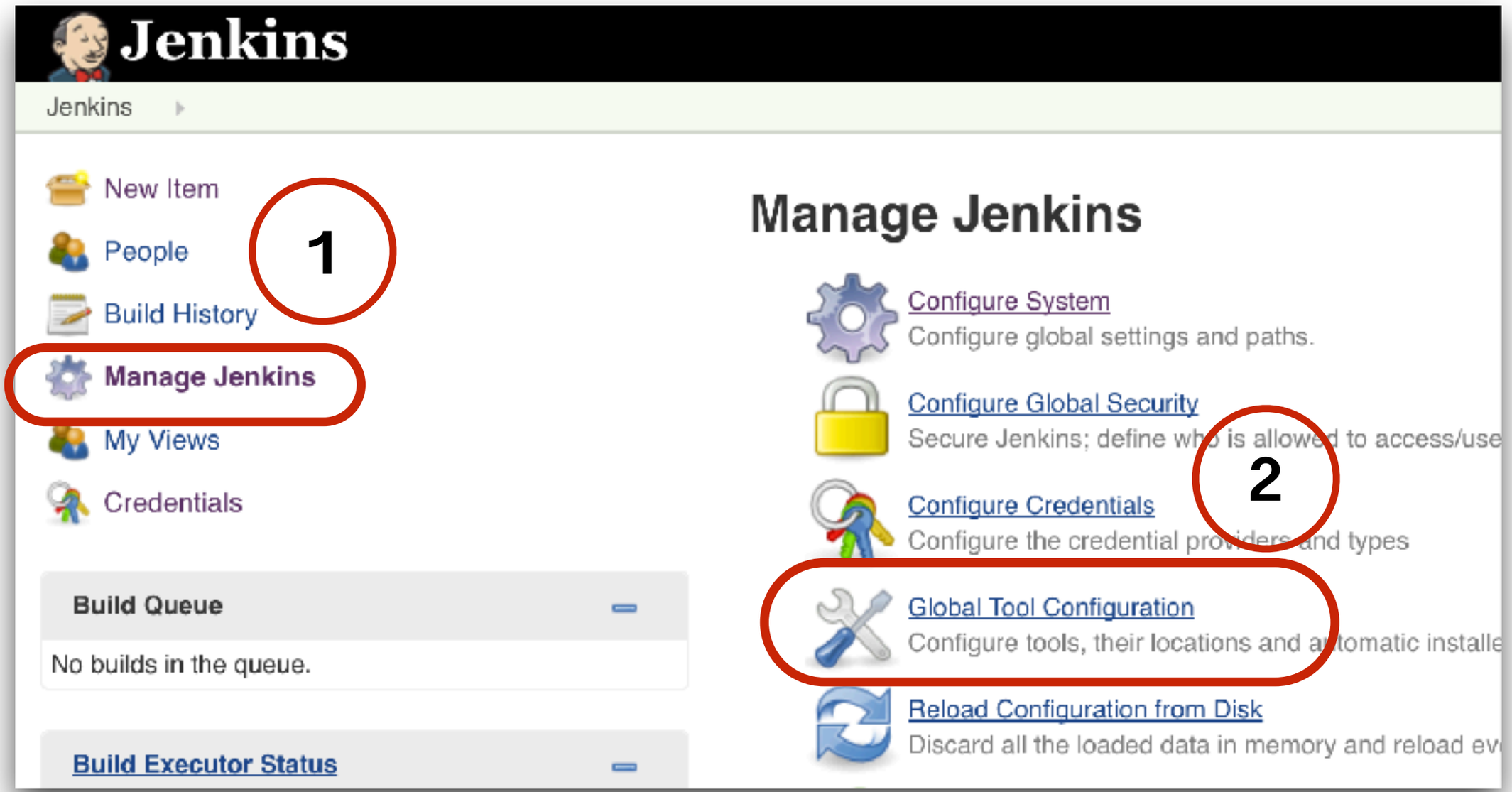
CI/CD Workshop

Jenkins

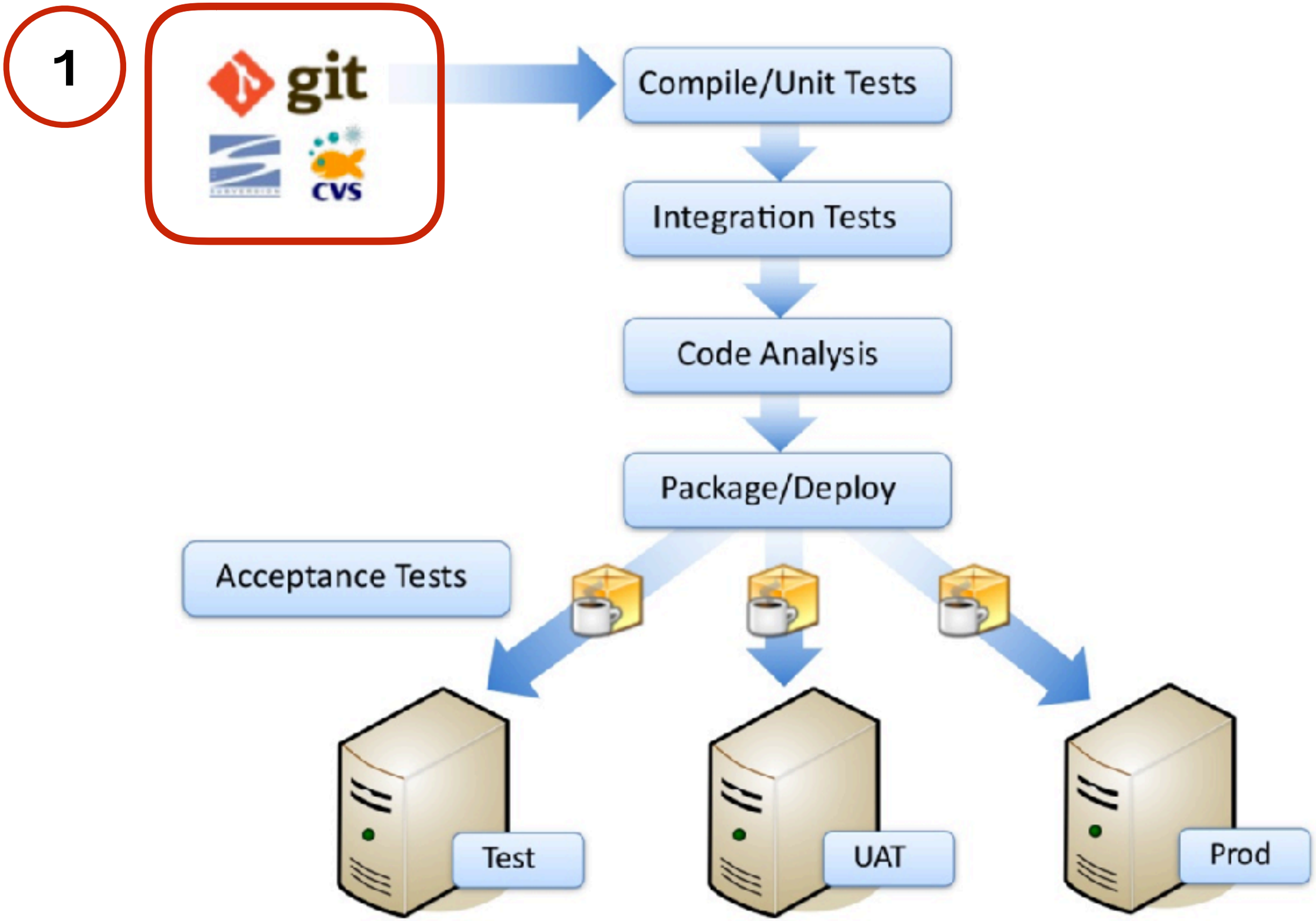
Configuration Jenkins

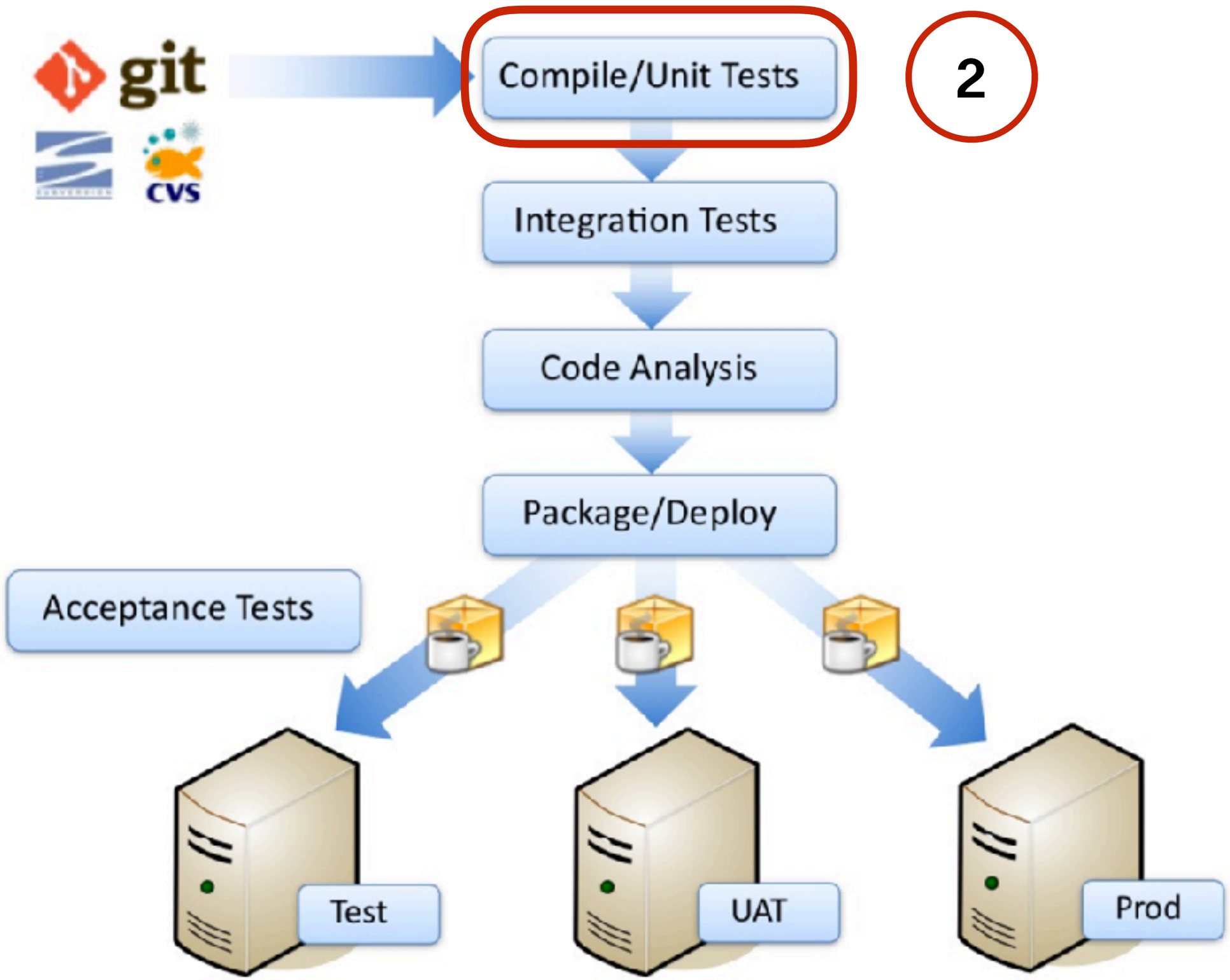
Configuration !!

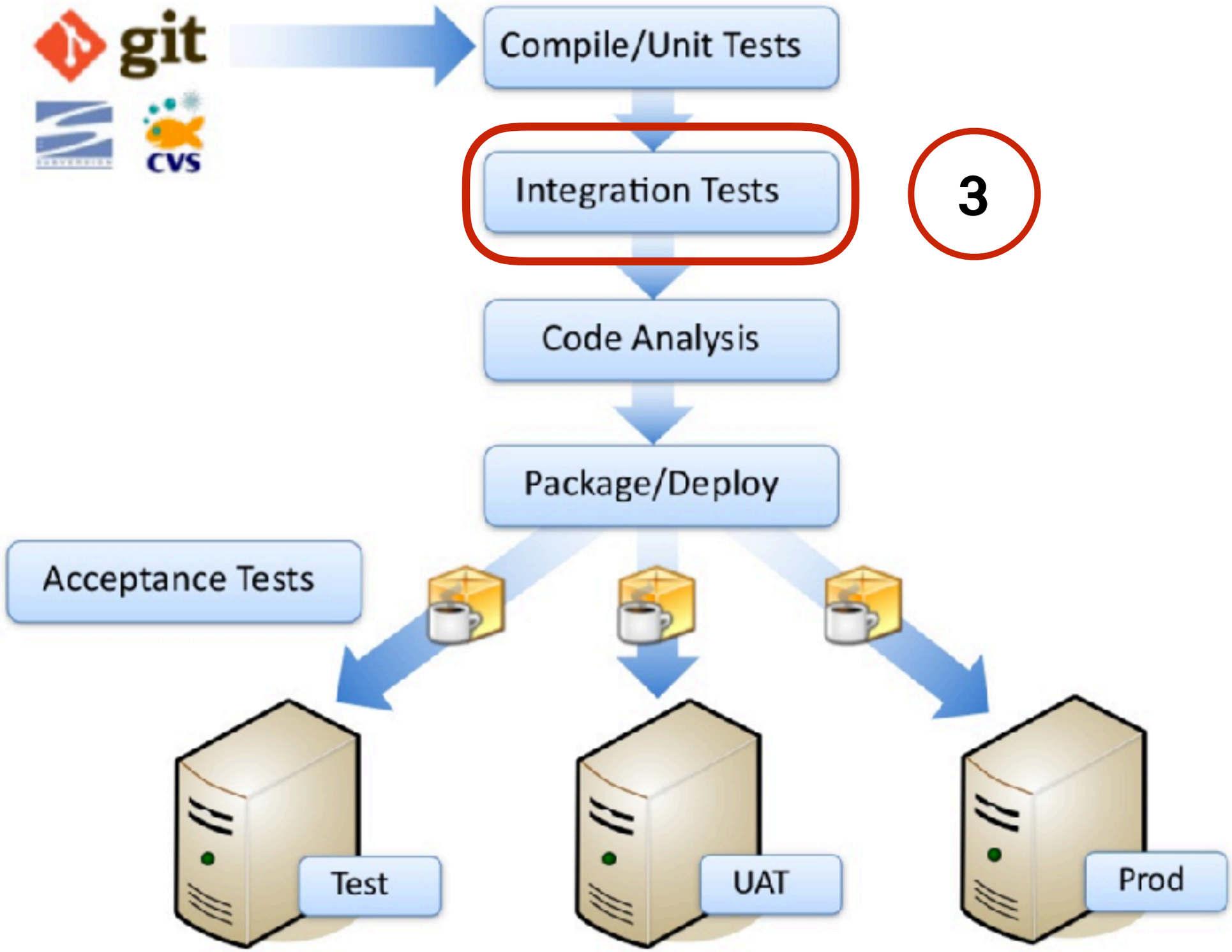
Manage Jenkins -> Global Tool Configuration

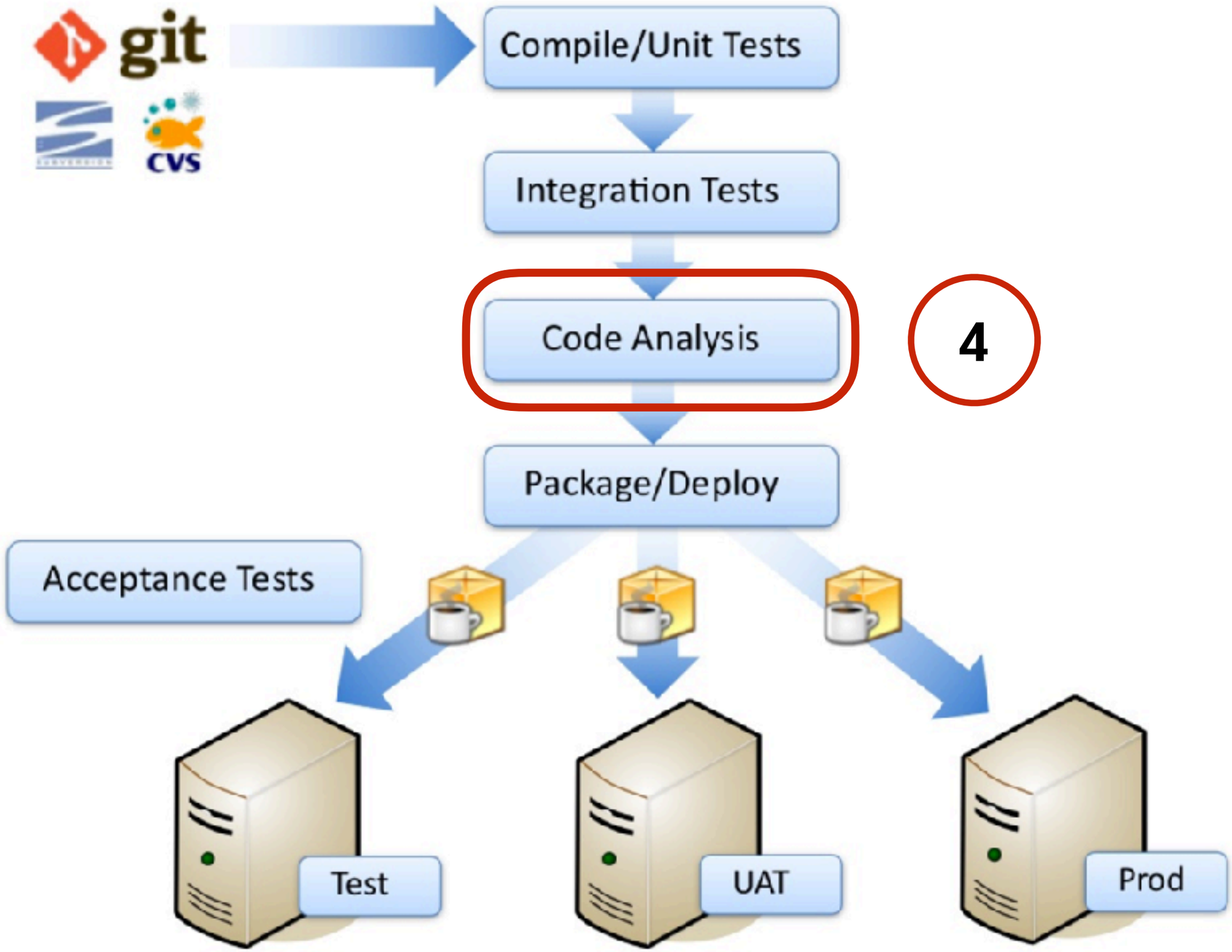


Let’ start with Jenkins


1
git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
Acceptance Tests
Test
UAT
Prod


git
CVS
Compile/Unit Tests
2
Integration Tests
Code Analysis
Package/Deploy
Acceptance Tests
Test
UAT
Prod

