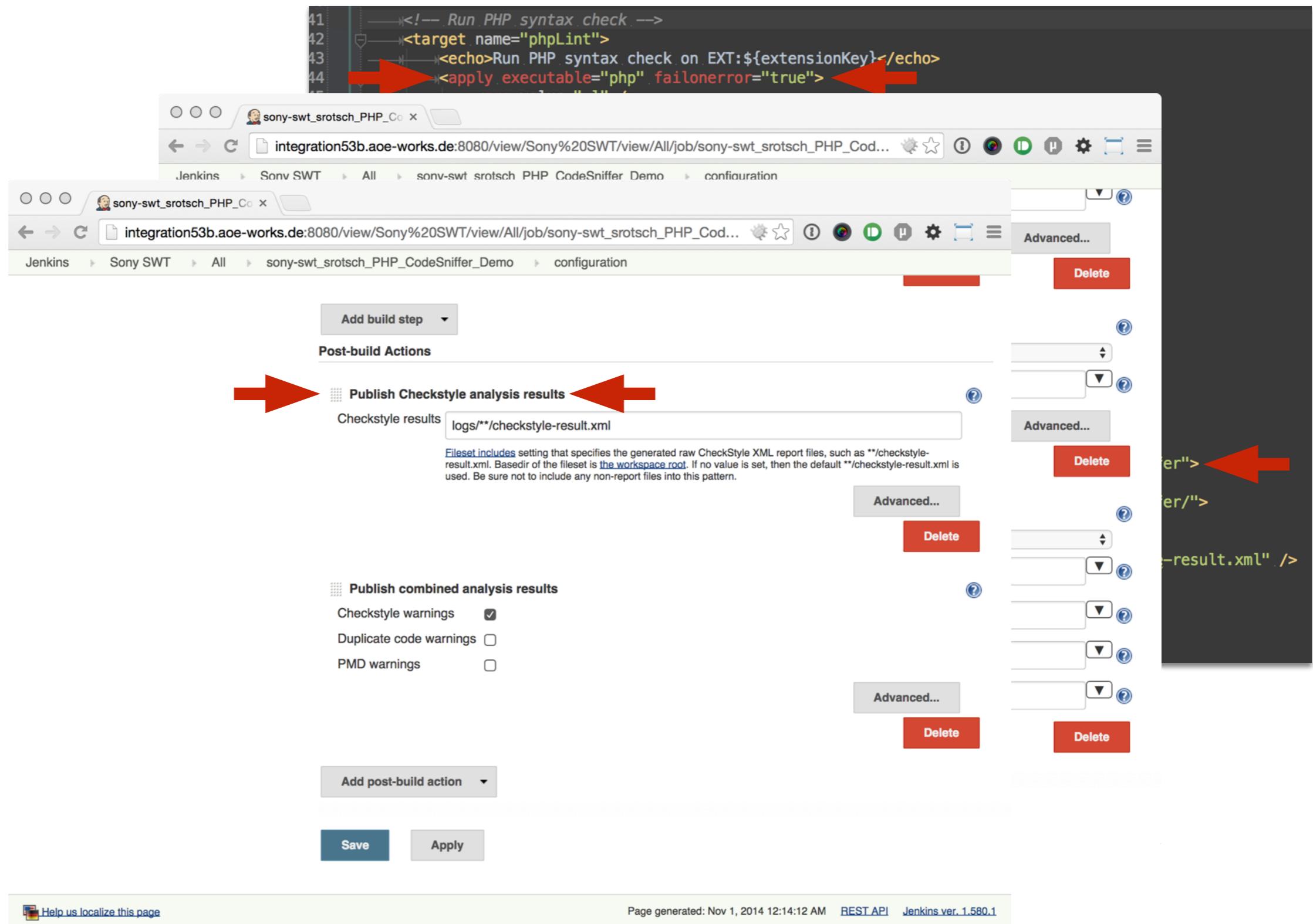
# PHP\_CodeSniffer Setup

```
41 	<!-- Run PHP syntax check --&gt;
42 	&lt;target name="phpLint"&gt;
43 	&lt;echo&gt;Run PHP syntax check on EXT:${extensionKey}&lt;/echo&gt;
44 	&lt;apply executable="php" failonerror="true"&gt; <img alt="Red arrow pointing to the apply tag" data-bbox="365 235 395 265"/>
45 	<arg value="-l" />
46 	<fileset dir="${workingDirectory}/extensions/${extensionKey}">
47 	<include name="**/*.php" />
48 	</fileset>
49 	</apply>
50 	<echo>done</echo>
51 	</target>
52
53 	<!-- Install PHP CodeSniffer --&gt;
54 	&lt;target name="composerInstallPhpCodeSniffer"&gt;
55 	&lt;echo&gt;Install PHP CodeSniffer&lt;/echo&gt;
56 	&lt;exec executable="composer" dir="${workingDirectory}/tests/PHPCodeSniffer/"&gt;
57 	&lt;arg value="install" /&gt;
58 	&lt;arg value="--quiet" /&gt;
59 	&lt;/exec&gt;
60 	&lt;echo&gt;done&lt;/echo&gt;
61 	&lt;/target&gt;
62
63 	<!-- Run PHP CodeSniffer --&gt;
64 	&lt;target name="phpCodeSniffer" depends="createLogDirectory,composerInstallPhpCodeSniffer"&gt; <img alt="Red arrow pointing to the target tag" data-bbox="365 565 395 595"/>
65 	<echo>Run PHP CodeSniffer on EXT:${extensionKey}</echo>
66 	<exec executable=".vendor/bin/phpcs" dir="${workingDirectory}/tests/PHPCodeSniffer/">
67 	<arg value="--standard=TYPO3CMS" />
68 	<arg value="--report=checkstyle" />
69 	<arg value="--report-file=${workingDirectory}/logs/${extensionKey}/checkstyle-result.xml" />
70 	<arg value="${ignorePattern}" />
71 	<arg path="${workingDirectory}/extensions/${extensionKey}" />
72 	</exec>
73 	<echo>done</echo>
74 	</target>
```

# PHP\_CodeSniffer Setup



# PHP\_CodeSniffer Setup



The screenshot shows the Jenkins configuration for a job named "sony-swt\_srotsch\_PHP\_CodeSniffer Demo". The configuration page displays a build script and a "Post-build Actions" section.

**Build Script (Top):**

```
41:  <!-- Run PHP syntax check -->
42:  <target name="phpLint">
43:  <echo>Run PHP syntax check on EXT:${extensionKey}</echo>
44:  <apply executable="php" failonerror="true">
45:  <!-- Run PHP syntax check -->
```

**Post-build Actions (Bottom):**

- Publish Checkstyle analysis results**
  - Checkstyle results: logs/\*\*/checkstyle-result.xml
  - Fileset includes: logs/\*\*/checkstyle-result.xml
  - Advanced...
  - Delete
- Publish combined analysis results**
  - Checkstyle warnings:
  - Duplicate code warnings:
  - PMD warnings:
  - Advanced...
  - Delete

**Buttons at the bottom:**

- Add build step
- Add post-build action
- Save
- Apply

**Background (Right):**

```
<!-- Run PHP syntax check -->
<target name="phpLint">
<echo>Run PHP syntax check on EXT:${extensionKey}</echo>
<apply executable="php" failonerror="true">
<!-- Run PHP syntax check -->
```

# PHP\_CodeSniffer Results

Jenkins

integration53b.aoe-works.de:8080/view/Sony%20SWT/view/All/job/sony-swt\_srotsch\_PHP\_Cod...