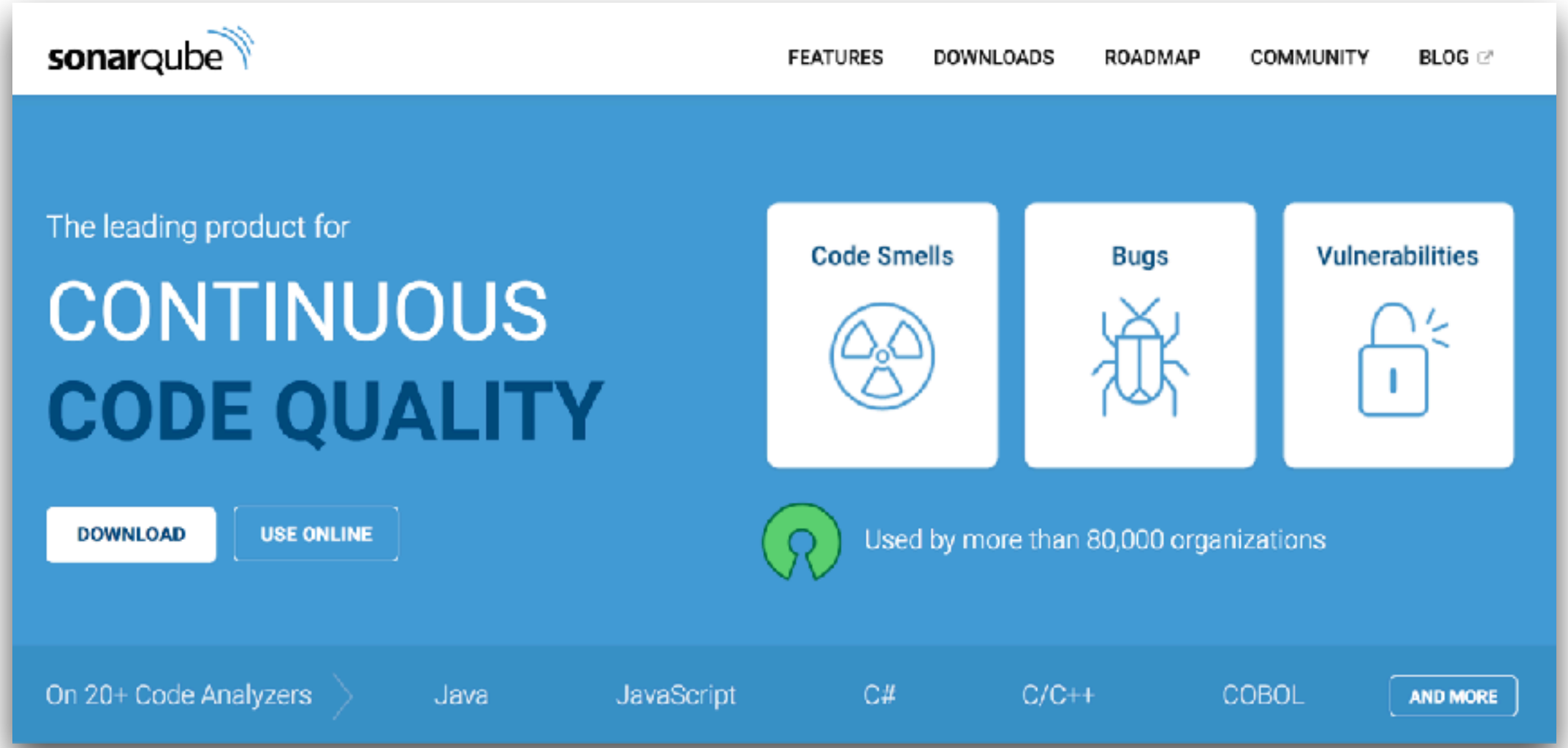

git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
3
Acceptance Tests
Test
UAT
Prod


git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
4
Acceptance Tests
Test
UAT
Prod