Stefan Rotsch | log out

Back to Project

Status

Changes

Console Output

Edit Build Information

Delete Build

Environment Variables

Git Build Data

Checkstyle Warnings

Static Analysis Warnings

Previous Build

# CheckStyle Result

## Warnings Trend

All Warnings	New Warnings	Fixed Warnings
725	0	0

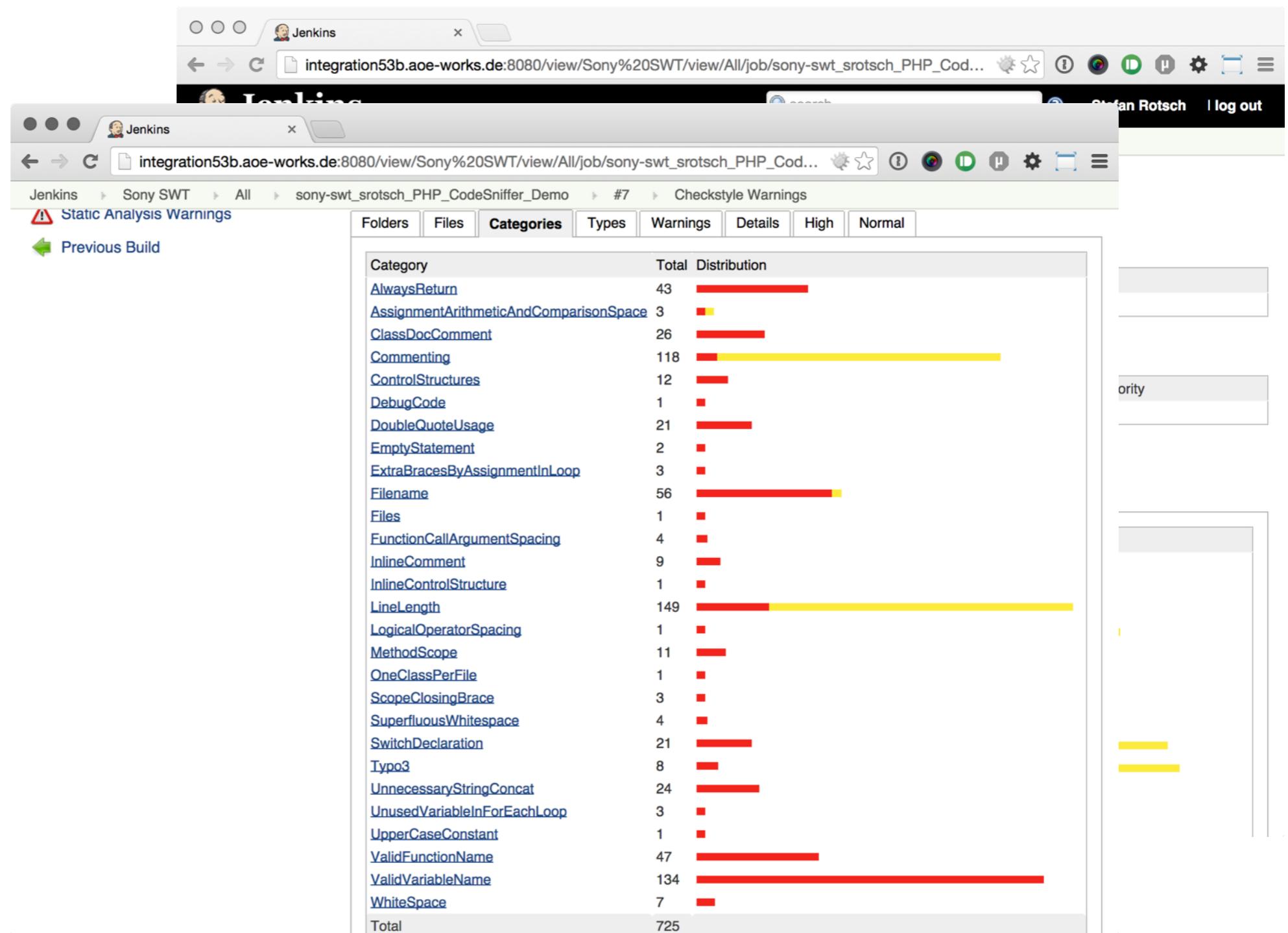
## Summary

Total	High Priority	Normal Priority	Low Priority
725	490	235	0

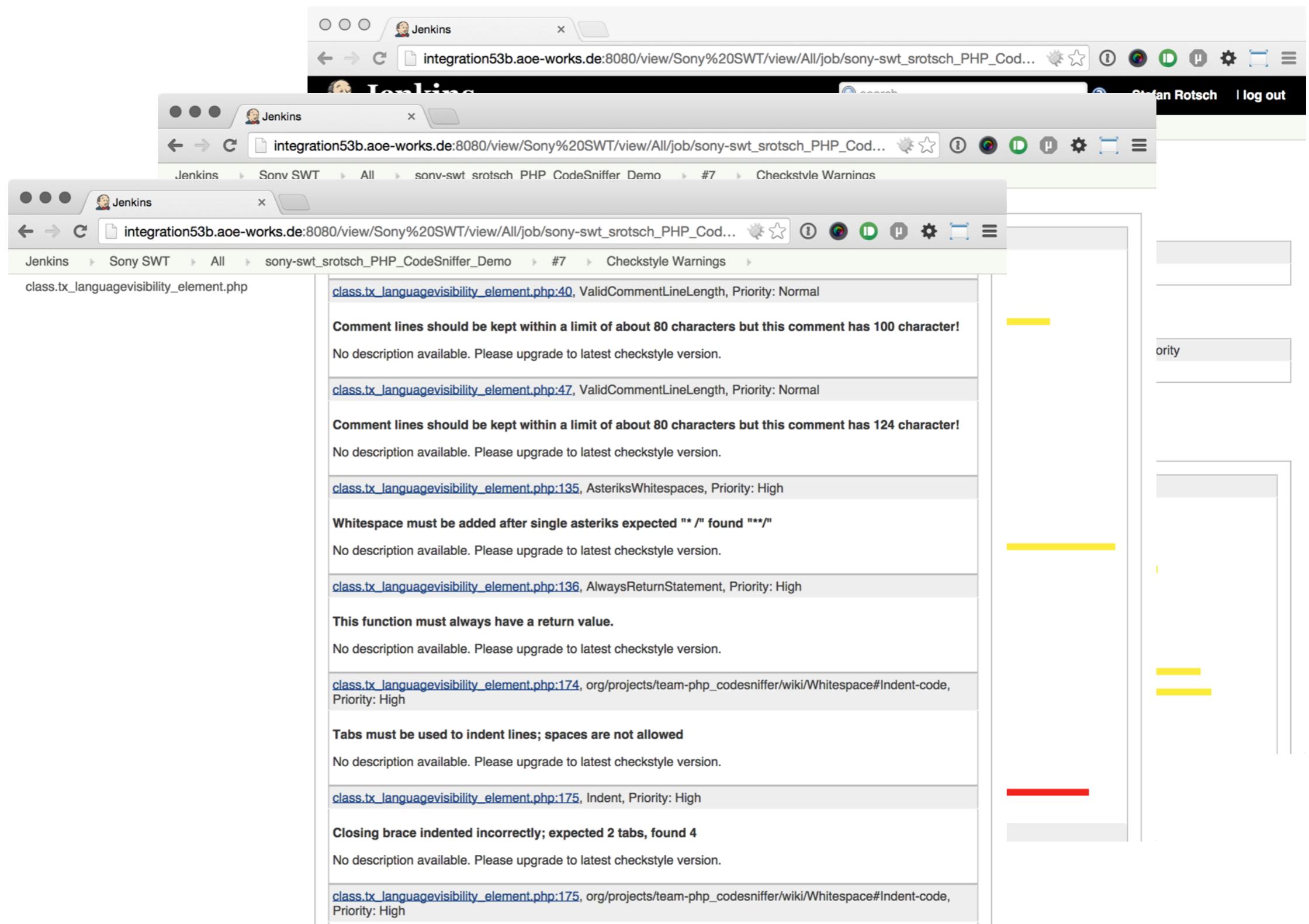
## Details

File	Total	Distribution
<a href="#">class.tx_languagevisibility_InvalidRowException.php</a>	3	
<a href="#">class.tx_languagevisibility_abstractservices.php</a>	4	
<a href="#">class.tx_languagevisibility_beUser.php</a>	1	
<a href="#">class.tx_languagevisibility_beservices.php</a>	38	
<a href="#">class.tx_languagevisibility_cacheManager.php</a>	2	
<a href="#">class.tx_languagevisibility_celement.php</a>	5	
<a href="#">class.tx_languagevisibility_daocommon.php</a>	3	
<a href="#">class.tx_languagevisibility_daocommon_stub.php</a>	1	
<a href="#">class.tx_languagevisibility_element.php</a>	49	
<a href="#">class.tx_languagevisibility_elementFactory.php</a>	52	
<a href="#">class.tx_languagevisibility_fceelement.php</a>	1	
<a href="#">class.tx_languagevisibility_fceoverlayelement.php</a>	1	
<a href="#">class.tx_languagevisibility_fconrices.php</a>	4	

# PHP\_CodeSniffer Results



# PHP\_CodeSniffer Results



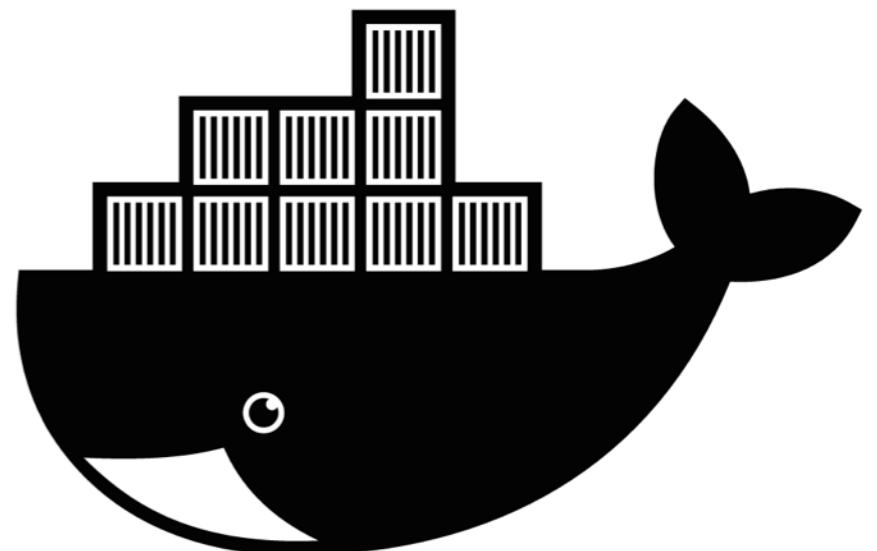
The screenshot shows a Jenkins interface with three windows open, all displaying the same job details: 'sony-swt\_srotsch PHP CodeSniffer Demo' (Job #7). The top window shows the overall job status and configuration. The middle window shows the 'Checkstyle Warnings' report for the file 'class.tx\_languagevisibility\_element.php'. The bottom window shows the 'Checkstyle Warnings' report for the same file. The reports list several code quality issues with their file, line, and column details, priority levels (Normal, High), and descriptions. The Jenkins interface includes a sidebar with various Jenkins-related icons and a user profile for 'Stefan Rotsch'.

File	Line	Priority	Description
class.tx_languagevisibility_element.php	40	Normal	Comment lines should be kept within a limit of about 80 characters but this comment has 100 character!
class.tx_languagevisibility_element.php	47	Normal	Comment lines should be kept within a limit of about 80 characters but this comment has 124 character!
class.tx_languagevisibility_element.php	135	High	Whitespace must be added after single asterisks expected '*' found "*/"
class.tx_languagevisibility_element.php	136	High	This function must always have a return value.
class.tx_languagevisibility_element.php	174	High	Tabs must be used to indent lines; spaces are not allowed
class.tx_languagevisibility_element.php	175	High	Closing brace indented incorrectly; expected 2 tabs, found 4

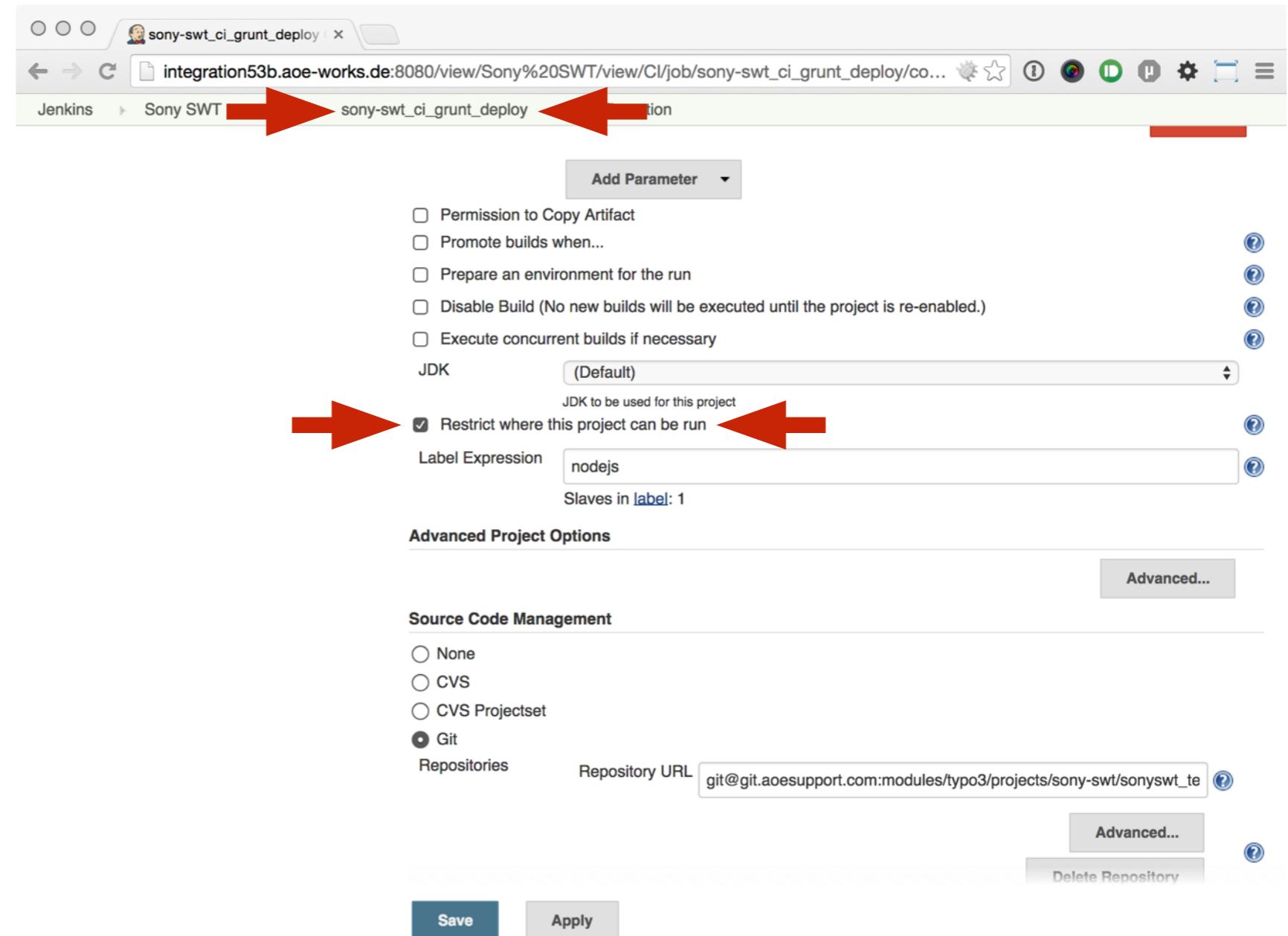
# Master/Slave Environments

# Master/Slave Benefits

- **Limited** build processors
- **Conflicts** between projects
- **Specialized** slave nodes
- **On-demand** nodes



# Master/Slave Setup



The screenshot shows the Jenkins project configuration page for 'sony-swt\_ci\_grunt\_deploy'. The URL in the browser bar is `integration53b.aoe-works.de:8080/view/Sony%20SWT/view/CI/job/sony-swt_ci_grunt_deploy/config`. The page displays various configuration options for the project, including build triggers, build steps, and post-build actions. A red arrow points to the 'Restrict where this project can be run' checkbox, which is checked. Another red arrow points to the 'Label Expression' field, which contains 'nodejs'. The 'Source Code Management' section shows 'Git' selected as the provider, with a repository URL of `git@git.aoesupport.com:modules/typo3/projects/sony-swt/sonyswt_te`. At the bottom, there are 'Save' and 'Apply' buttons.

sony-swt\_ci\_grunt\_deploy

Jenkins > Sony SWT > sony-swt\_ci\_grunt\_deploy

Add Parameter

Permission to Copy Artifact

Promote builds when...