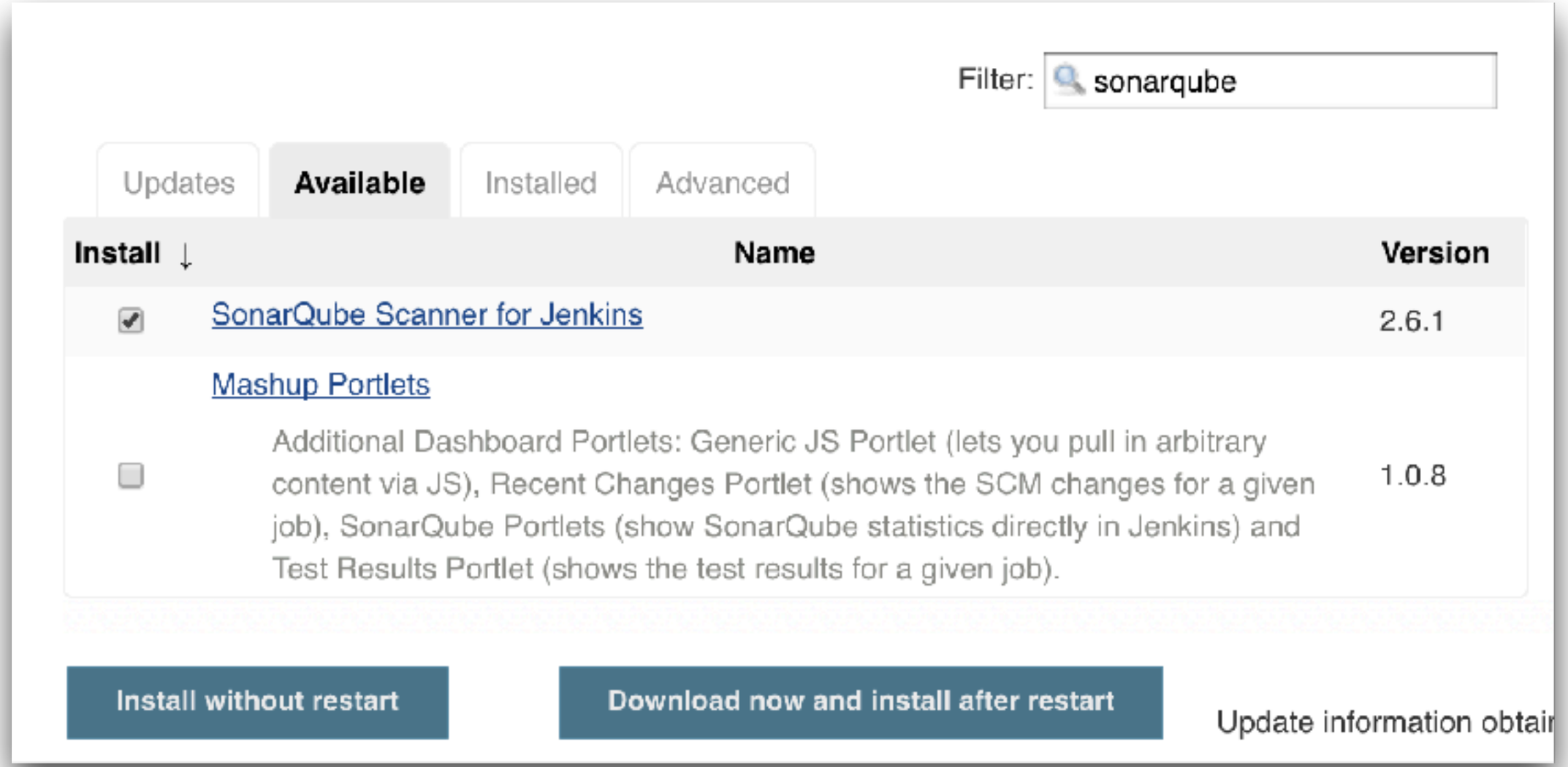
Code analysis with SonarQube

Setup SonarQube

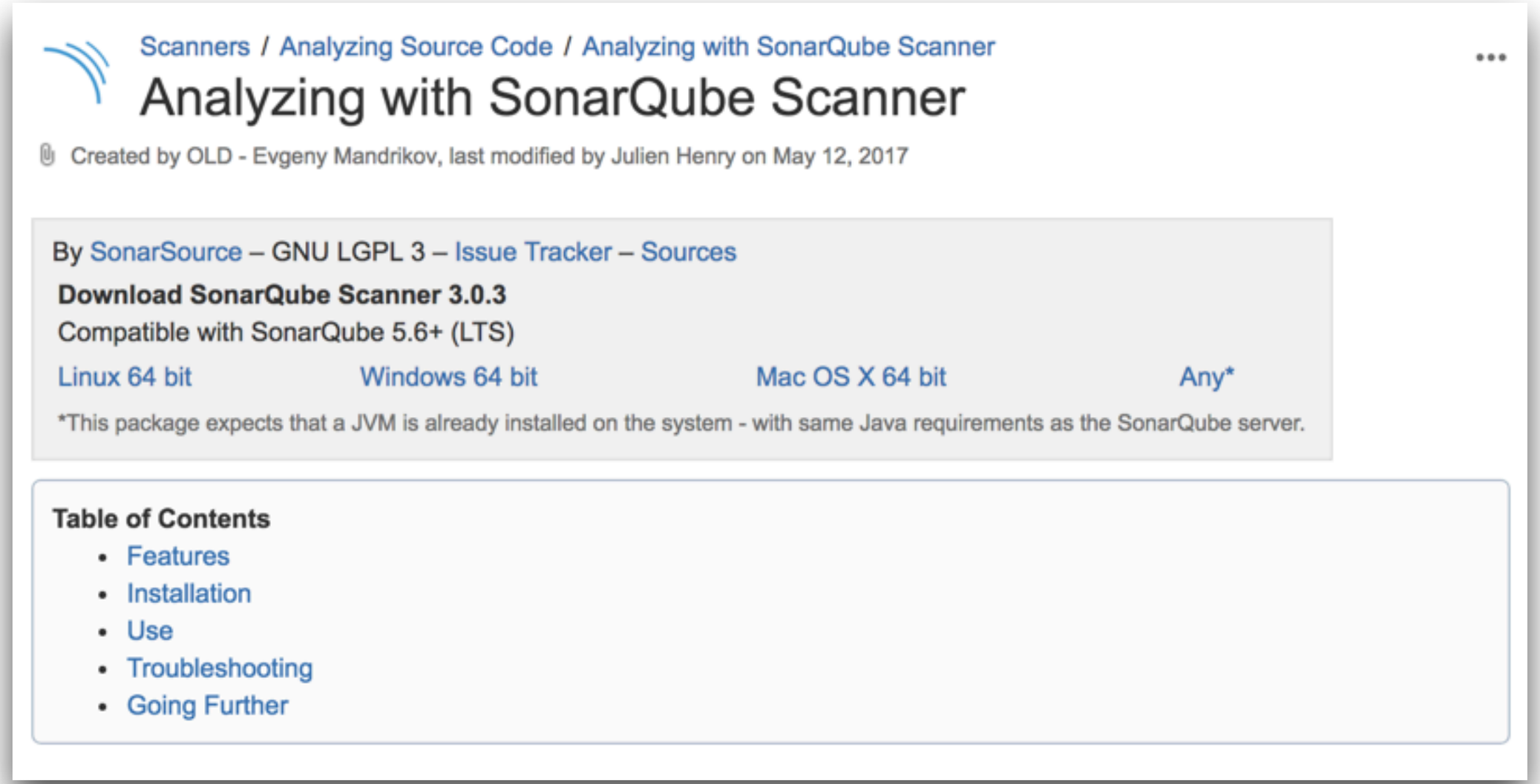
Download from sonarqube.org



Install SonarQube Scanner plugin



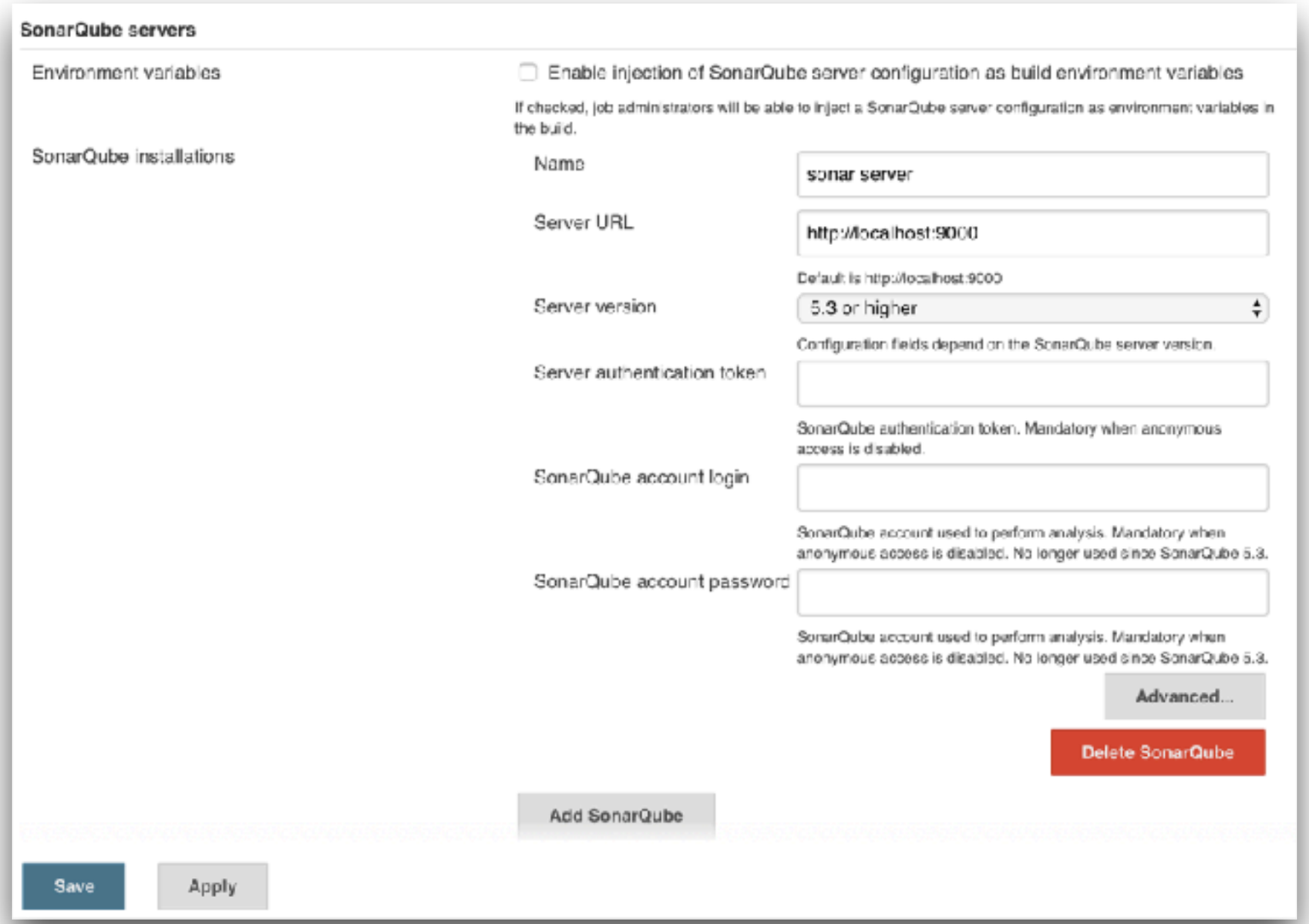
Download SonarQube Scanner



https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner

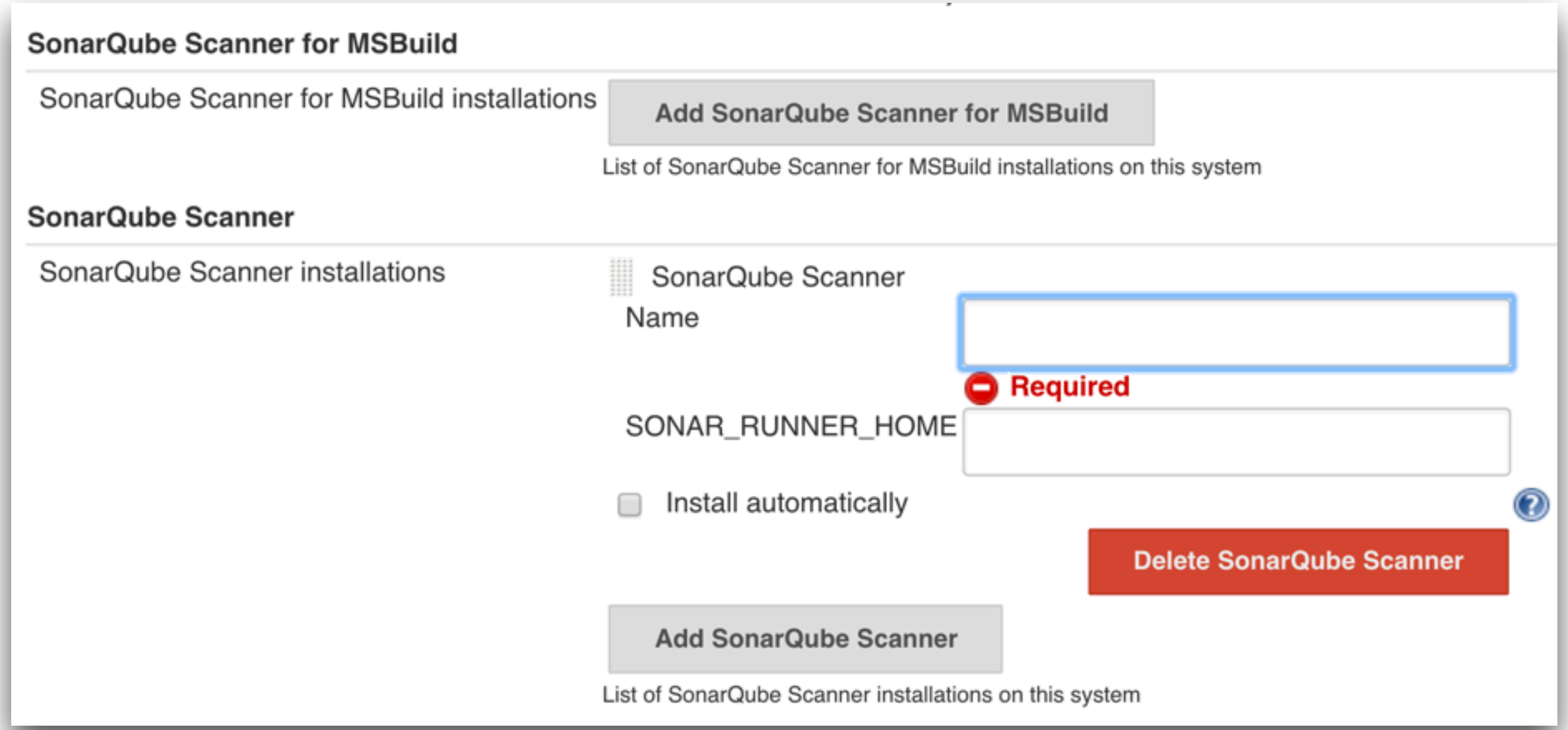
Config SonarQube Server

Manage Jenkins -> Configure System



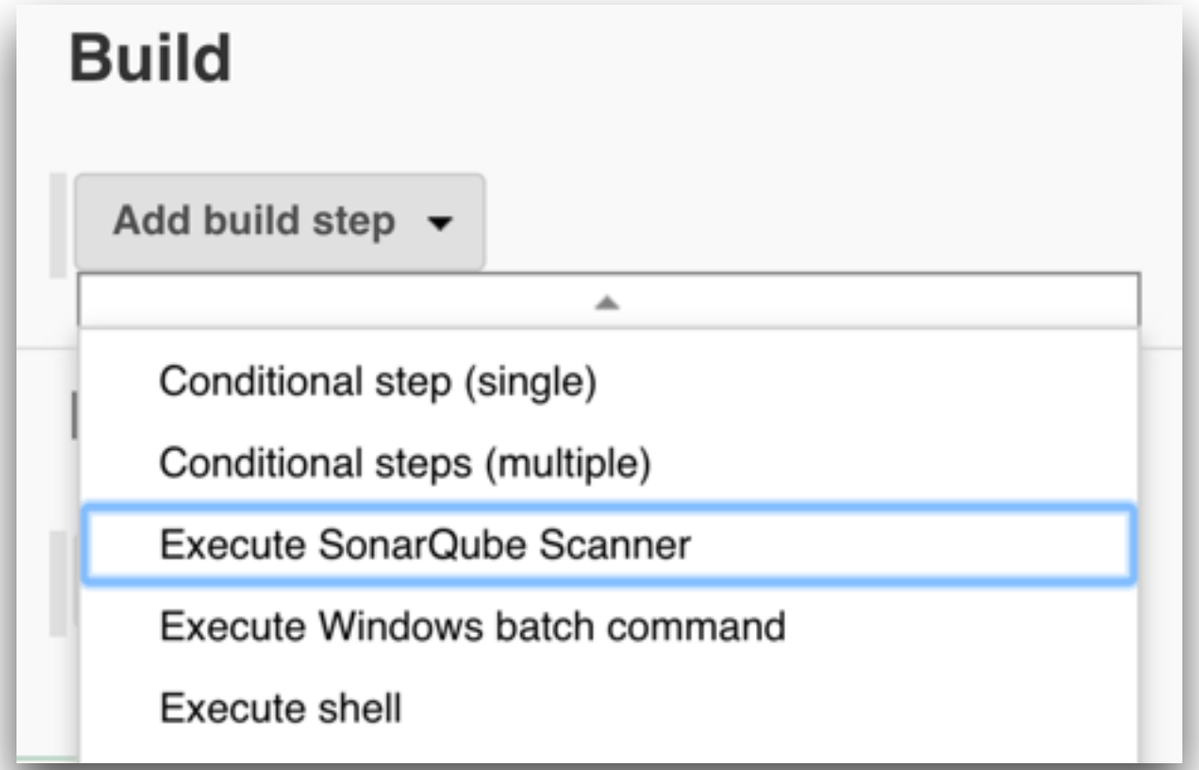
Config SonarQube Scanner

Manage Jenkins -> Global Tool Configuration



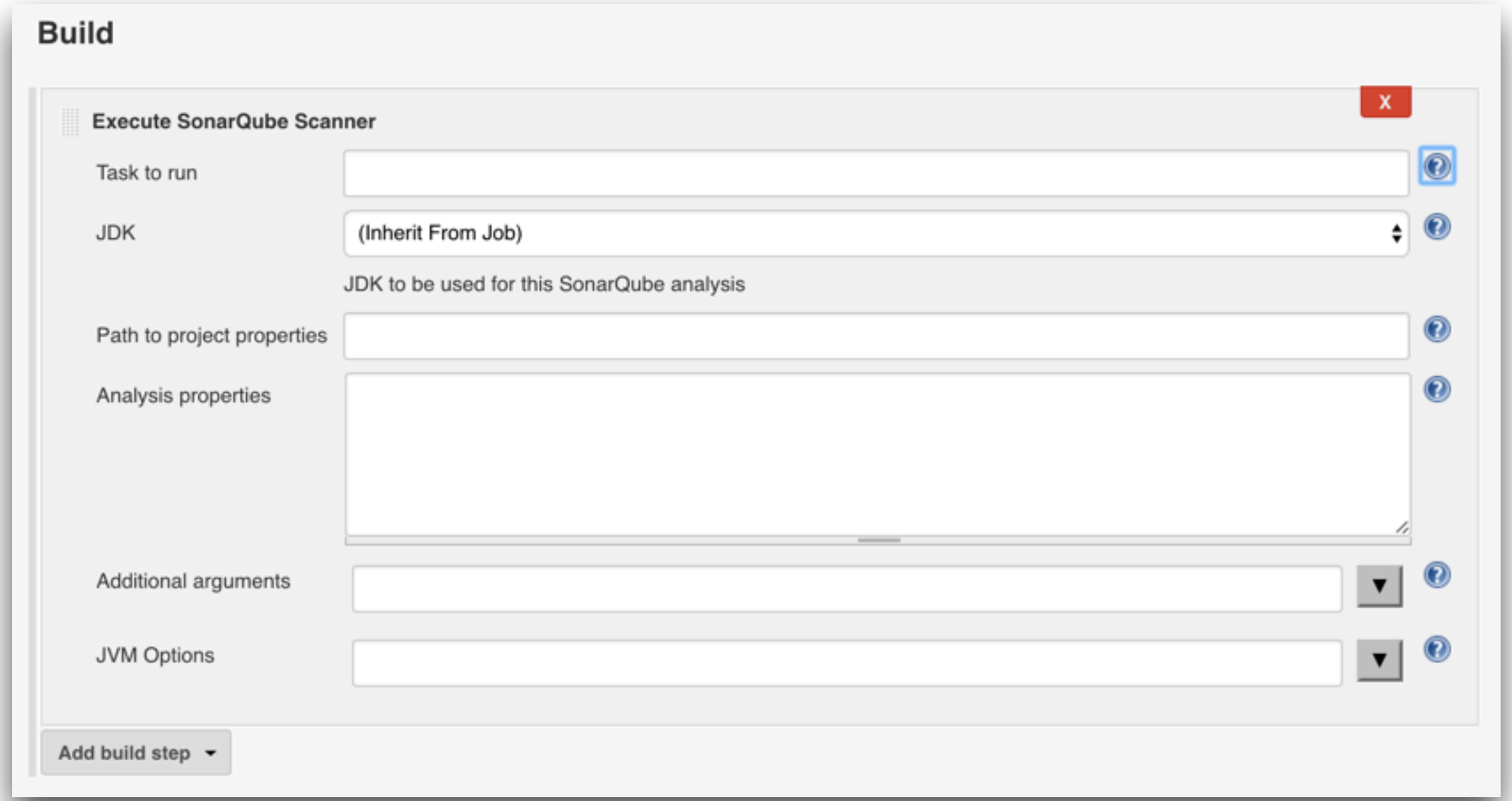
Using SonarQube Scanner

Choose execute SonarQube Scanner



Using SonarQube Scanner

Config in your job



Add sonar-project.properties

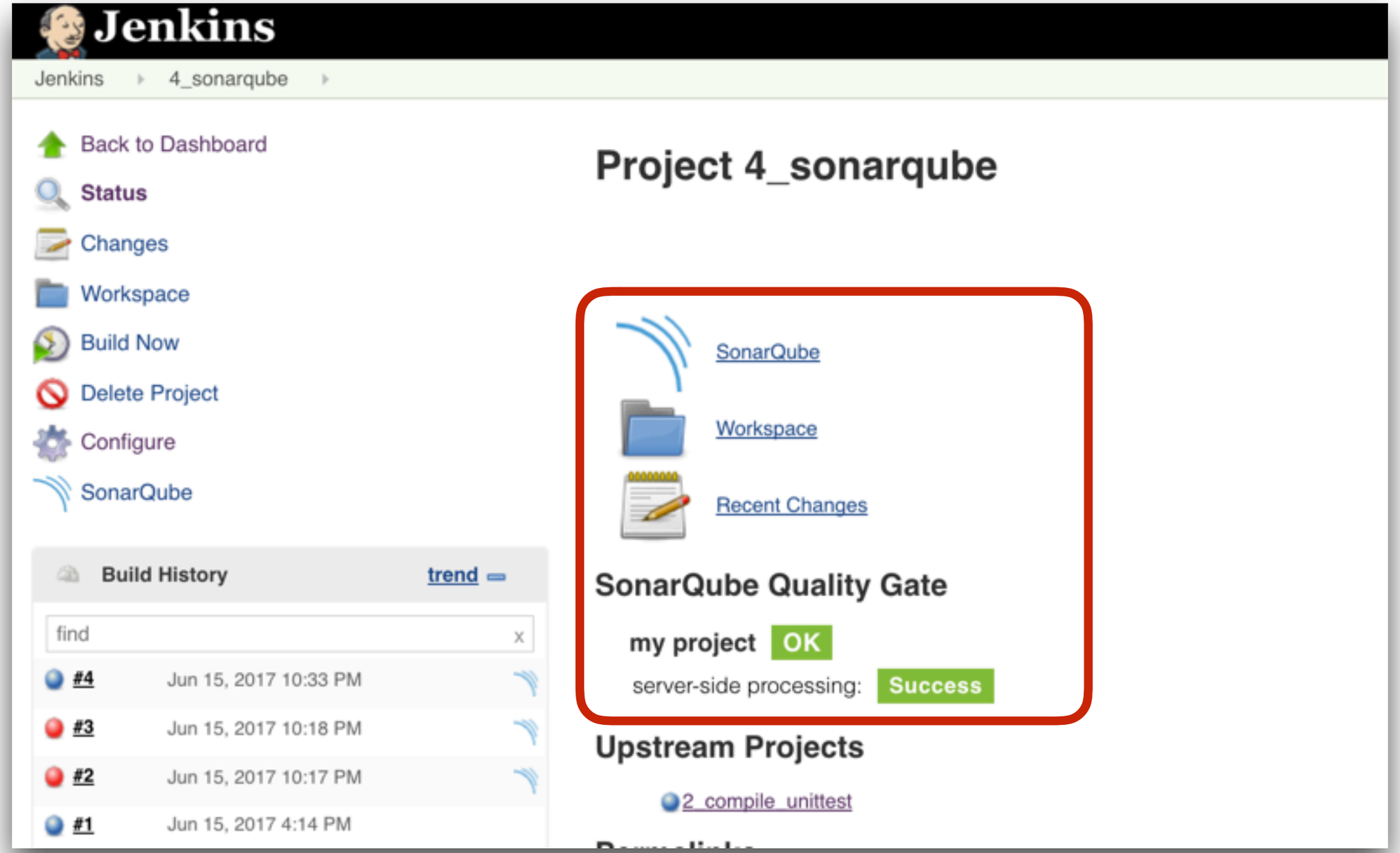
configuration file of project

sonar-project.properties

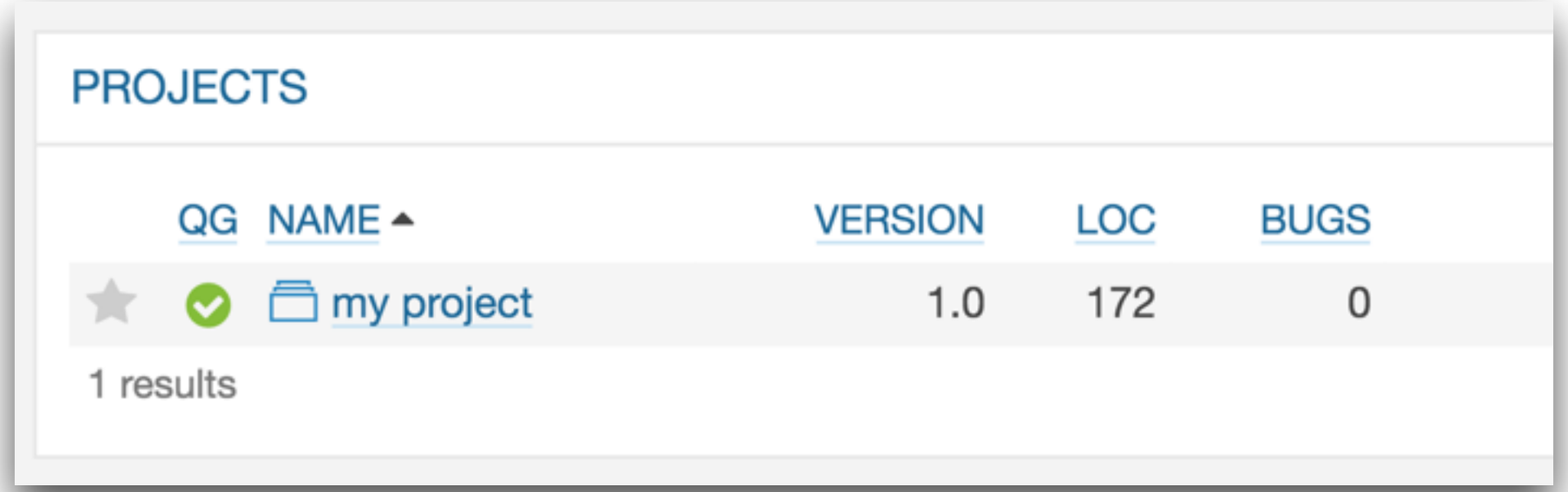
```
# must be unique in a given SonarQube instance
sonar.projectKey=my:project
# this is the name and version displayed in the SonarQube UI. Was
sonar.projectName=My project
sonar.projectVersion=1.0

# Path is relative to the sonar-project.properties file. Replace '
# This property is optional if sonar.modules is set.
sonar.sources=.

# Encoding of the source code. Default is default system encoding
#sonar.sourceEncoding=UTF-8
```

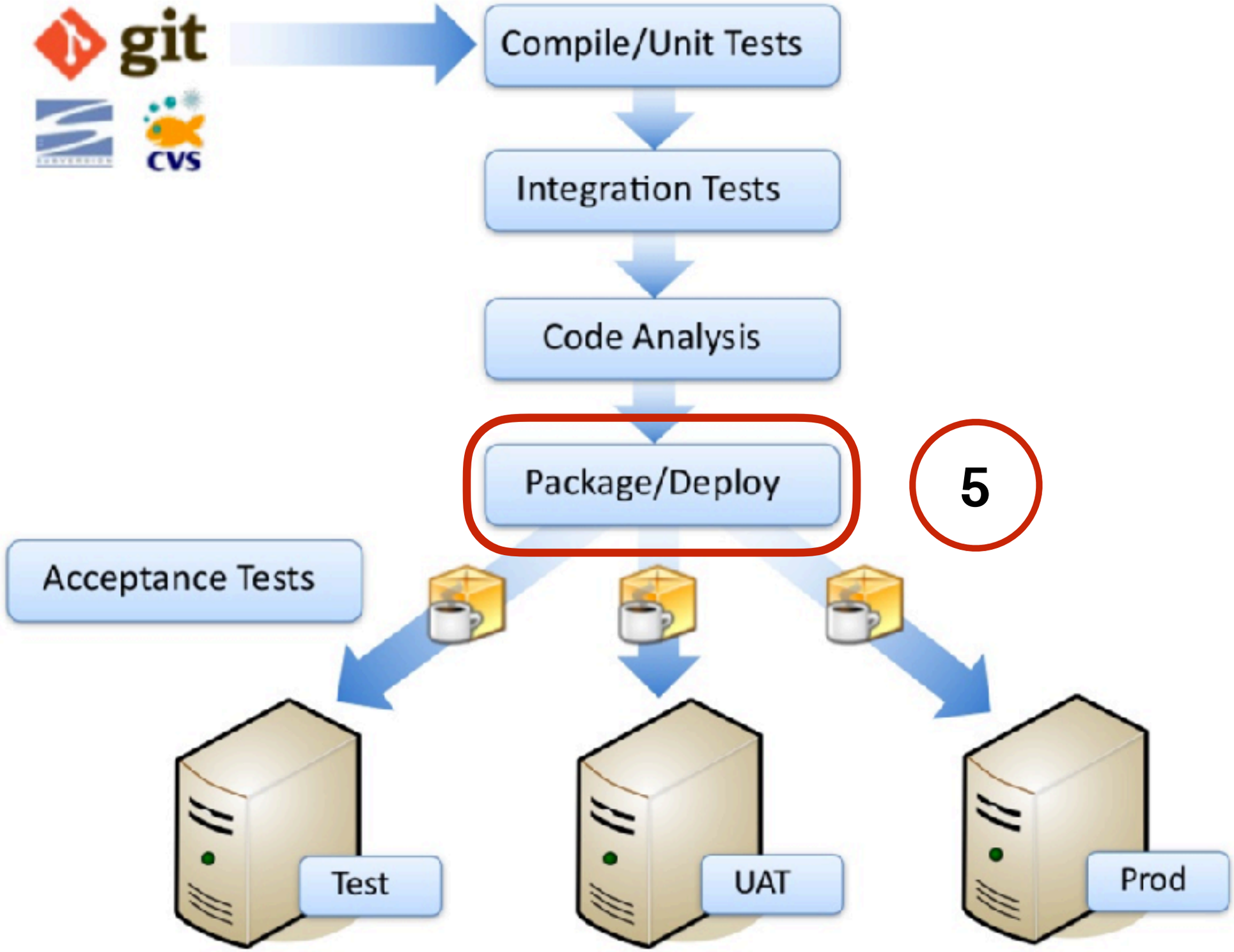
https://github.com/SonarSource/sonar-scanning-examples

Result Jenkins



See result in SonarQube

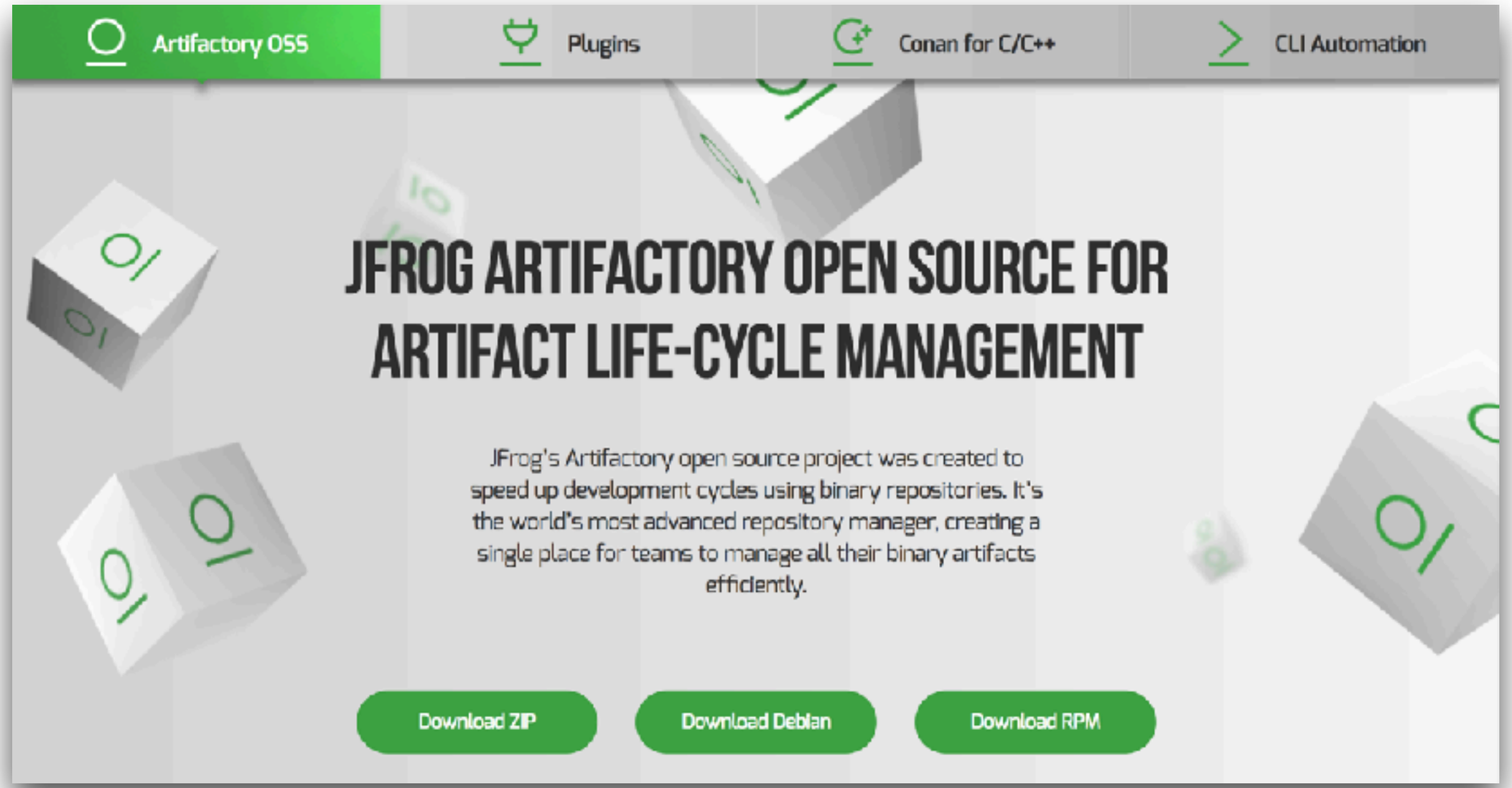



git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
5
Acceptance Tests
Test
UAT
Prod

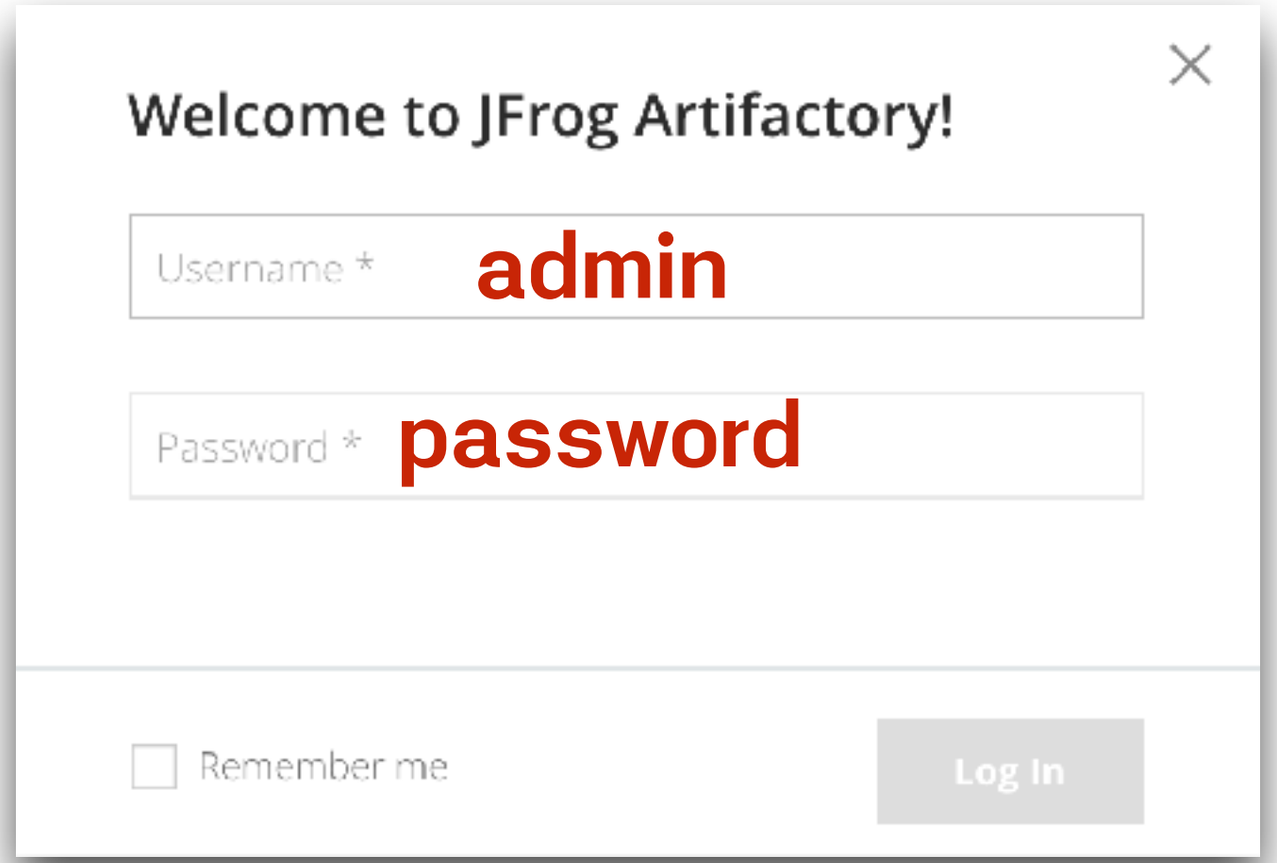
Keep artifact files with JFrog artifactory