Prepare an environment for the run

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

Execute concurrent builds if necessary

JDK (Default)

Restrict where this project can be run

Label Expression nodejs

Slaves in [label](#): 1

Advanced Project Options

Advanced...

Source Code Management

None

CVS

CVS Projectset

Git

Repositories

Repository URL `git@git.aoesupport.com:modules/typo3/projects/sony-swt/sonyswt_te`

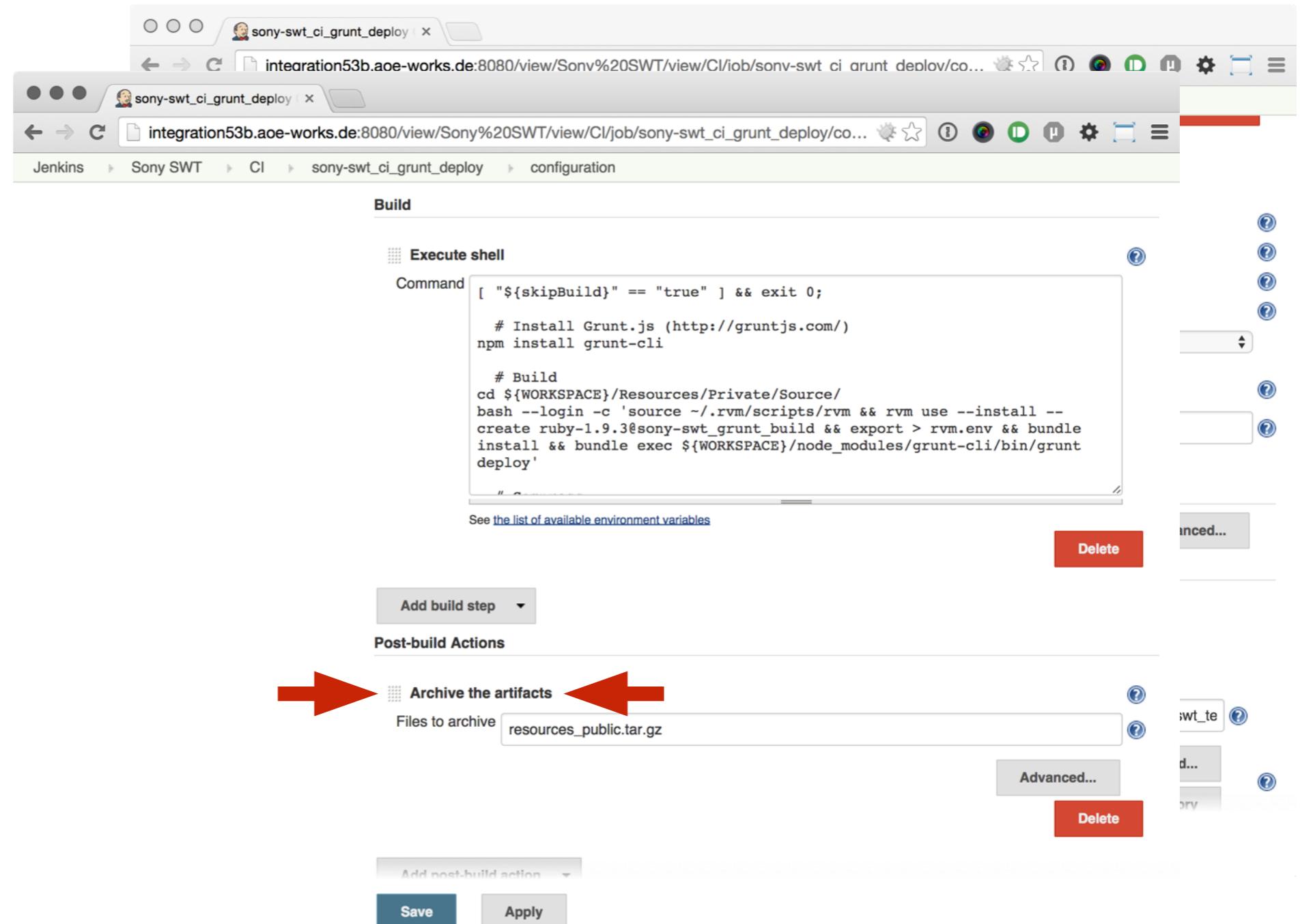
Advanced...

Delete Repository

Save

Apply

# Master/Slave Setup



The screenshot shows the Jenkins configuration page for a job named "sony-swt\_ci\_grunt\_deploy". The page is divided into two main sections: "Build" and "Post-build Actions".

**Build Section:**

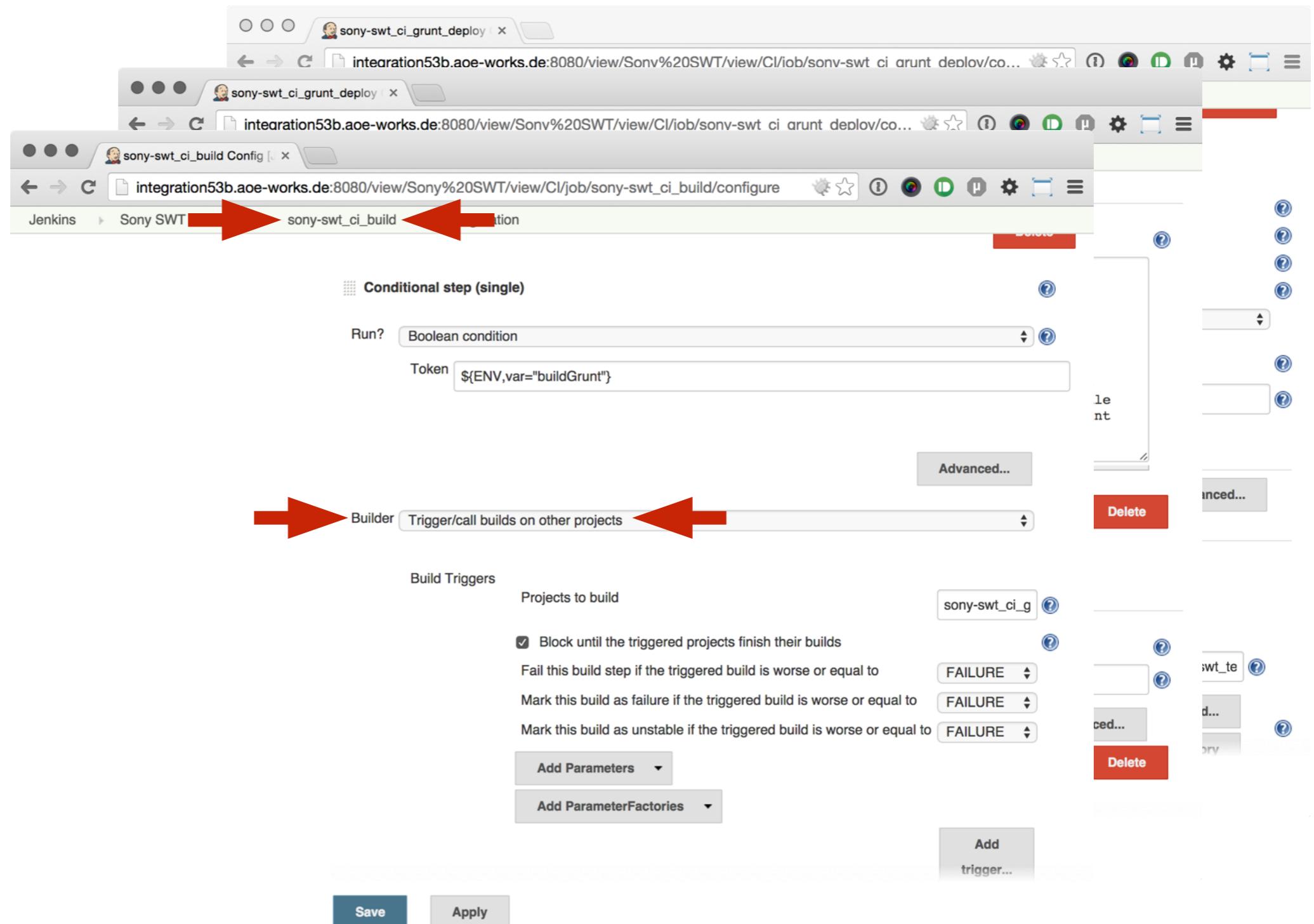
- Execute shell:**  
Command:  
[ "\${skipBuild}" == "true" ] && exit 0;  
  