Setup JFrog artifactory

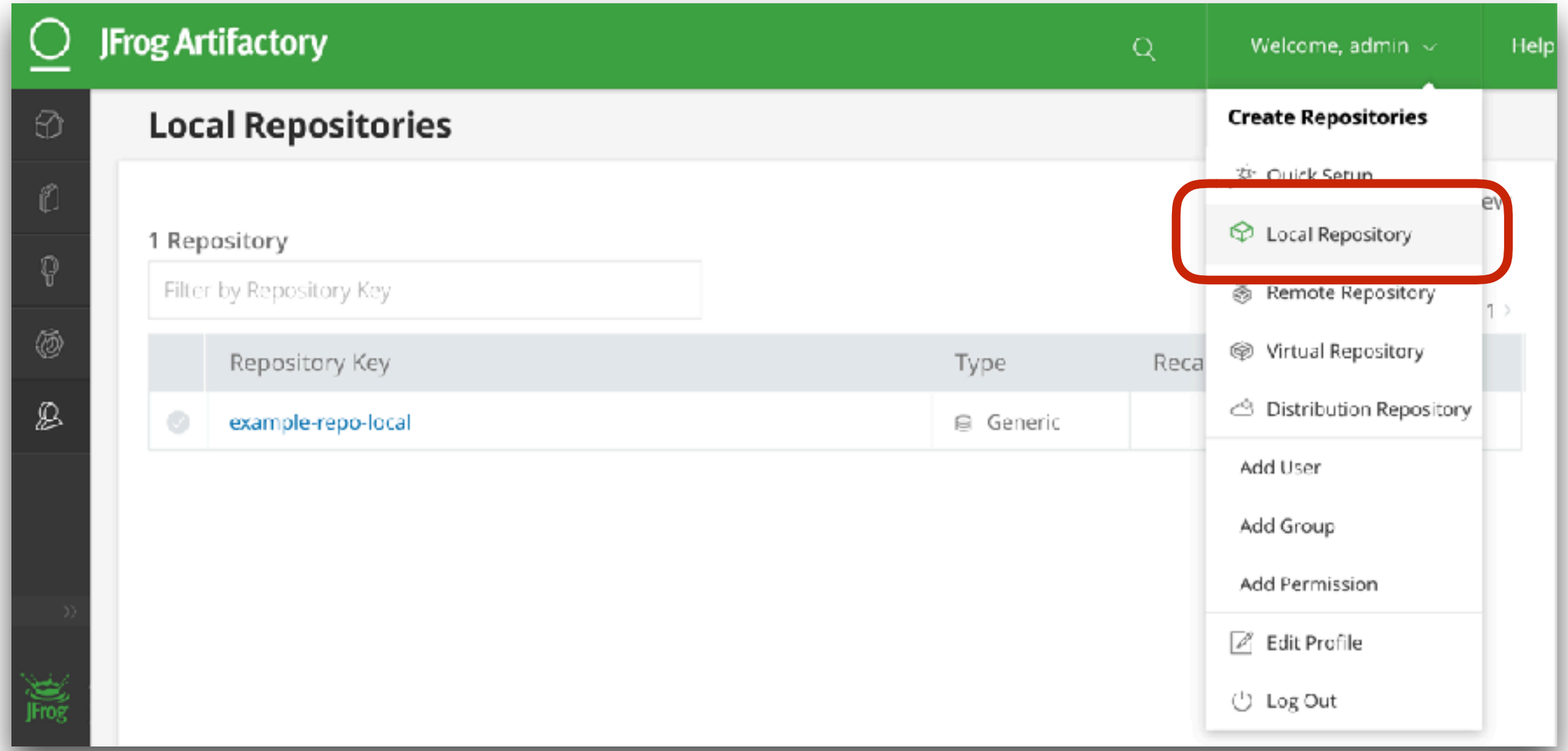
Download from **jfrog.com**



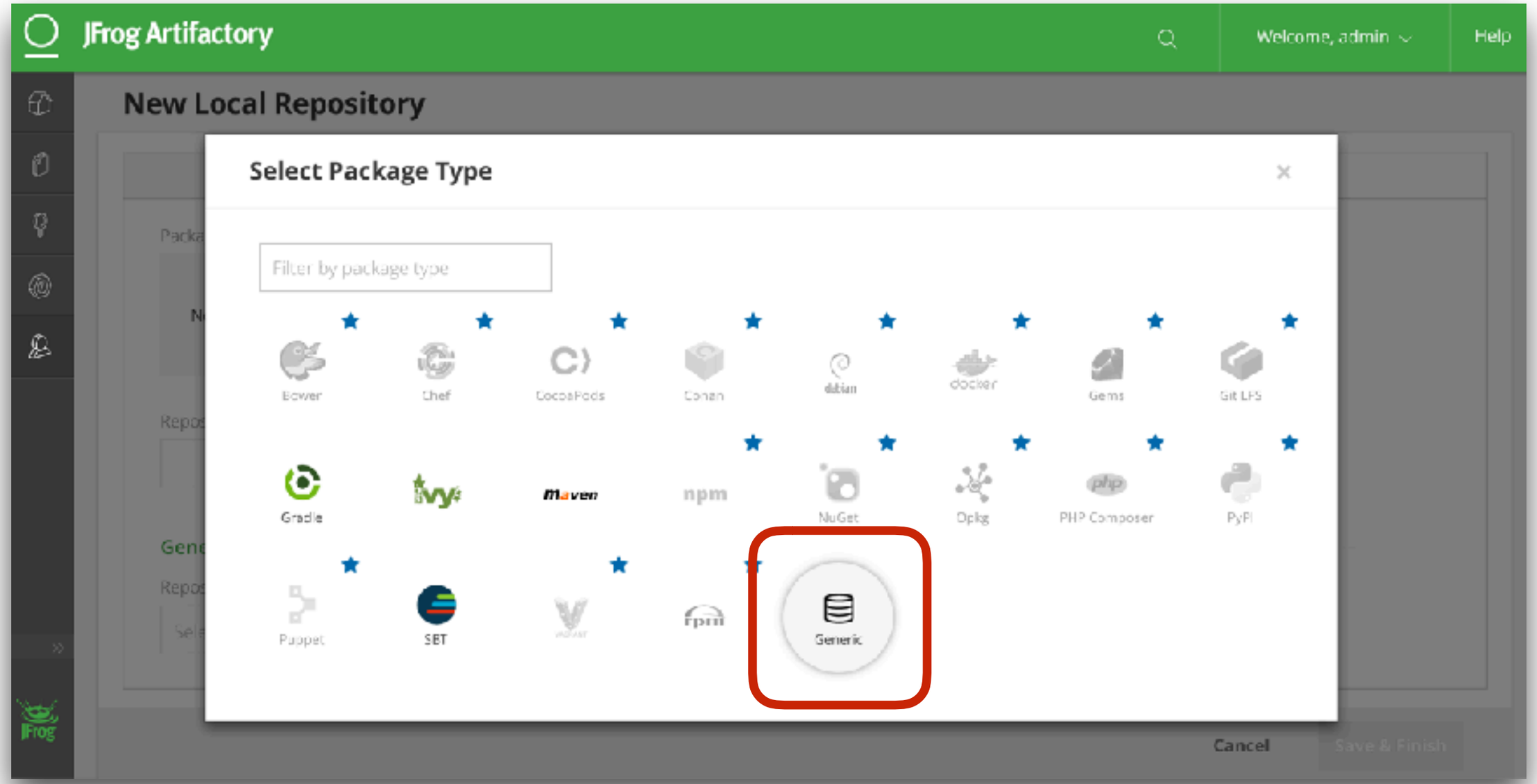
Login as admin



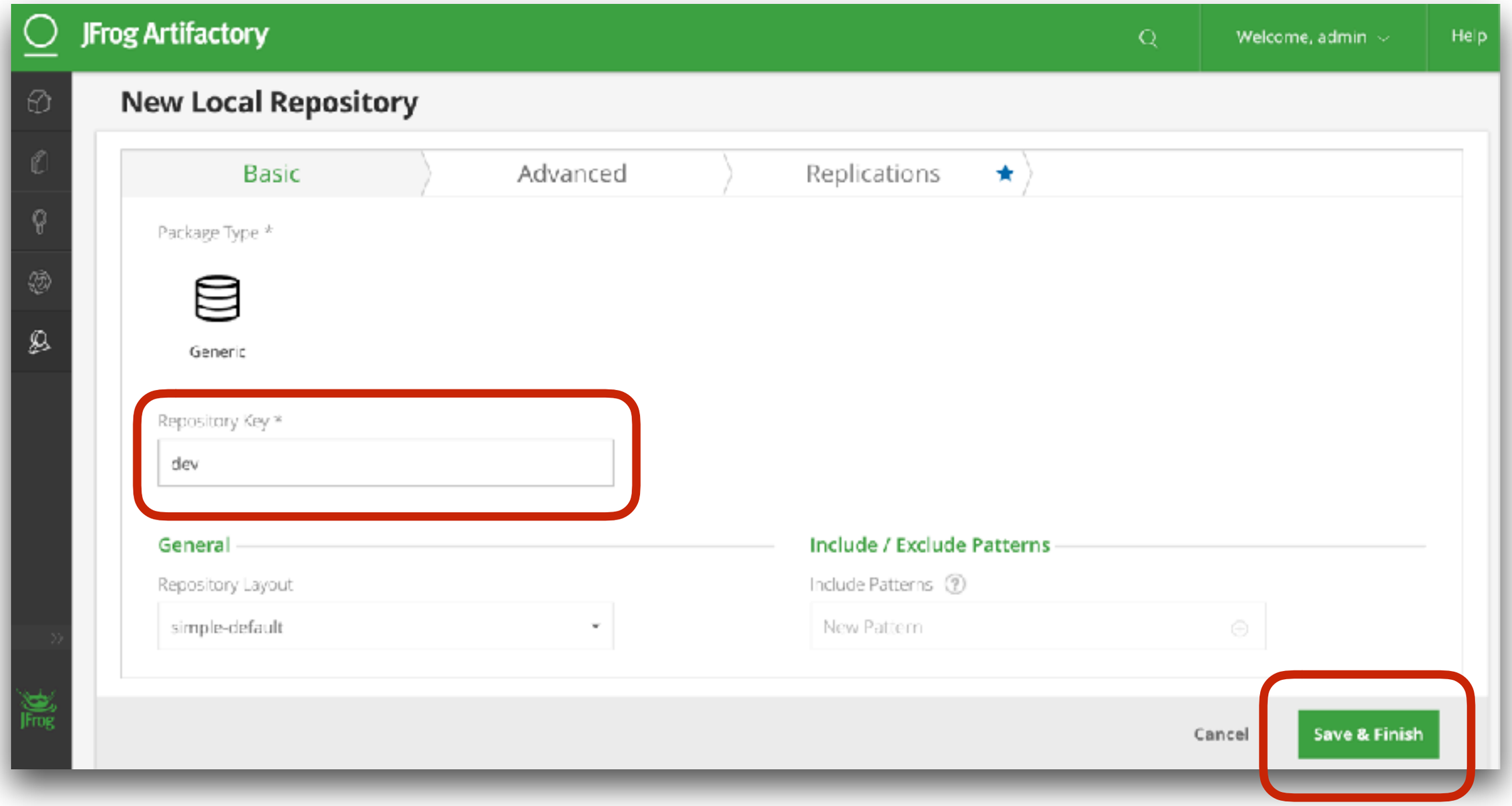
Create local repository



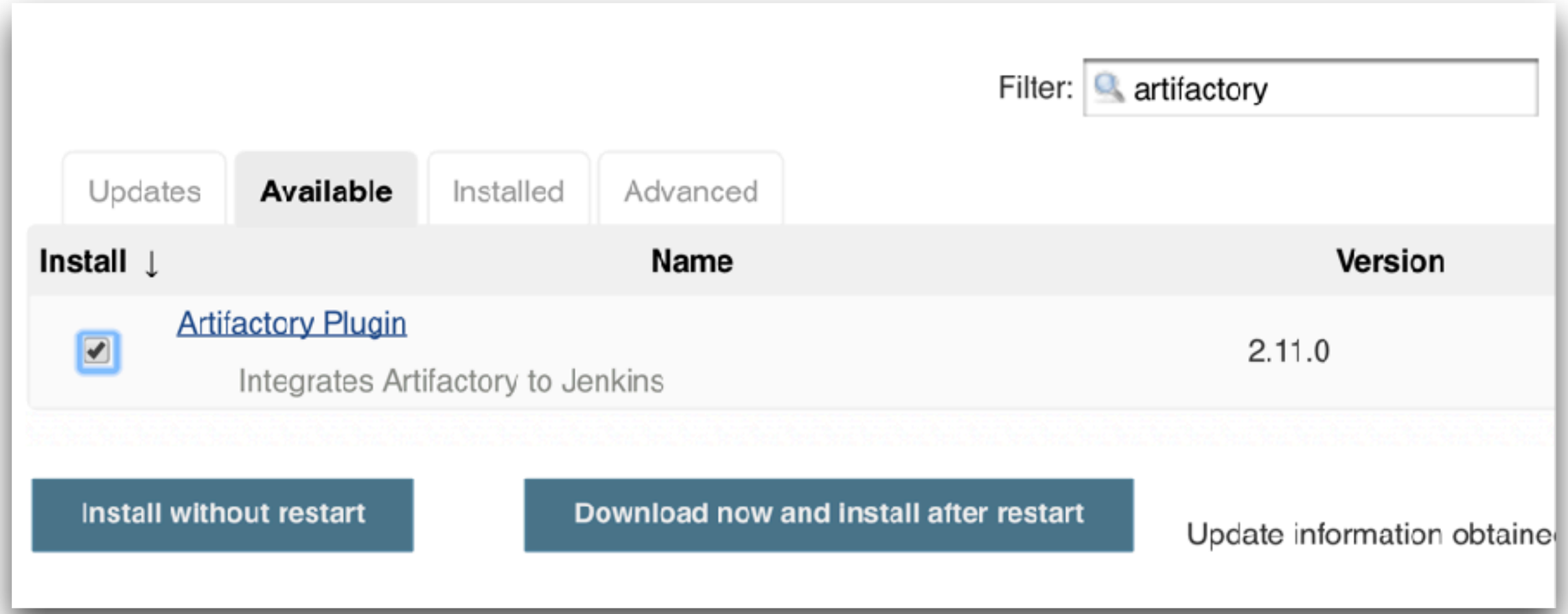
Choose Generic type



Fill in name and save

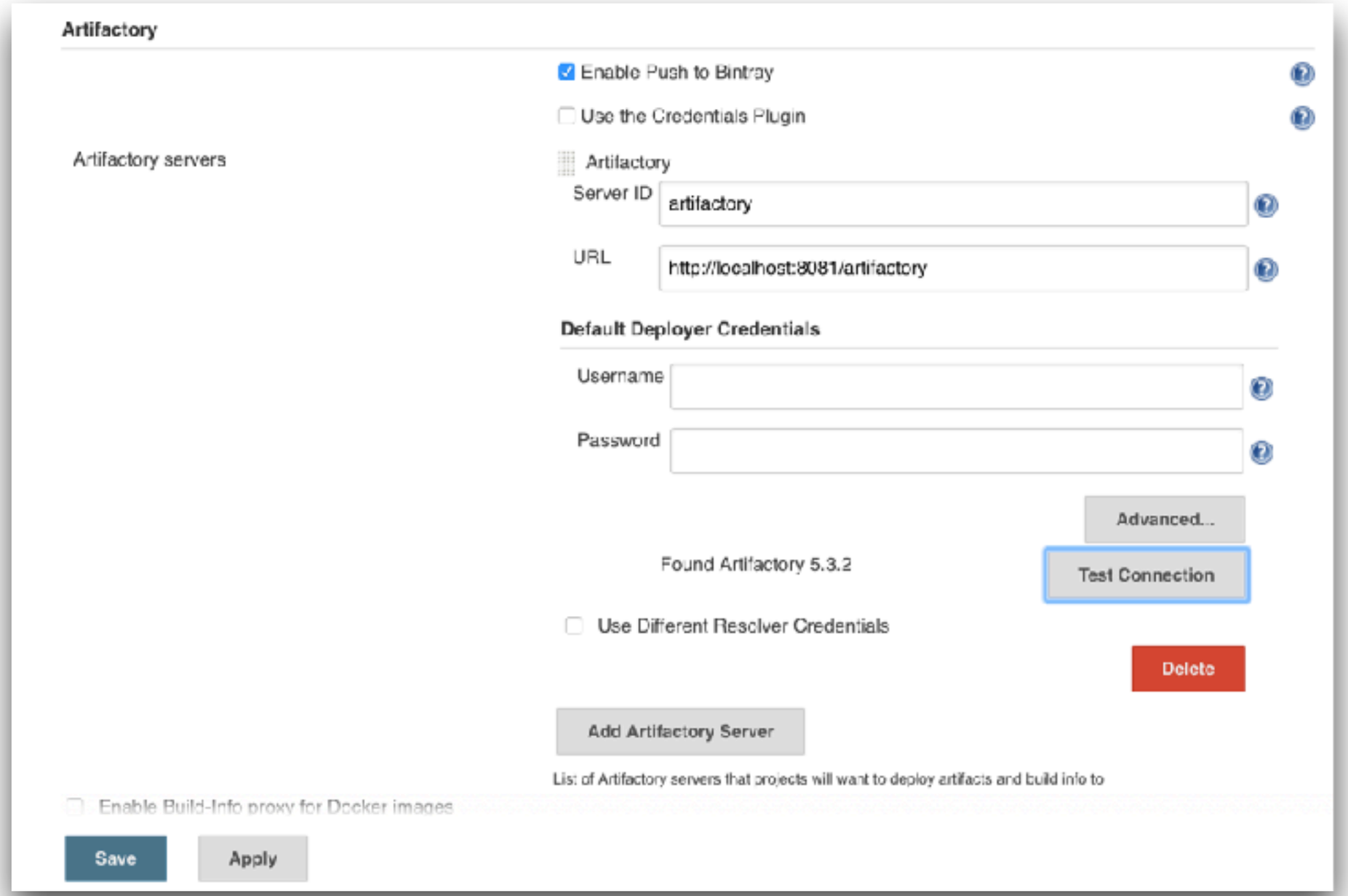


Install Artifactory plugin



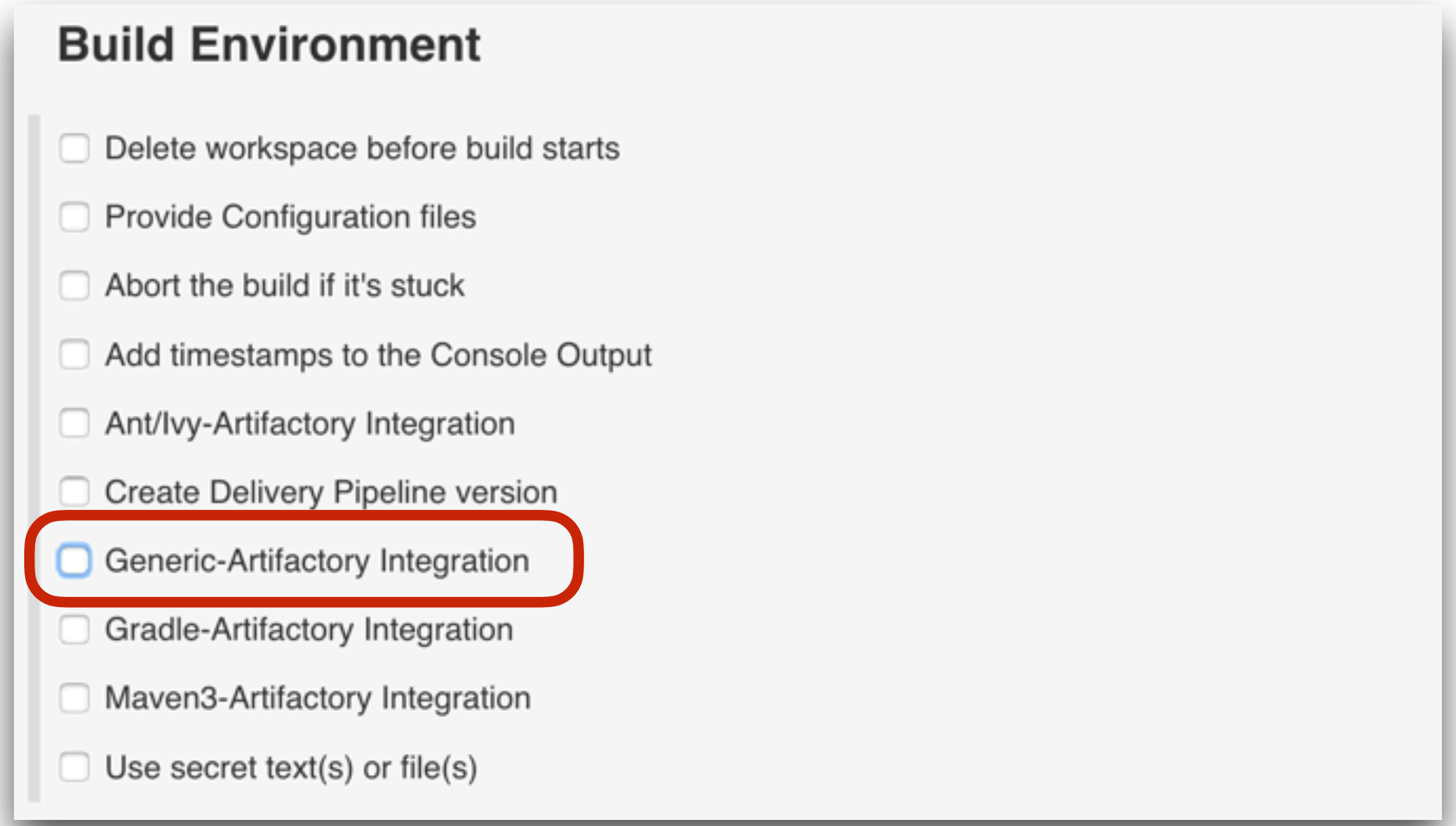
Config Artifactory Server

Manage Jenkins -> Configure System

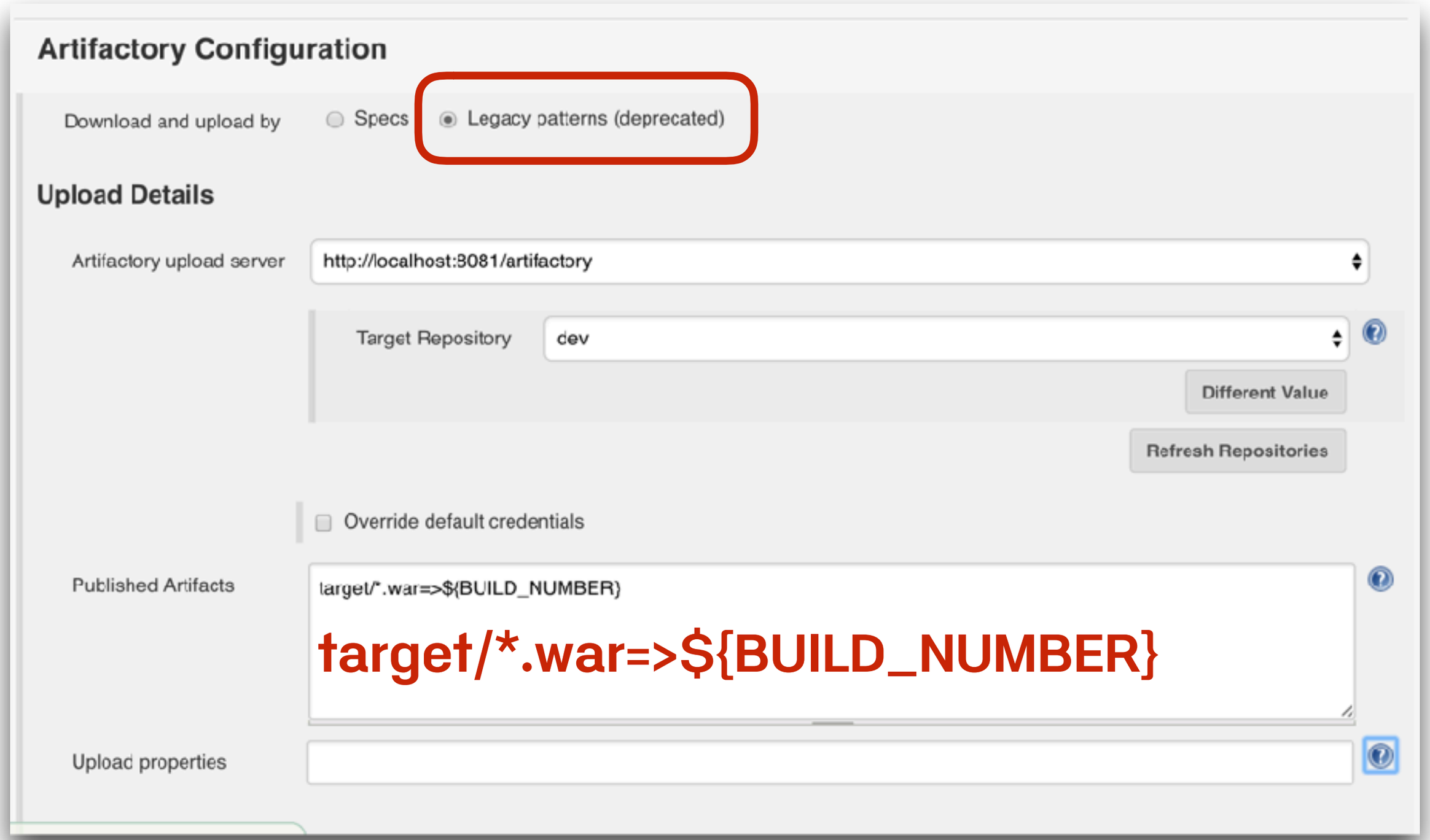


Config job to upload artifactory

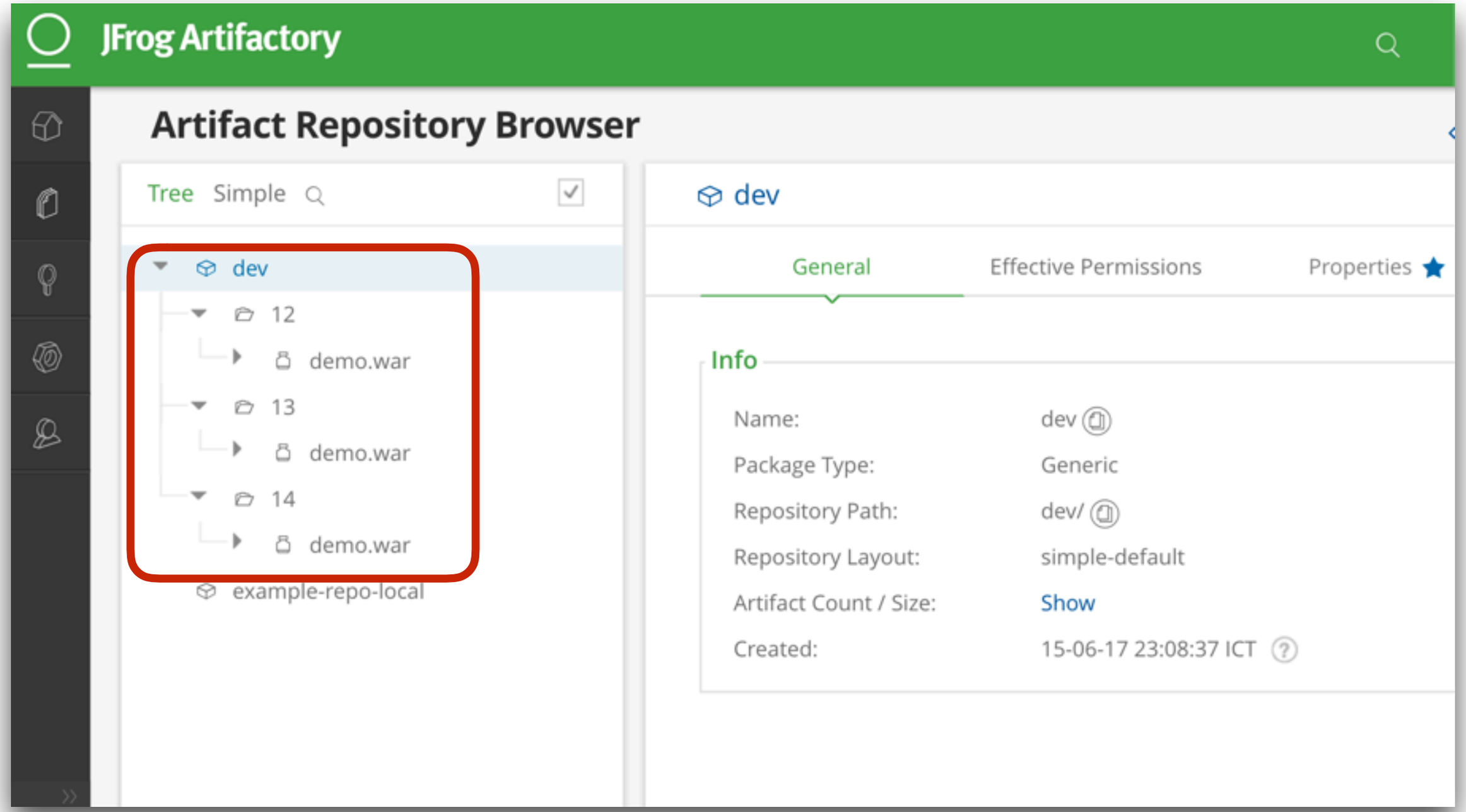
Choose **Generic-Artifactory Integration**



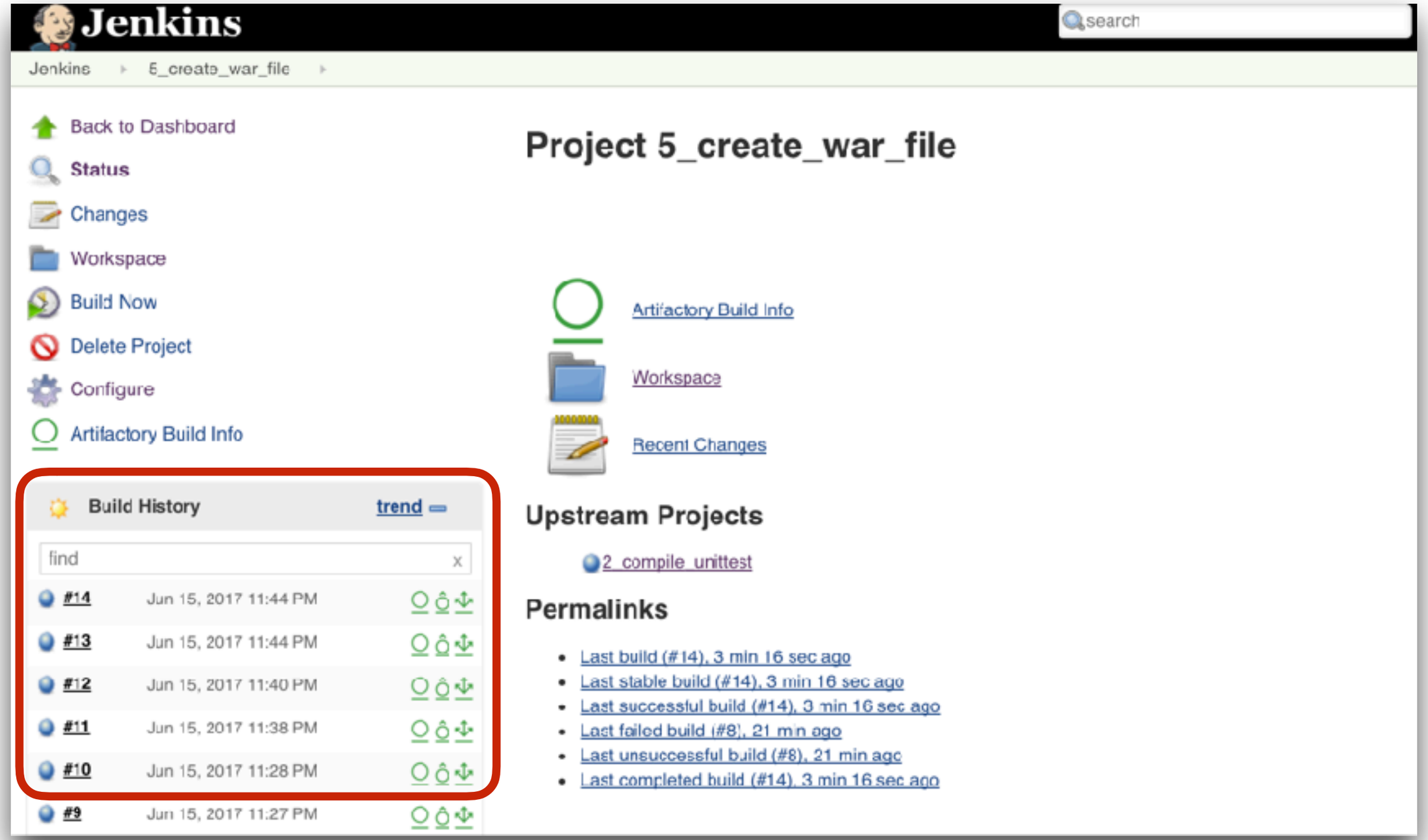
Config job to upload artifactory



Result in Artifactory server

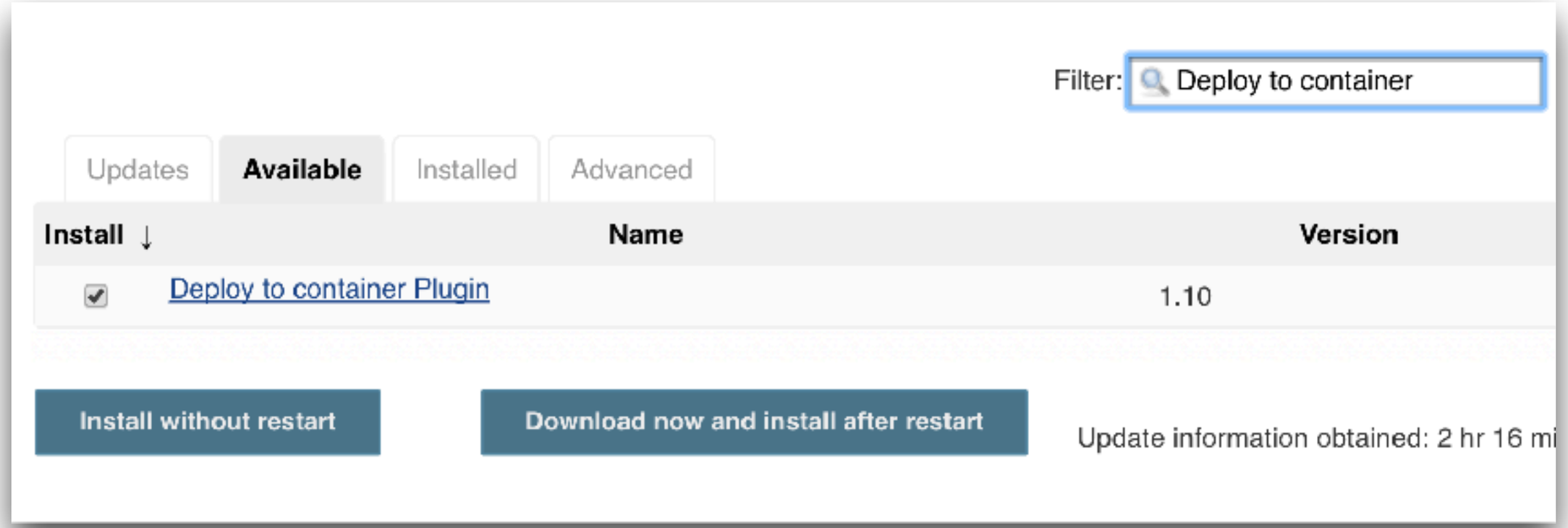


Result Jenkins

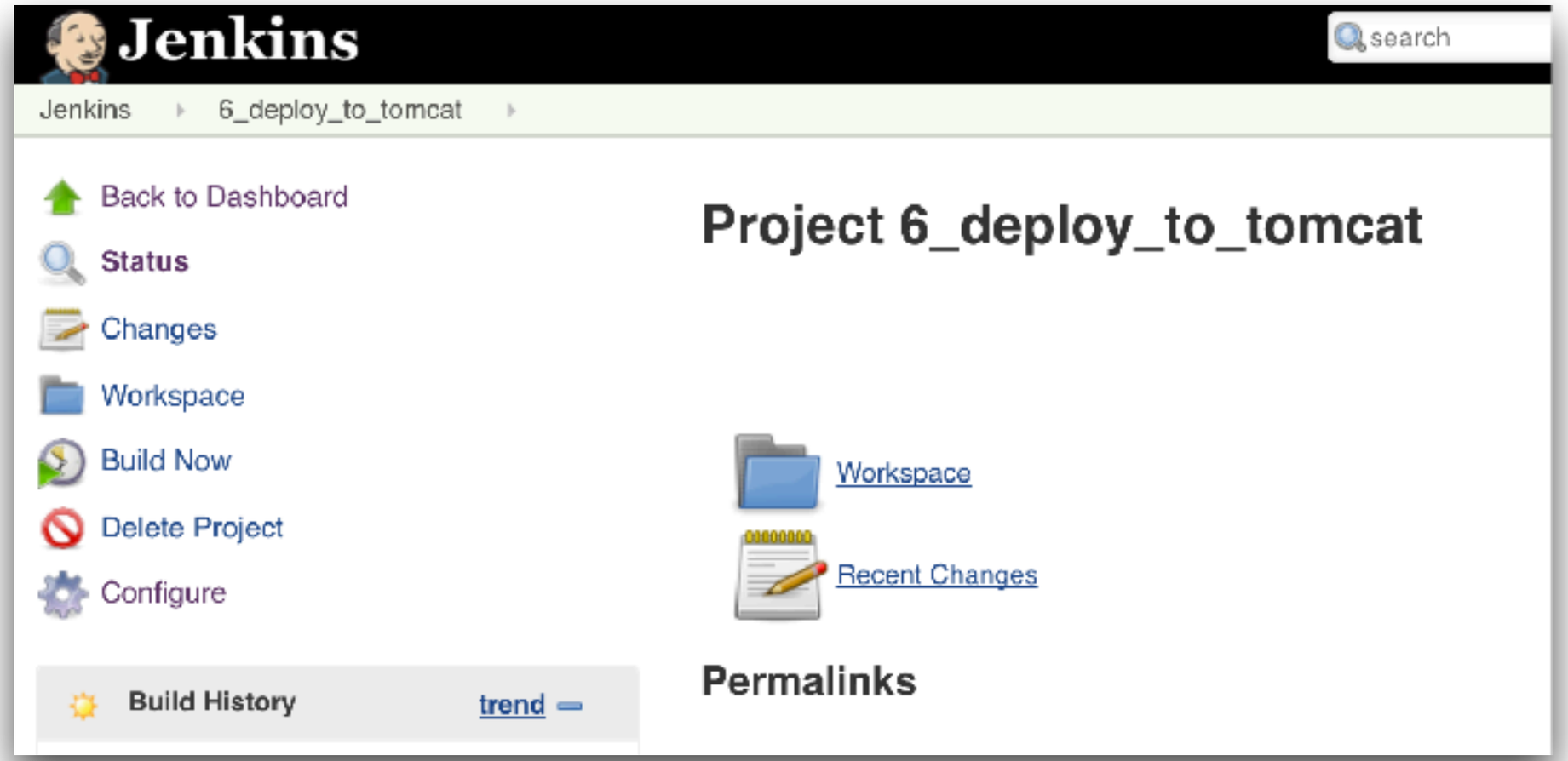


Deploy artifact to Apache Tomcat

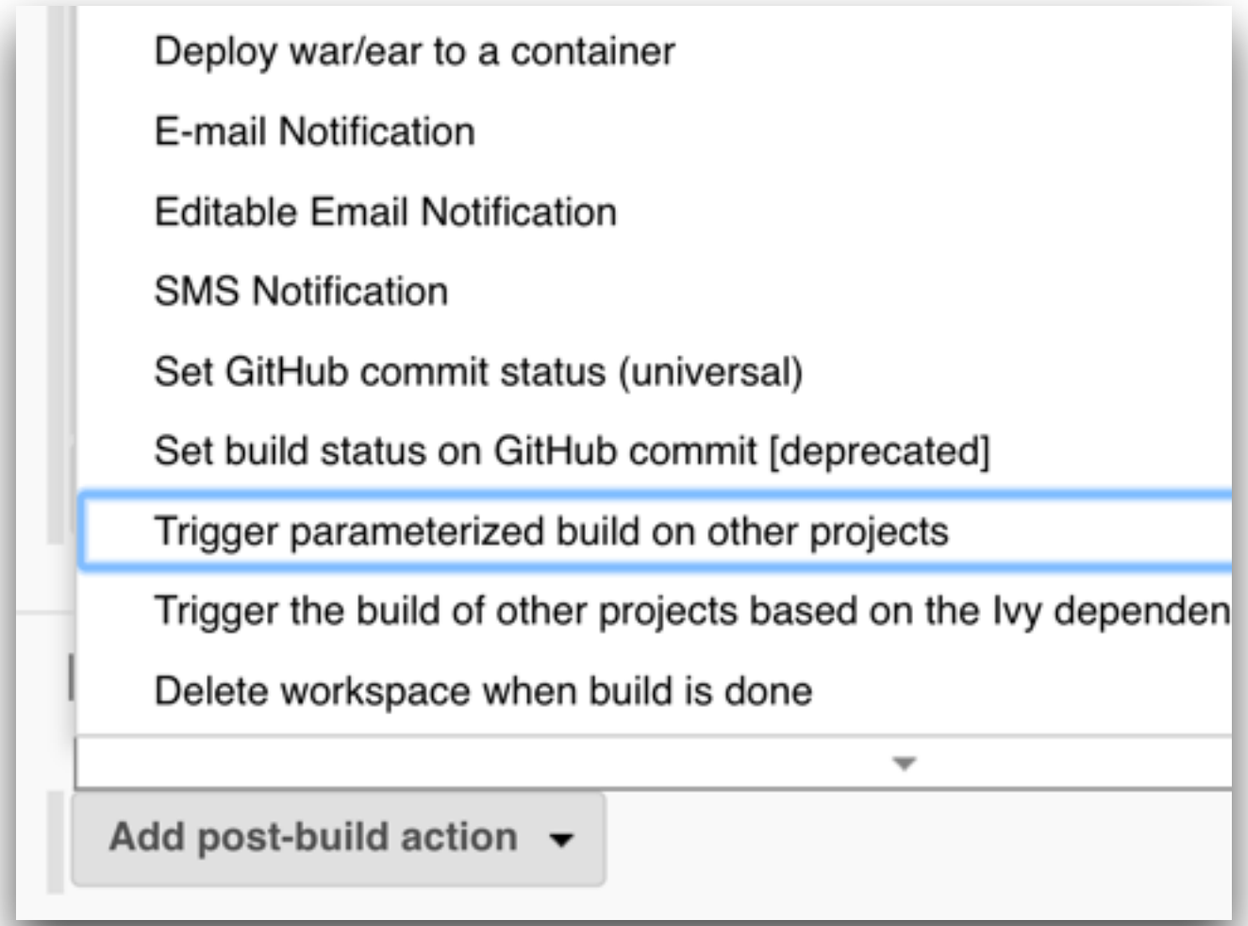
Install Deploy to container plugin



Create new job to deploy

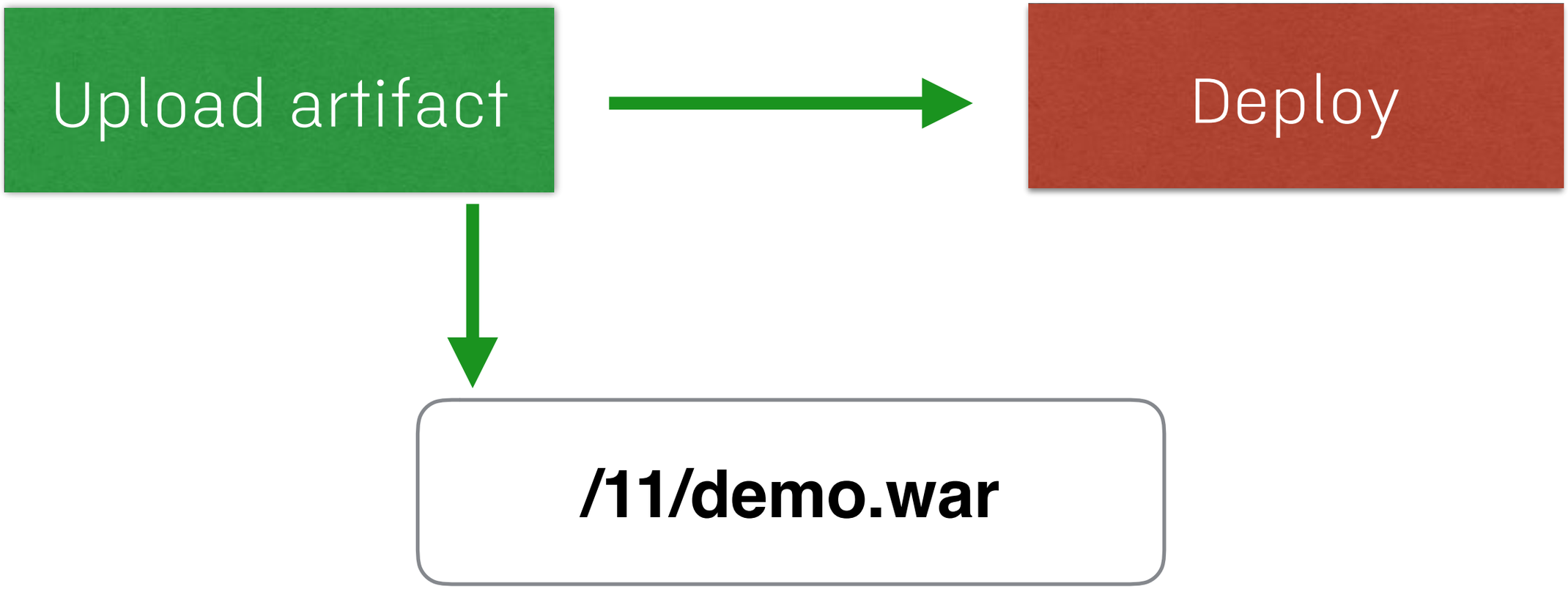


Trigger with parameterized

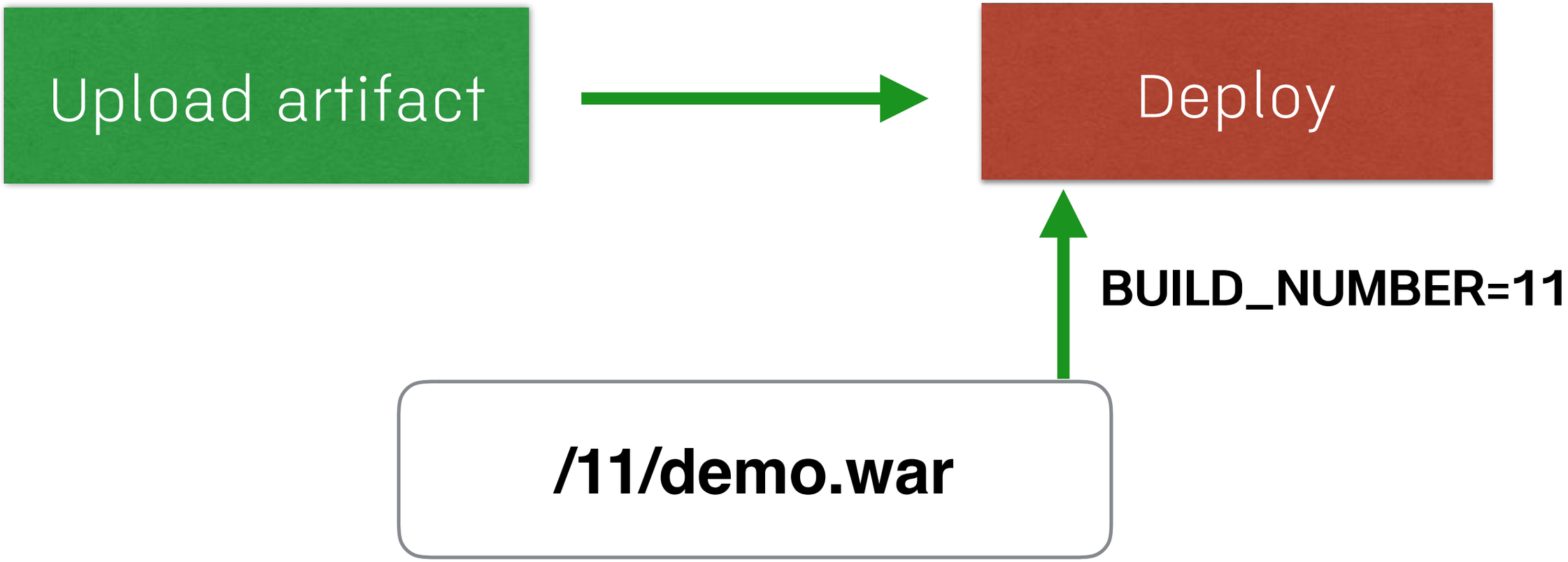


Trigger with parameterized

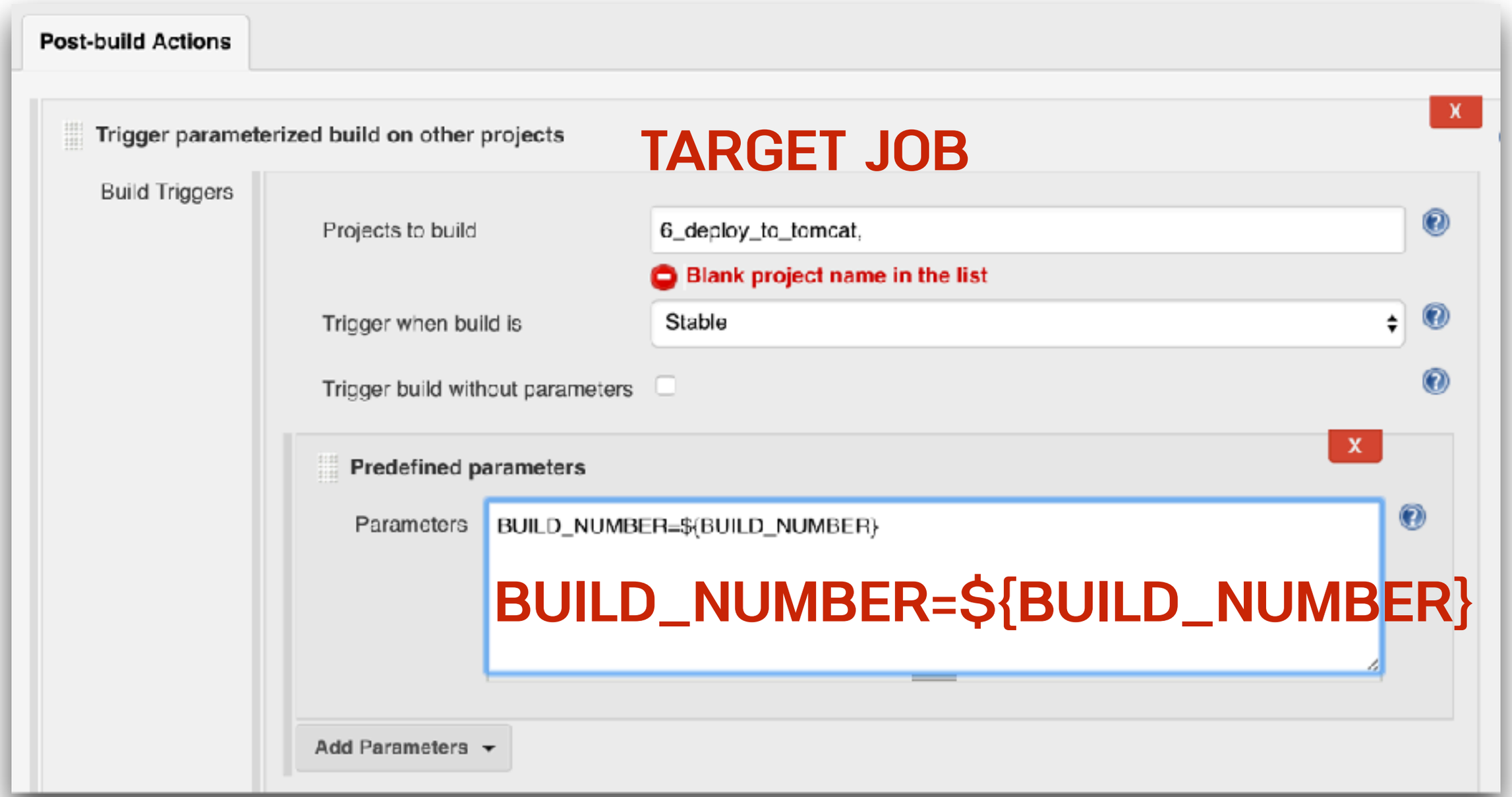
BUILD_NUMBER=11



Trigger with parameterized

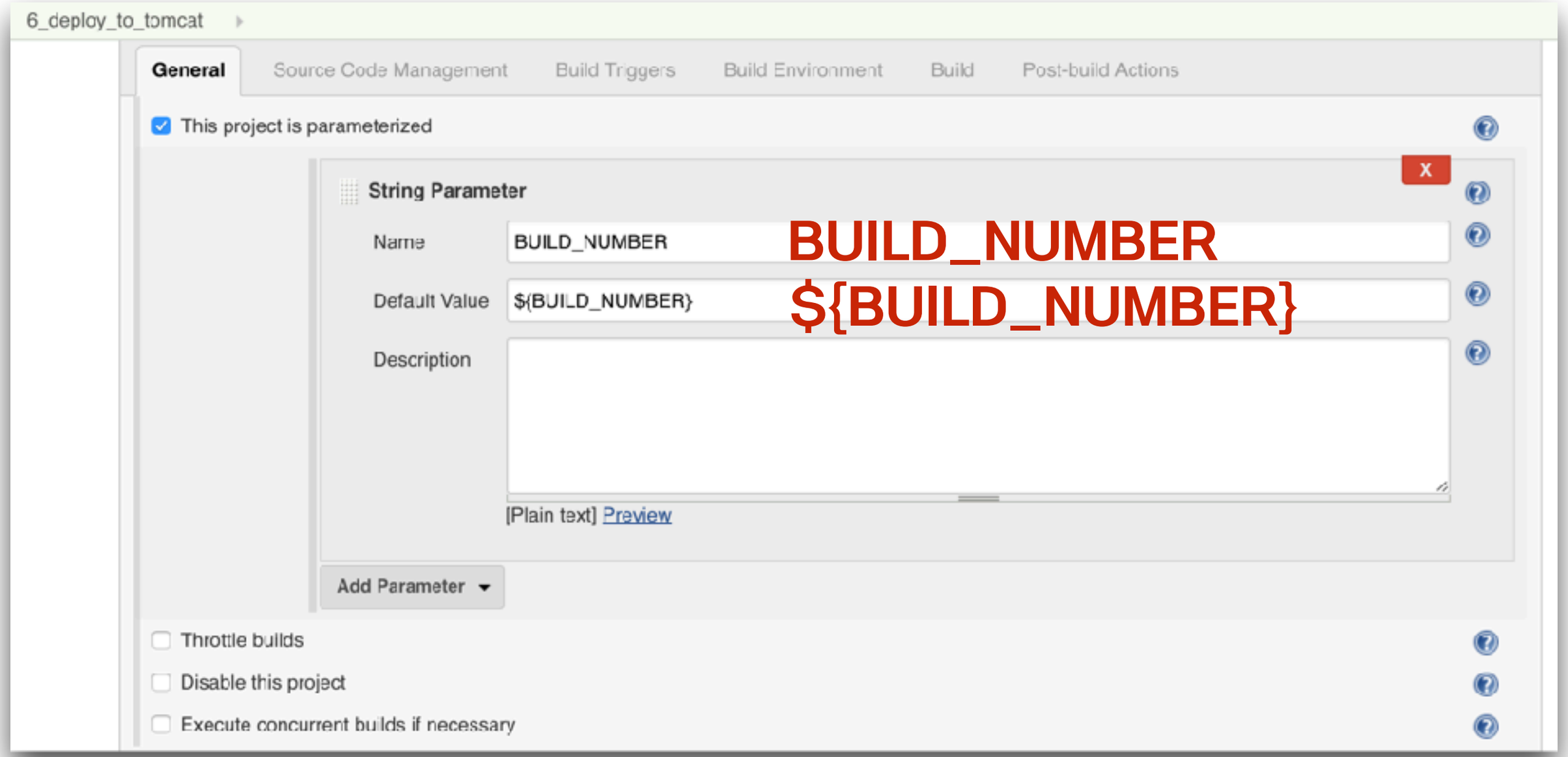


Trigger with parameterized



Download from Artifactory server

Add String parameter



Download artifact file

wget http://localhost:8081/artifactory/dev/$BUILD_NUMBER/demo.war

Wget for windows

http://gnuwin32.sourceforge.net/packages/wget.htm

Wget for Windows

Wget: retrieve files from the WWW

Version

1.11.4

Description

GNU Wget is a free network utility to retrieve files from the World Wide Web using HTTP and FTP, the two most widely used Internet protocols. It works non-interactively, thus enabling work in the background, after having logged off.

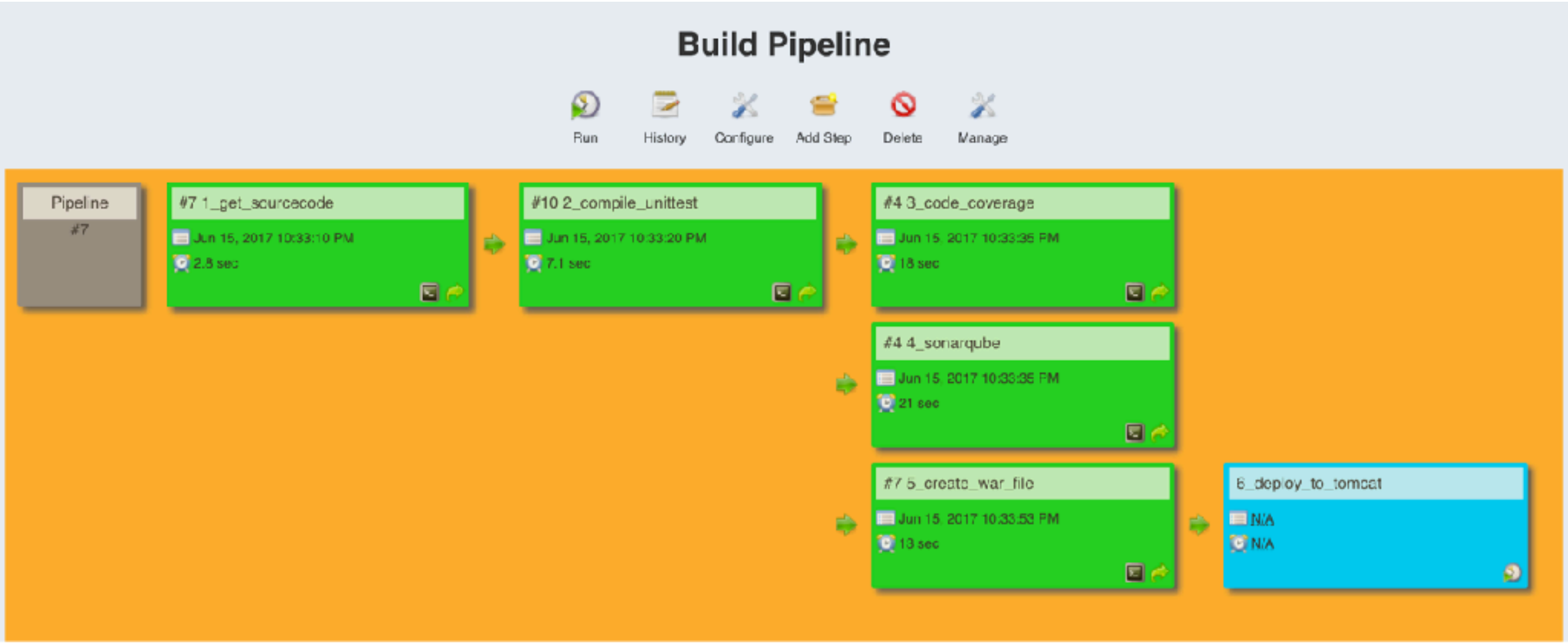