# Install Grunt.js (<http://gruntjs.com/>)  
npm install grunt-cli  
  
# Build  
cd \${WORKSPACE}/Resources/Private/Source/  
bash --login -c 'source ~/.rvm/scripts/rvm && rvm use --install --create ruby-1.9.3@sony-swt\_grunt\_build && export > rvm.env && bundle install && bundle exec \${WORKSPACE}/node\_modules/grunt-cli/bin/grunt deploy'

**Post-build Actions Section:**

- Archive the artifacts:**  
Files to archive: resources\_public.tar.gz

Two red arrows point to the "Archive the artifacts" section, highlighting the configuration for artifact archiving.

# Master/Slave Setup



sony-swt\_ci\_build

Conditional step (single)

Run? Boolean condition

Token \${ENV,var="buildGrunt"}

Builder Trigger/call builds on other projects

Build Triggers

Projects to build

sony-swt\_ci\_g

Block until the triggered projects finish their builds

Fail this build step if the triggered build is worse or equal to FAILURE

Mark this build as failure if the triggered build is worse or equal to FAILURE

Mark this build as unstable if the triggered build is worse or equal to FAILURE

Add Parameters

Add ParameterFactories

Add trigger...

Save

Apply

# Master/Slave Setup

Copy artifacts from another project

Project name: sony-swt\_ci\_grunt\_deploy

Which build: Latest successful build

Stable build only

Artifacts to copy: resources\_public.tar.gz

Target directory:

Parameter filters:

Flatten directories  Optional  Fingerprint Artifacts

Invoke Ant

Ant Version: Default

Targets: build

Save Apply

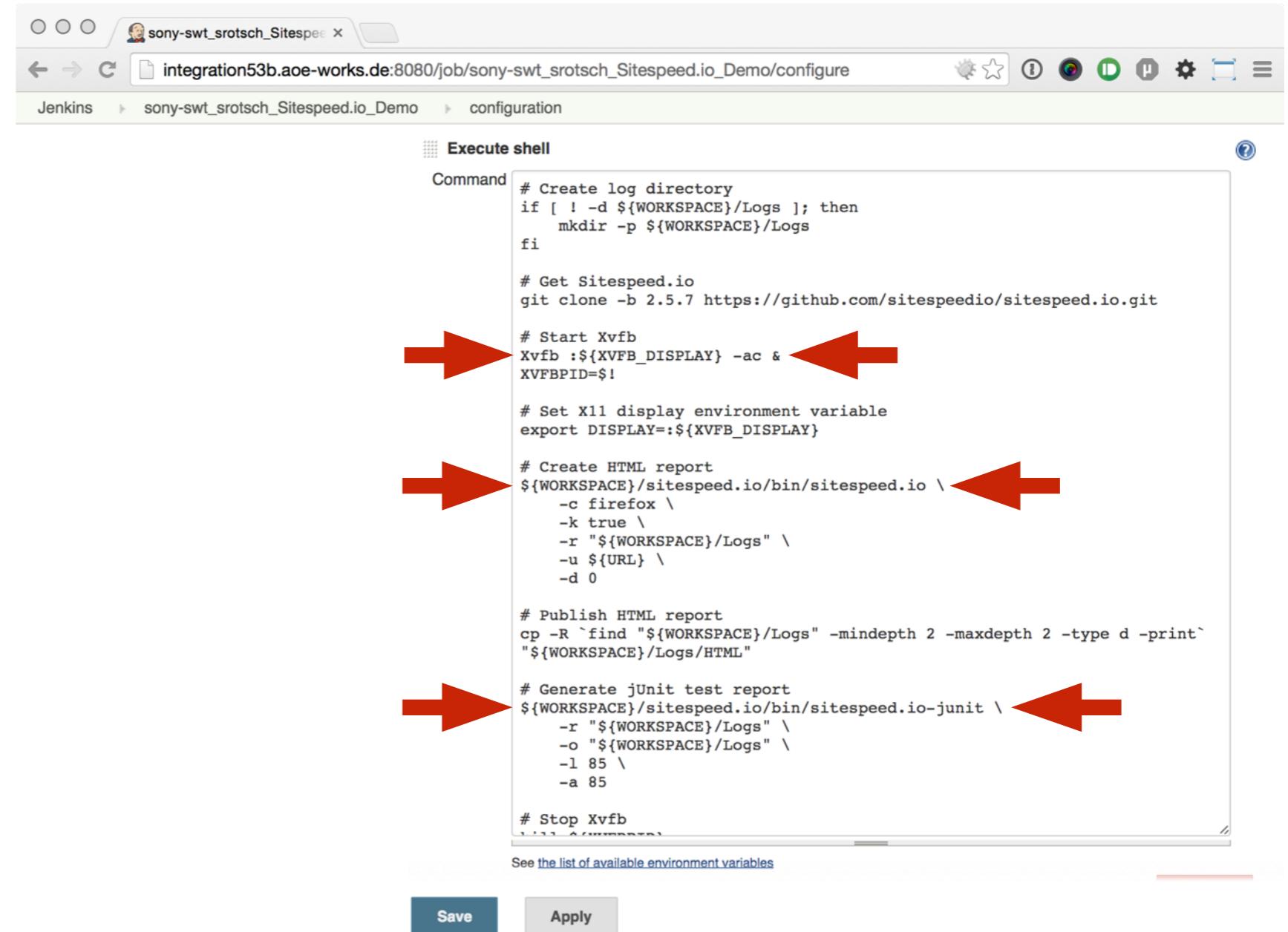
# Headless Sitespeed.io

# Sitespeed.io

- Website **speed** and **performance** analyzer
- Advanced metrics require "real" **browser**
- **Xvfb** as a lightweight alternative to Selenium
- **Firefox** only