The recursive retrieval of HTML pages, as well as FTP sites is supported -- you can use Wget to make mirrors of archives and home pages, or traverse the web like a WWW robot (Wget understands /robots.txt).

Wget works exceedingly well on slow or unstable connections, keeping getting the document until it is fully retrieved. Re-getting files from where it left off works on servers (both HTTP and FTP) that support it. Matching of wildcards and recursive mirroring of directories are available when retrieving via FTP. Both HTTP and FTP retrievals can be time-stamped, thus Wget can see if the remote file has changed since last retrieval and automatically retrieve the new version if it has.

Wget supports proxy servers, which can lighten the network load, speed up retrieval and provide access behind firewalls. If you are behind a firewall that requires the use of a socks style gateway, you can get the socks library and compile wget with support for socks.

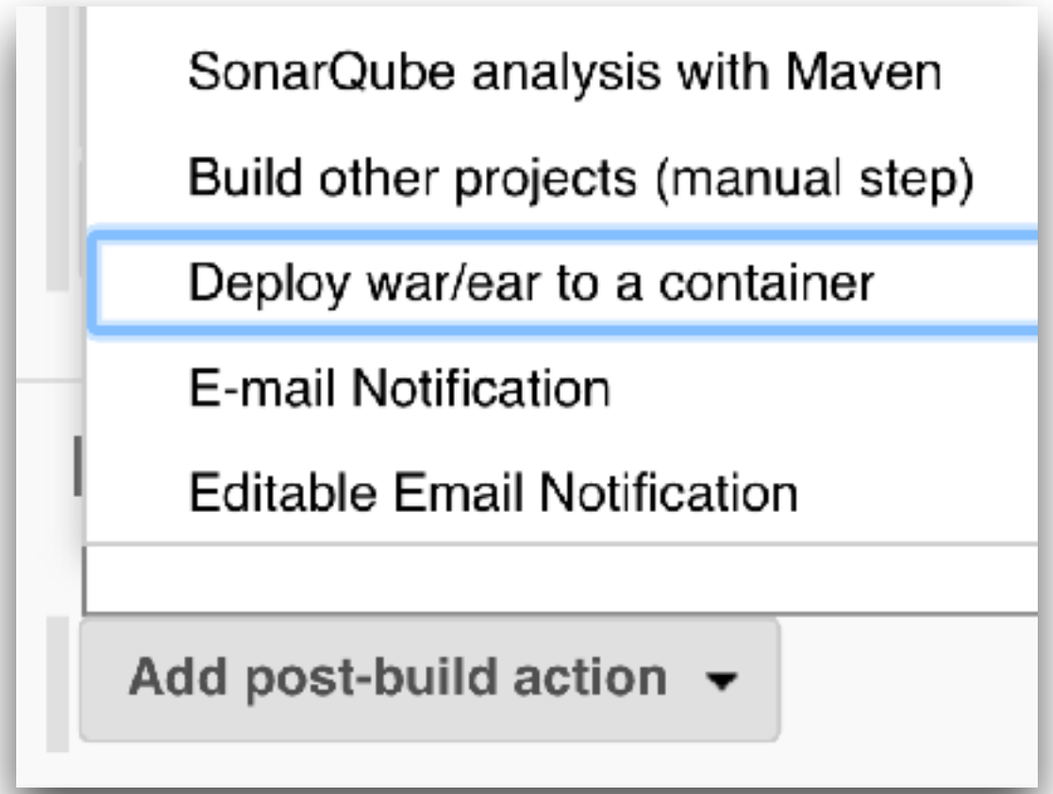
Most of the features are configurable, either through command-line options, or via initialization file .wgetrc. Wget allows you to install a global startup file (etc/wgetrc by default) for site settings.