# Sitespeed.io Setup



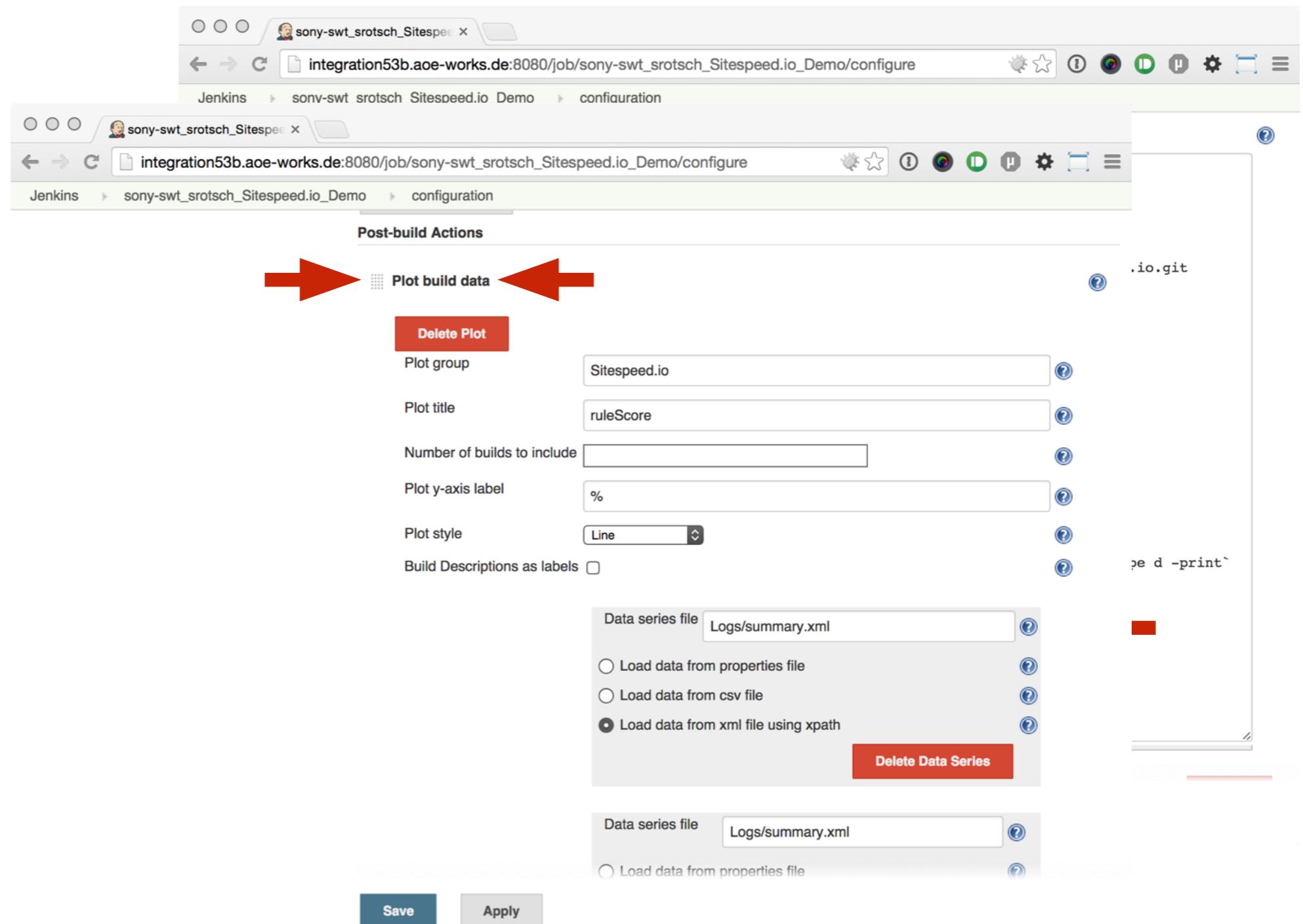
The screenshot shows a Jenkins configuration page for a job named 'sony-swt\_srotsch\_Sitespeed.io\_Demo'. The 'configuration' section is open, and the 'Execute shell' tab is selected. The command field contains a shell script for setting up and running Sitespeed.io. Several red arrows point to specific lines in the command, highlighting the following steps:

- A red arrow points to the line: `Xvfb :${XVFB_DISPLAY} -ac & XVFBPID=$!`. This line starts the Xvfb display.
- A red arrow points to the line: `#{WORKSPACE}/sitespeed.io/bin/sitespeed.io \`. This line starts the Sitespeed.io command.
- A red arrow points to the line: `-c firefox \`. This line specifies the browser to use.
- A red arrow points to the line: `-r "${WORKSPACE}/Logs" \`. This line specifies the log directory.
- A red arrow points to the line: `-u ${URL} \`. This line specifies the URL to test.
- A red arrow points to the line: `-d 0`. This line specifies the depth of the report.
- A red arrow points to the line: `cp -R `find "${WORKSPACE}/Logs" -mindepth 2 -maxdepth 2 -type d -print` "${WORKSPACE}/Logs/HTML"`. This line publishes the HTML report.
- A red arrow points to the line: `#{WORKSPACE}/sitespeed.io/bin/sitespeed.io-junit \`. This line generates the jUnit test report.
- A red arrow points to the line: `-r "${WORKSPACE}/Logs" \`. This line specifies the log directory for the jUnit report.
- A red arrow points to the line: `-o "${WORKSPACE}/Logs" \`. This line specifies the output directory for the jUnit report.
- A red arrow points to the line: `-l 85 \`. This line specifies the low threshold for the jUnit report.
- A red arrow points to the line: `-a 85`. This line specifies the average threshold for the jUnit report.

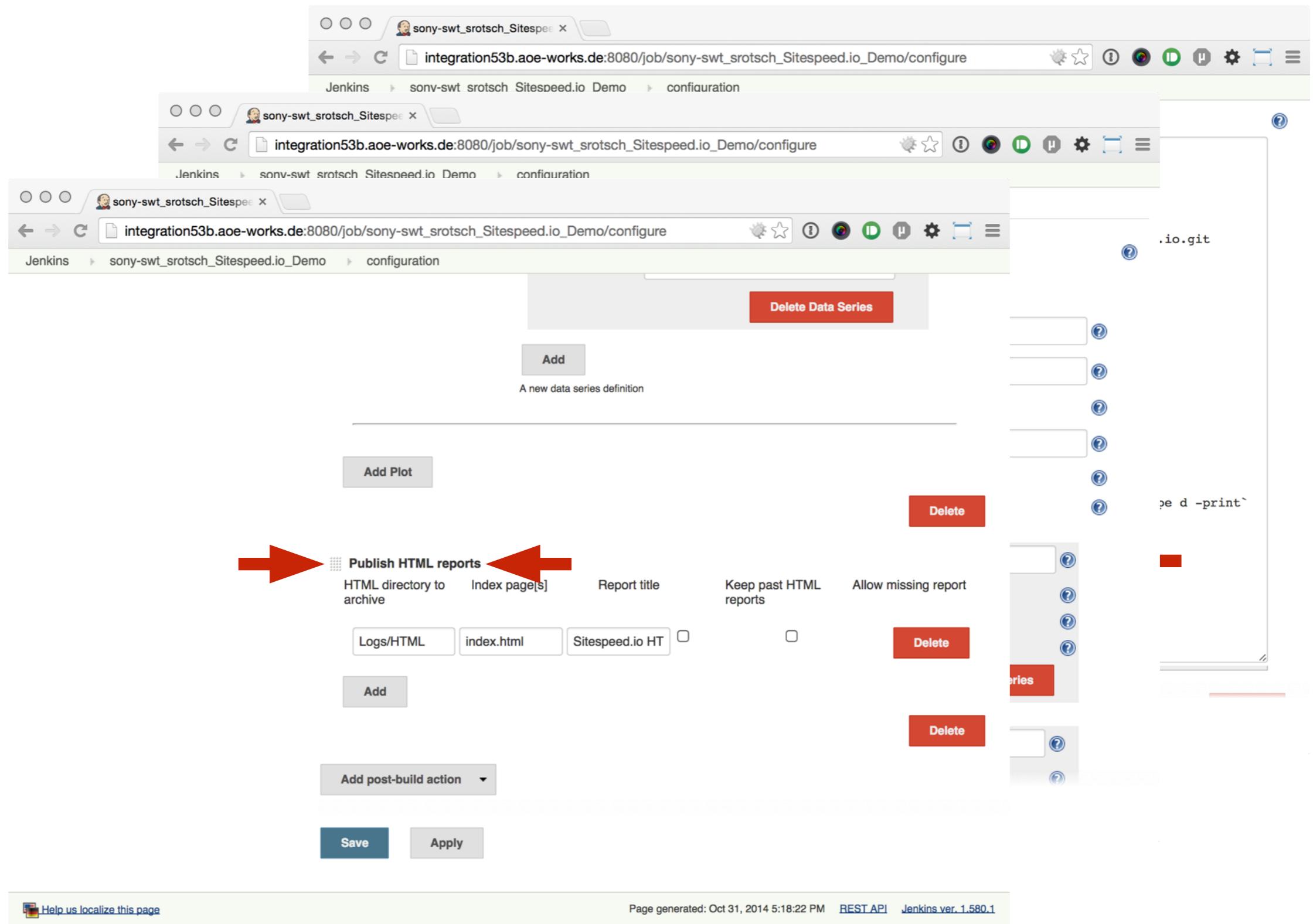
The command concludes with a line: `# Stop Xvfb`.

At the bottom of the configuration page, there are 'Save' and 'Apply' buttons.

# Sitespeed.io Setup



# Sitespeed.io Setup



The screenshot shows the Jenkins configuration page for a job named 'sony-swt\_srotsch\_Sitespeed.io\_Demo'. The 'Publish HTML reports' section is highlighted with a red arrow pointing to the 'Index page[s]' input field. The 'Logs/HTML' radio button is selected. The 'Save' and 'Apply' buttons are at the bottom.

**Publish HTML reports**

HTML directory to archive: Logs/HTML

Index page[s]: index.html

Report title: Sitespeed.io HT

Keep past HTML reports:

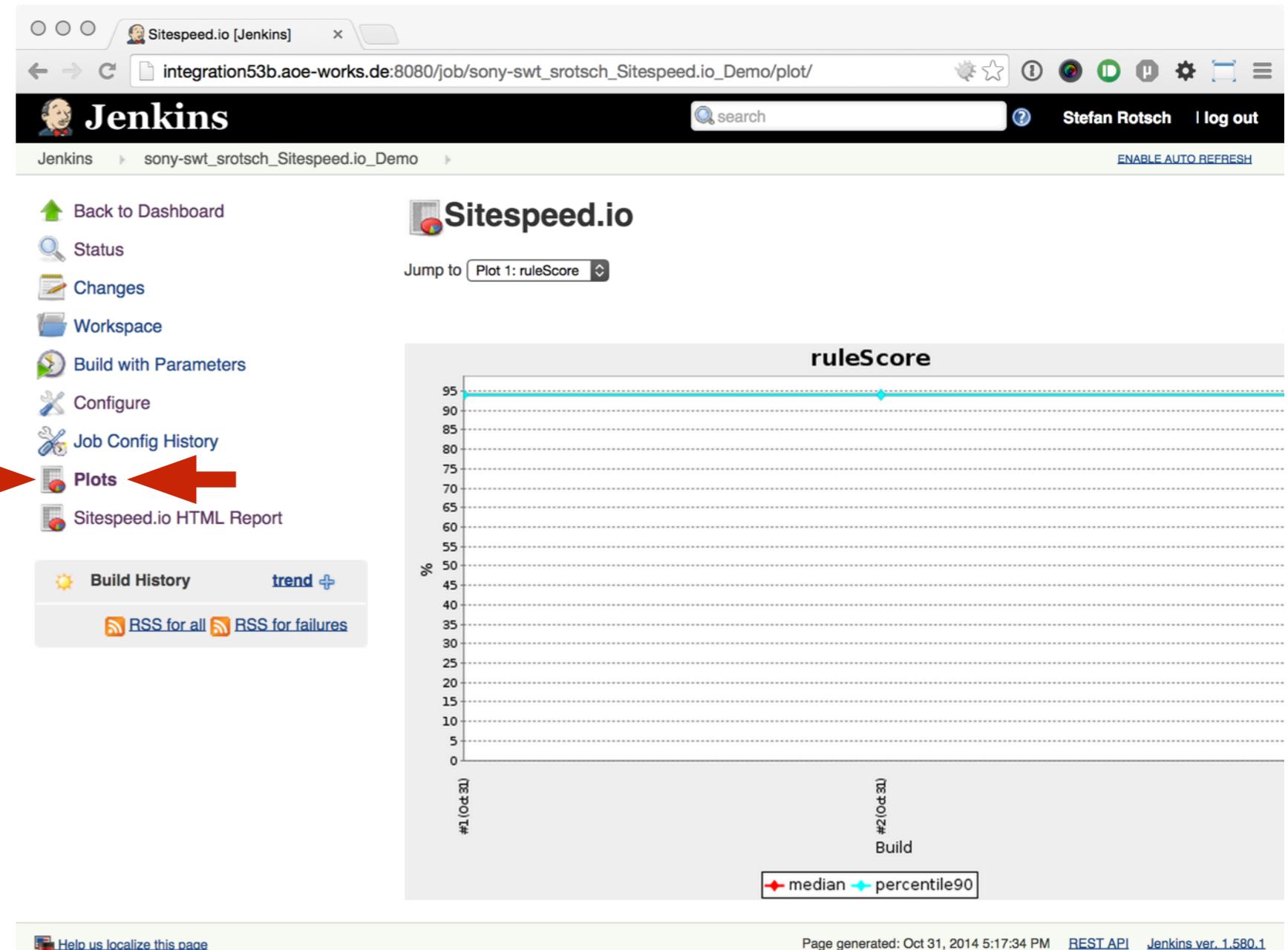
Allow missing report:

**Add post-build action**

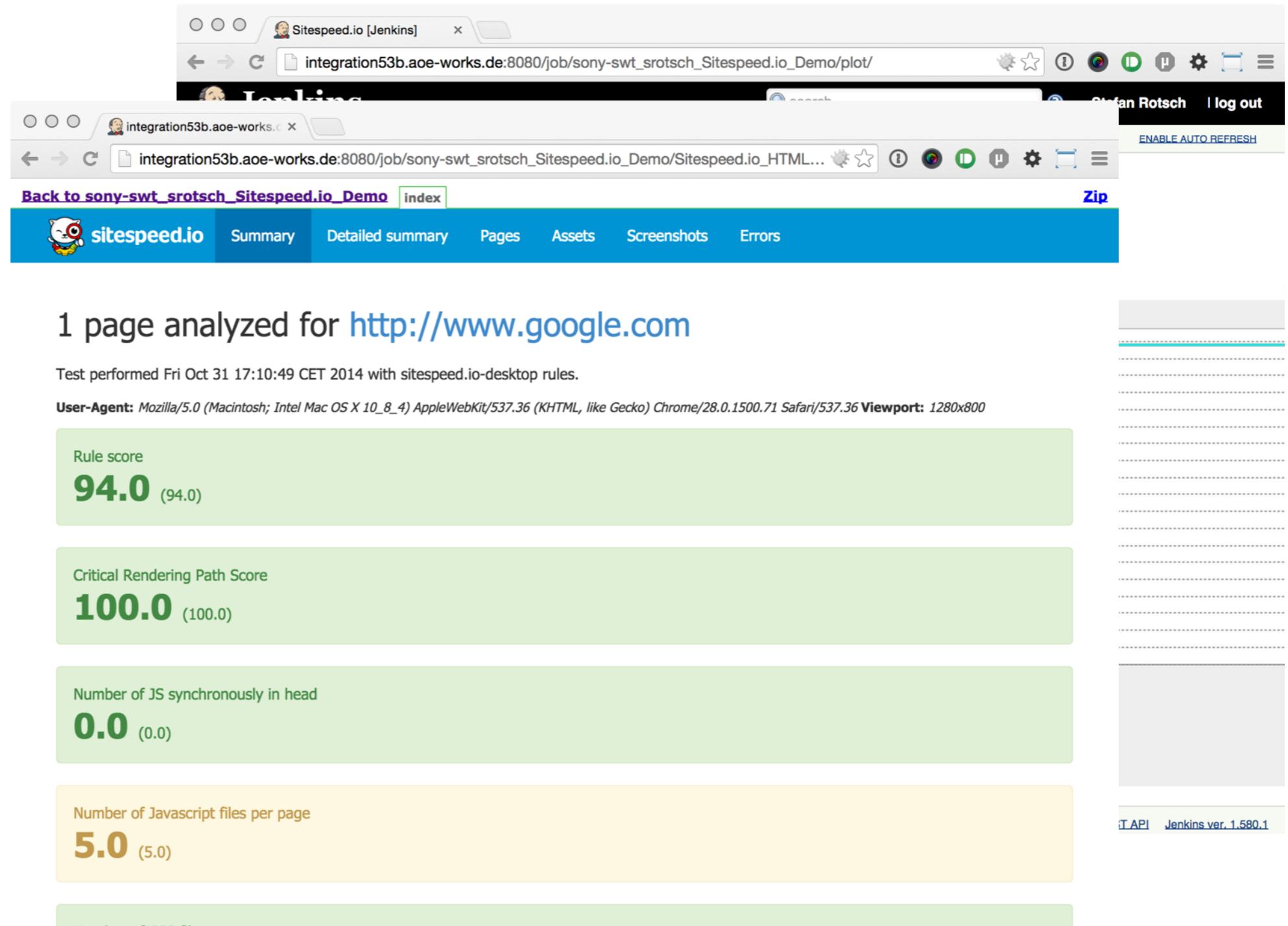
**Save** **Apply**

Page generated: Oct 31, 2014 5:18:22 PM [REST API](#) [Jenkins ver. 1.580.1](#)

# Sitespeed.io Results



# Sitespeed.io Results



1 page analyzed for <http://www.google.com>

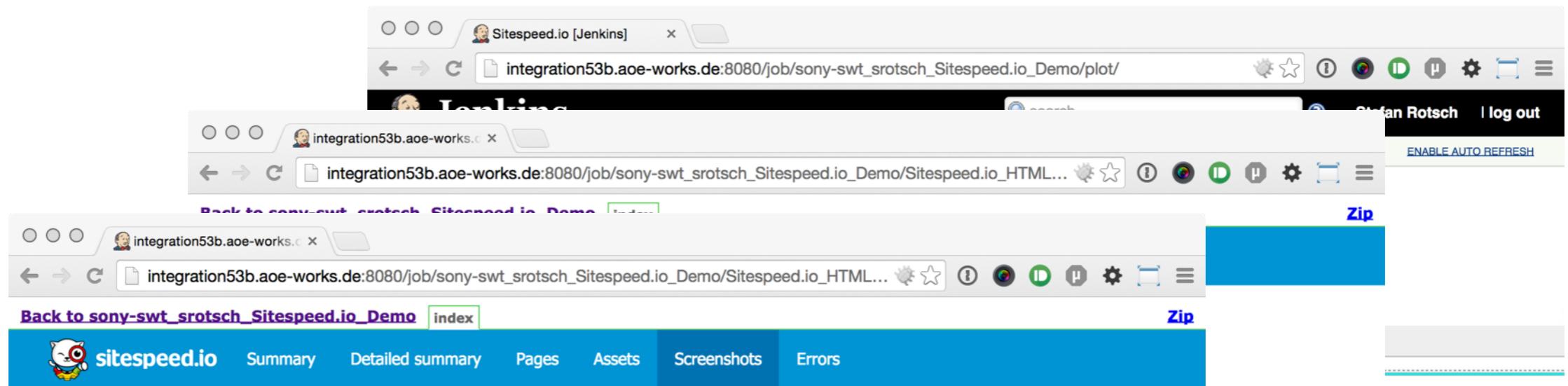
Test performed Fri Oct 31 17:10:49 CET 2014 with sitespeed.io-desktop rules.

**User-Agent:** Mozilla/5.0 (Macintosh; Intel Mac OS X 10\_8\_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/28.0.1500.71 Safari/537.36 **Viewport:** 1280x800

Metric	Score	Details
Rule score	<b>94.0</b> (94.0)	
Critical Rendering Path Score	<b>100.0</b> (100.0)	
Number of JS synchronously in head	<b>0.0</b> (0.0)	
Number of Javascript files per page	<b>5.0</b> (5.0)	

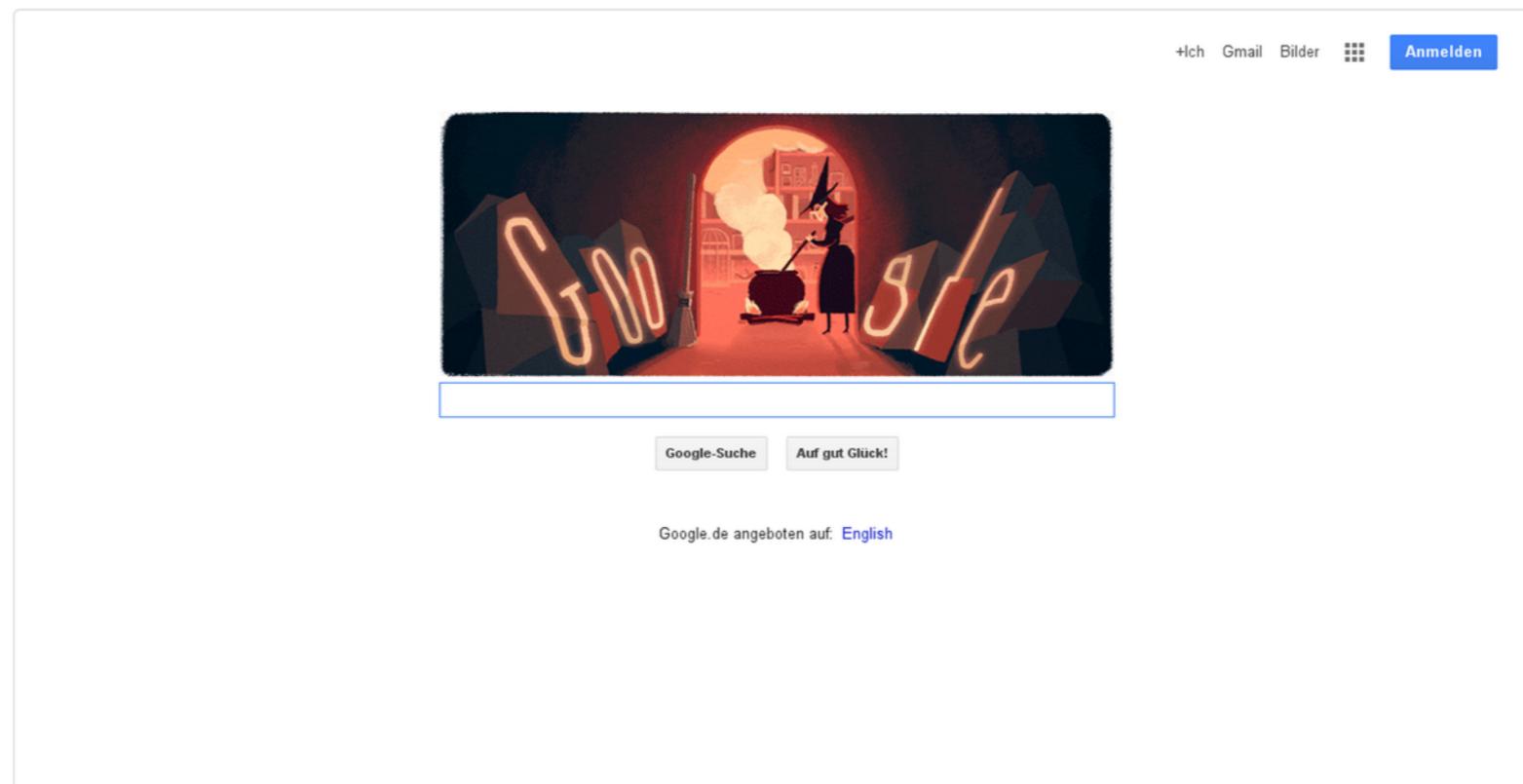
Jenkins ver. 1.580.1

# Sitespeed.io Results



The screenshots for viewport 1280x800

<http://www.google.com>

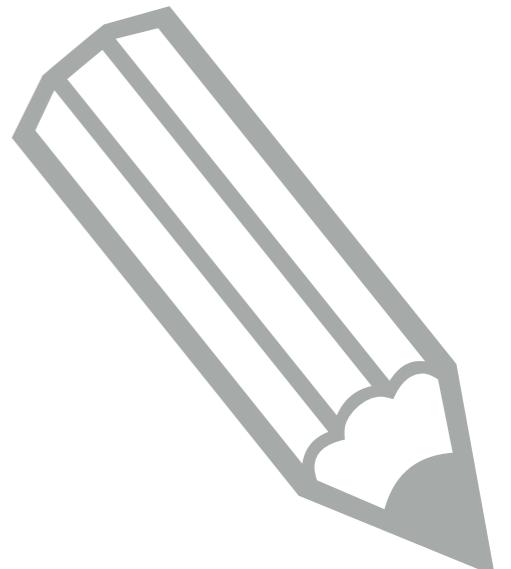


Jenkins ver. 1.580.1

# Resources

# Resources

- <http://jenkins-ci.org/>
- <http://www.eclipse.org/hudson/the-hudson-book/book-hudson.pdf>
- <http://www.sitespeed.io/>
- <http://yslow.org/>



# Questions



# Contact

stefan.rotsch@aoe.com  
@ritschie