My build pipeline

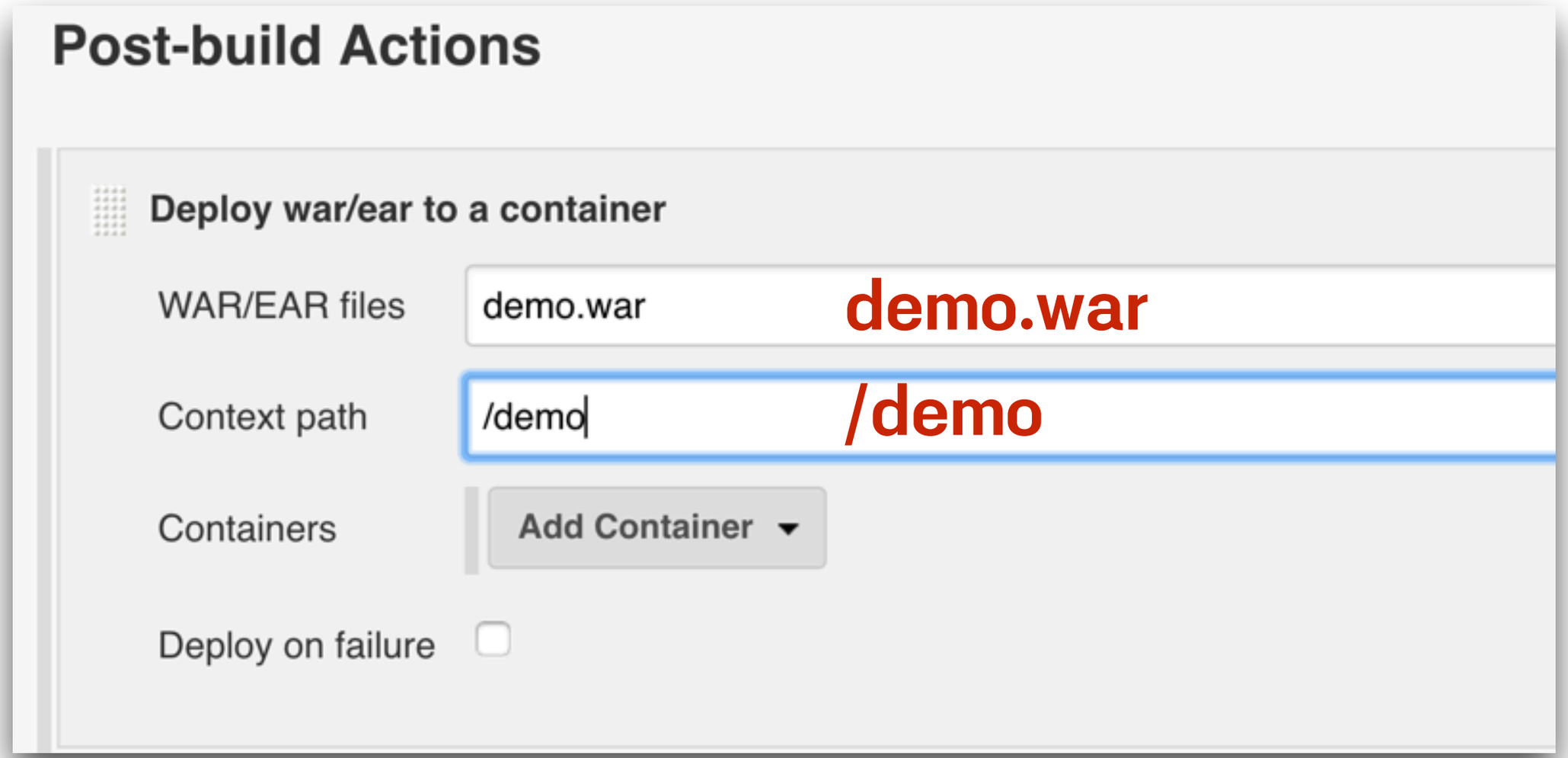


Deploy WAR file to Apache Tomcat

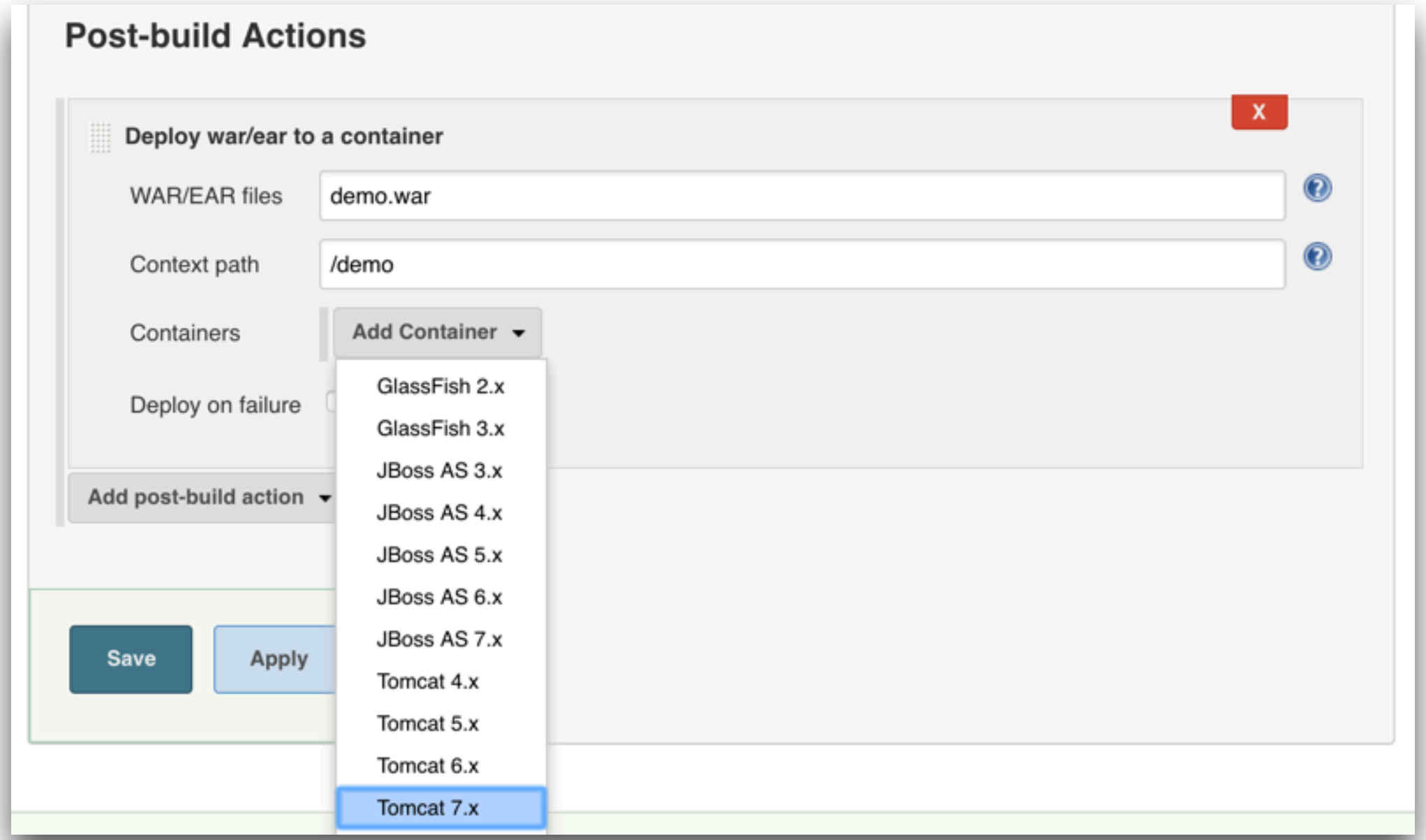
Add post-build action



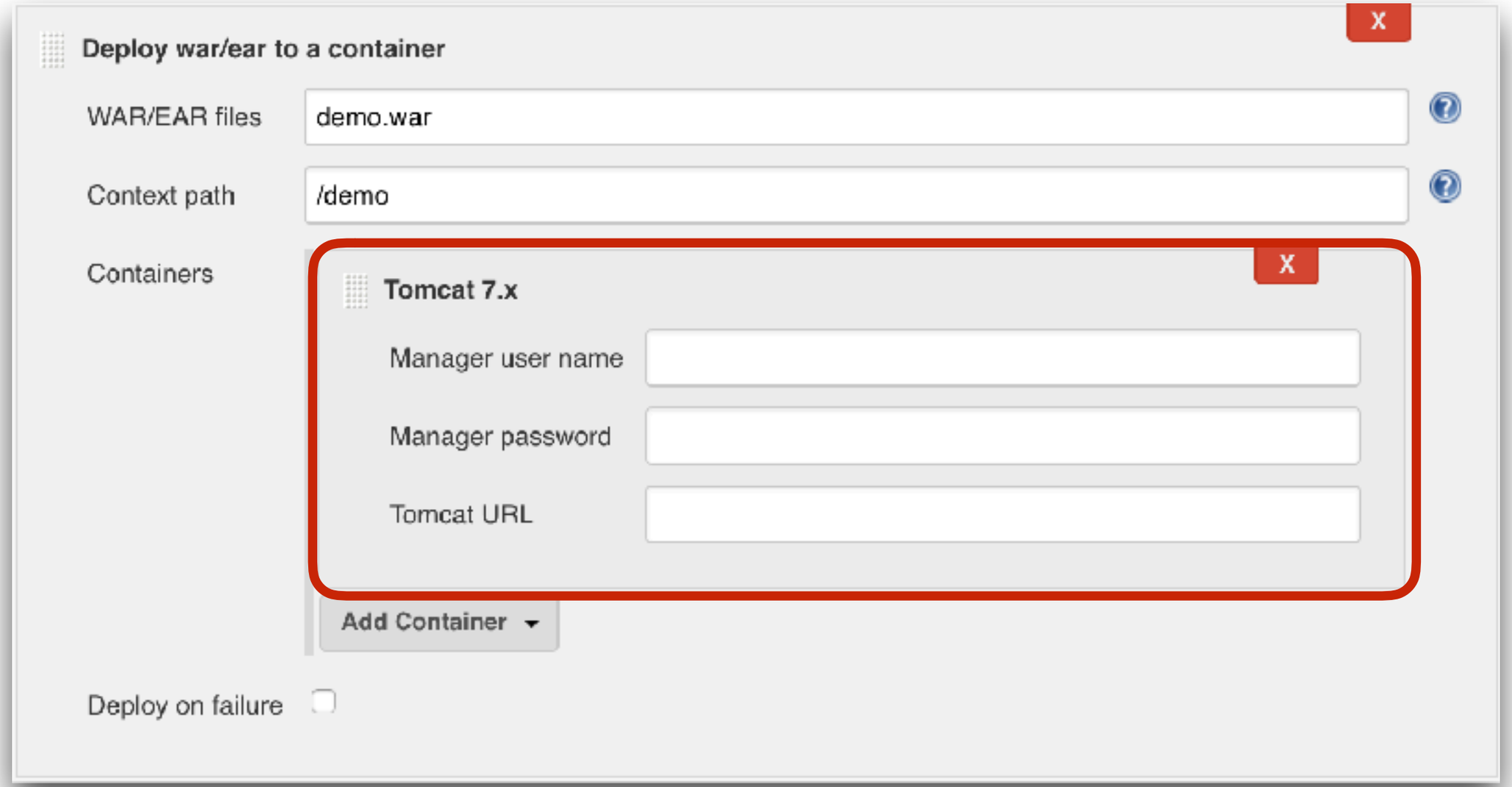
Deploy war/ear to a container



Choose Tomcat container



Config tomcat in jenkins



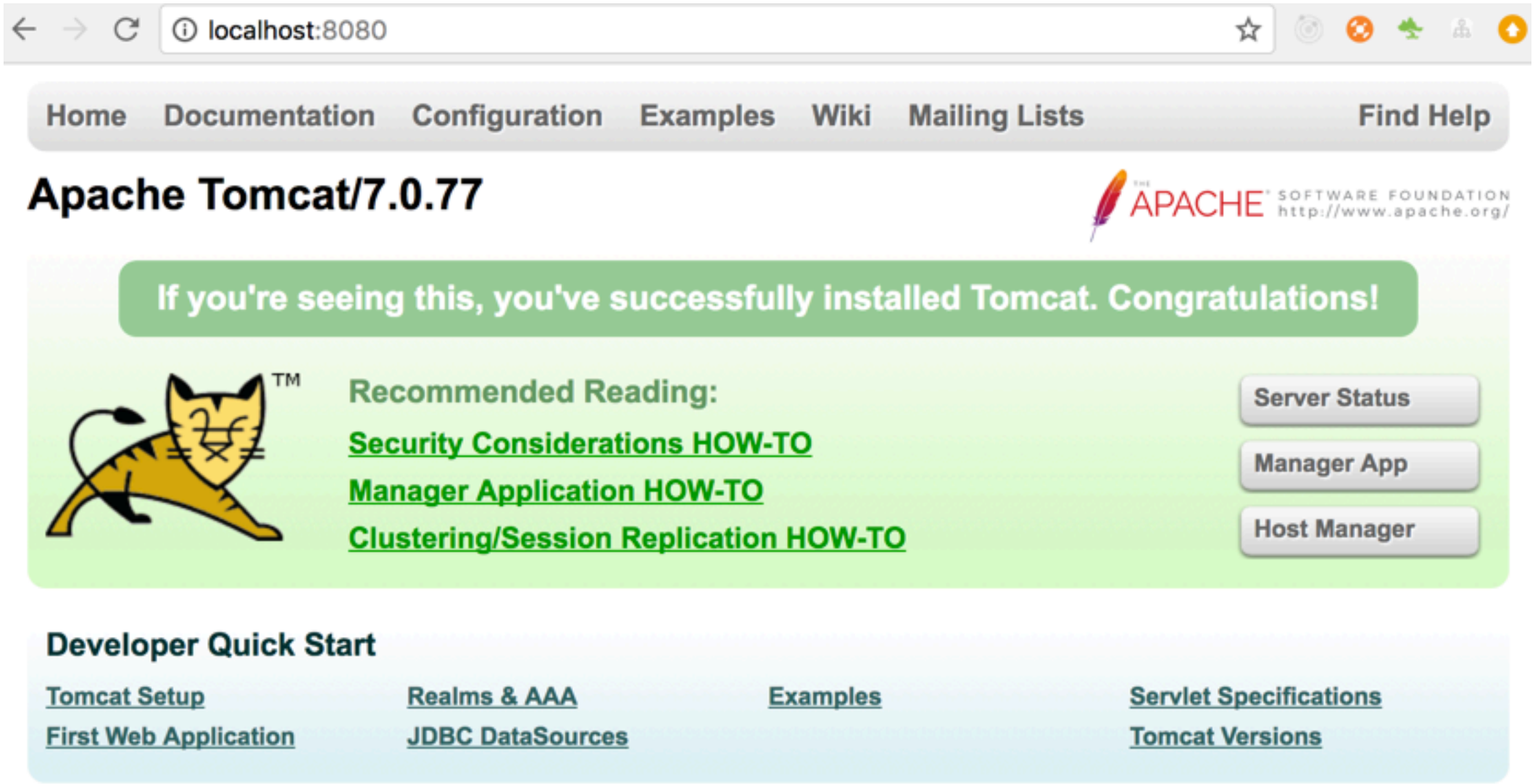
Install Apache Tomcat

Edit tomcat user

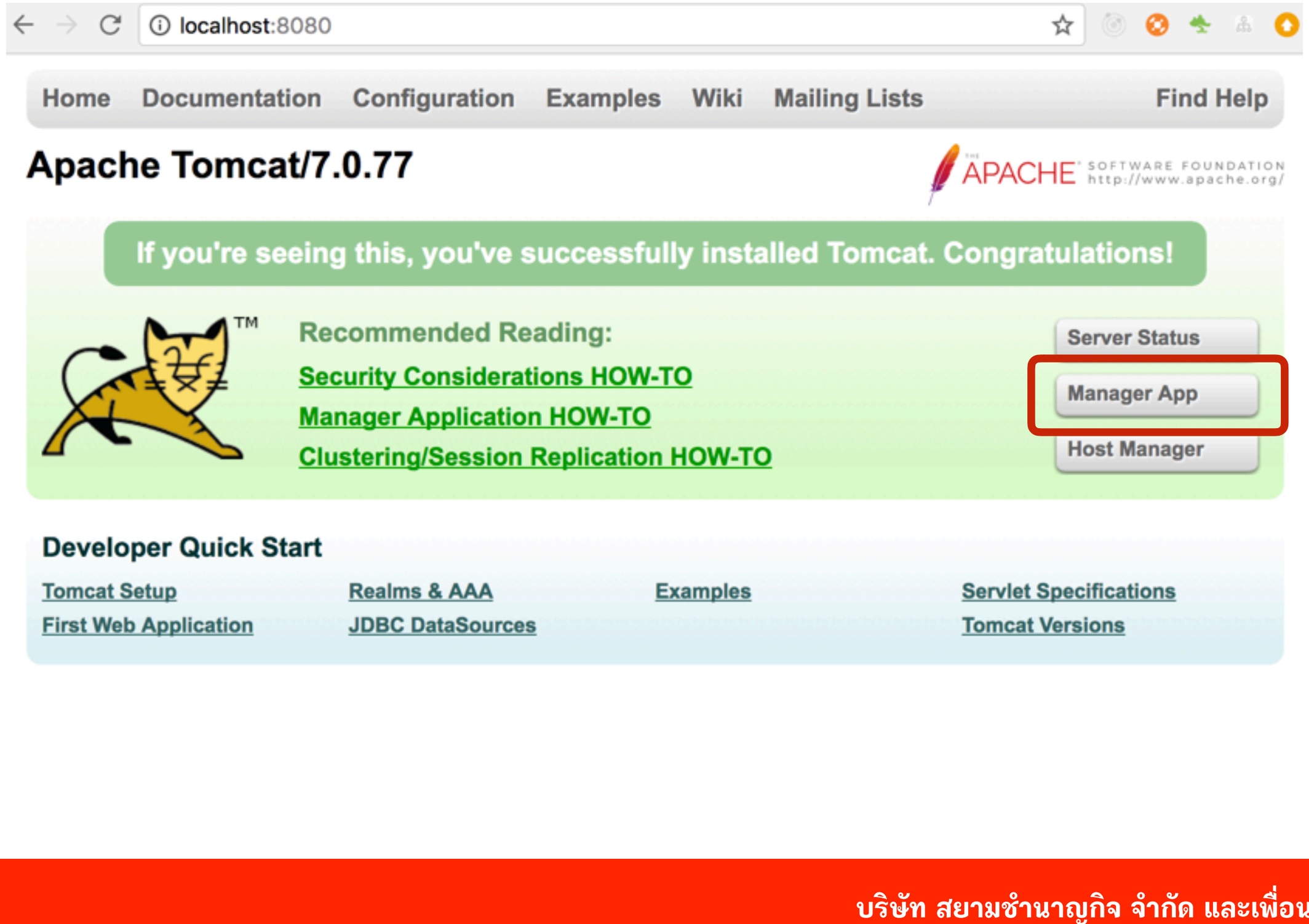
$TOMCAT_HOME/conf/**tomcat-users.xml**

```
<tomcat-users>
  <role rolename="manager"/>
  <role rolename="manager-script"/>
  <role rolename="manager-gui"/>
  <user username="deploy" password="deploy"
    roles="manager,manager-gui,manager-script"/>
</tomcat-users>
```

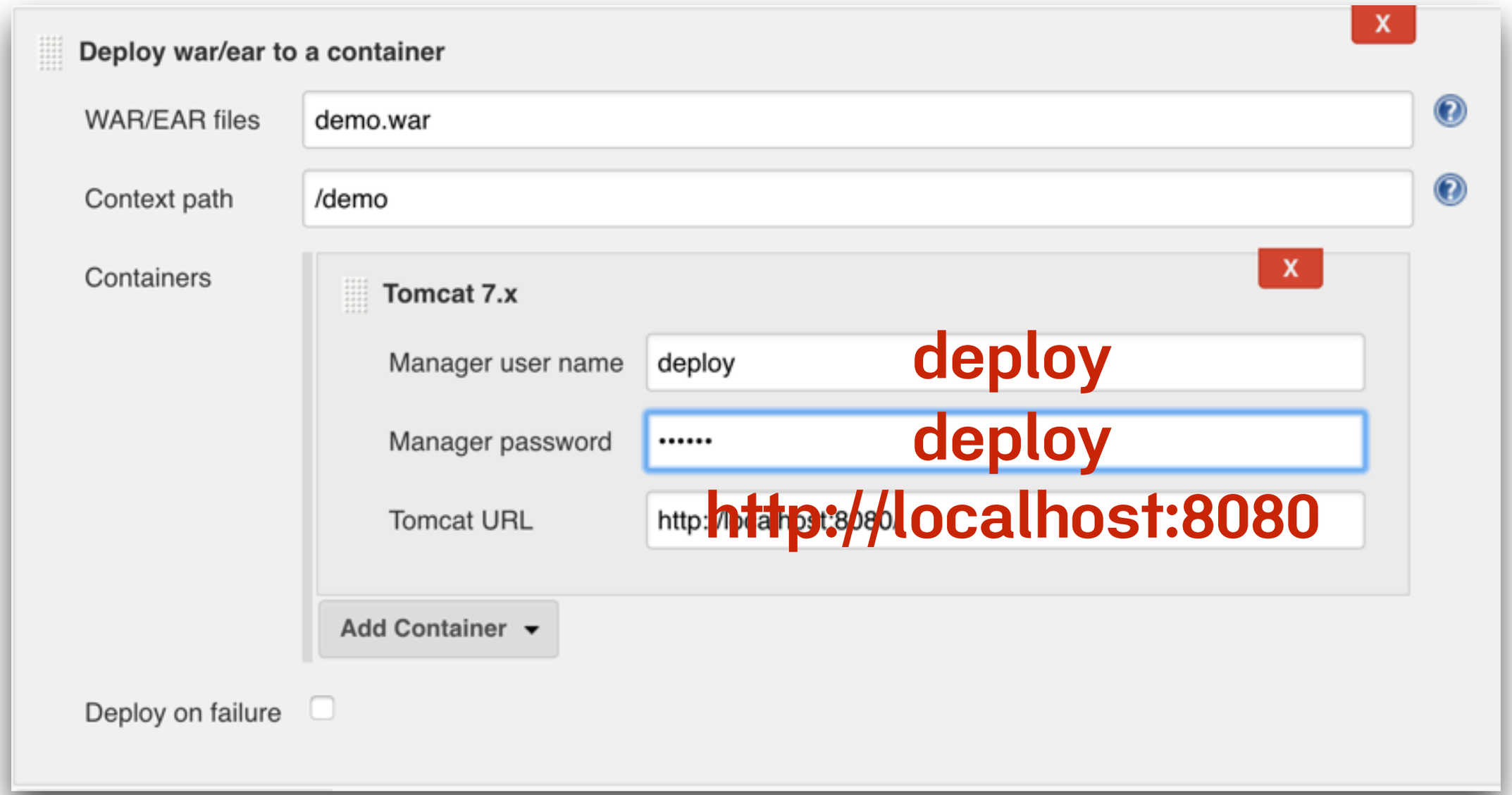
Start Tomcat Server



Try to login


localhost:8080
Home
Documentation
Configuration
Examples
Wiki
Mailing Lists
Find Help
Apache Tomcat/7.0.77
APACHE SOFTWARE FOUNDATION http://www.apache.org/
If you're seeing this, you've successfully installed Tomcat. Congratulations!
Recommended Reading:
Security Considerations HOW-TO
Manager Application HOW-TO
Clustering/Session Replication HOW-TO
Server Status
Manager App
Host Manager
Developer Quick Start
Tomcat Setup
First Web Application
Realms & AAA
JDBC DataSources
Examples
Servlet Specifications
Tomcat Versions

Config jenkins again



Try to build !!

Jenkins ▸ 6_deploy_to_tomcat ▸ #5

- Back to Project
- Status
- Changes
- Console Output
  - View as plain text
- Edit Build Information
- Delete Build
- Parameters
- Previous Build

Console Output

```
Started by upstream project "5_create_war_file" build number 19
originally caused by:
 Started by user Somkiat
Building in workspace /Users/somkiat/data/slide/ci-
cd/swpark/software/keep/workspace/6_deploy_to_tomcat
[6_deploy_to_tomcat] $ /bin/sh -xe
/var/folders/t5/8kg23s_97z9dw44tfcld6dqw0000gn/T/hudson2611444727417694445.sh
+ wget http://localhost:8081/artifactory/dev/19/demo.war
--2017-06-16 01:07:44--  http://localhost:8081/artifactory/dev/19/demo.war
Resolving localhost... ::1, 127.0.0.1
Connecting to localhost|::1|:8081... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10879 (11K) [application/java-archive]
Saving to: 'demo.war.2'

     OK ..........                                            100% 76.3M=0s

2017-06-16 01:07:44 (76.3 MB/s) - 'demo.war.2' saved [10879/10879]

Deploying /Users/somkiat/data/slide/ci-
cd/swpark/software/keep/workspace/6_deploy_to_tomcat/demo.war to container Tomcat 7.x Remote
  [/Users/somkiat/data/slide/ci-cd/swpark/software/keep/workspace/6_deploy_to_tomcat/demo.war] is
not deployed. Doing a fresh deployment.
  Deploying [/Users/somkiat/data/slide/ci-
cd/swpark/software/keep/workspace/6_deploy_to_tomcat/demo.war]
Finished: SUCCESS
```

Check result on Tomcat

localhost:8080/manager/html

THE APACHE SOFTWARE FOUNDATION

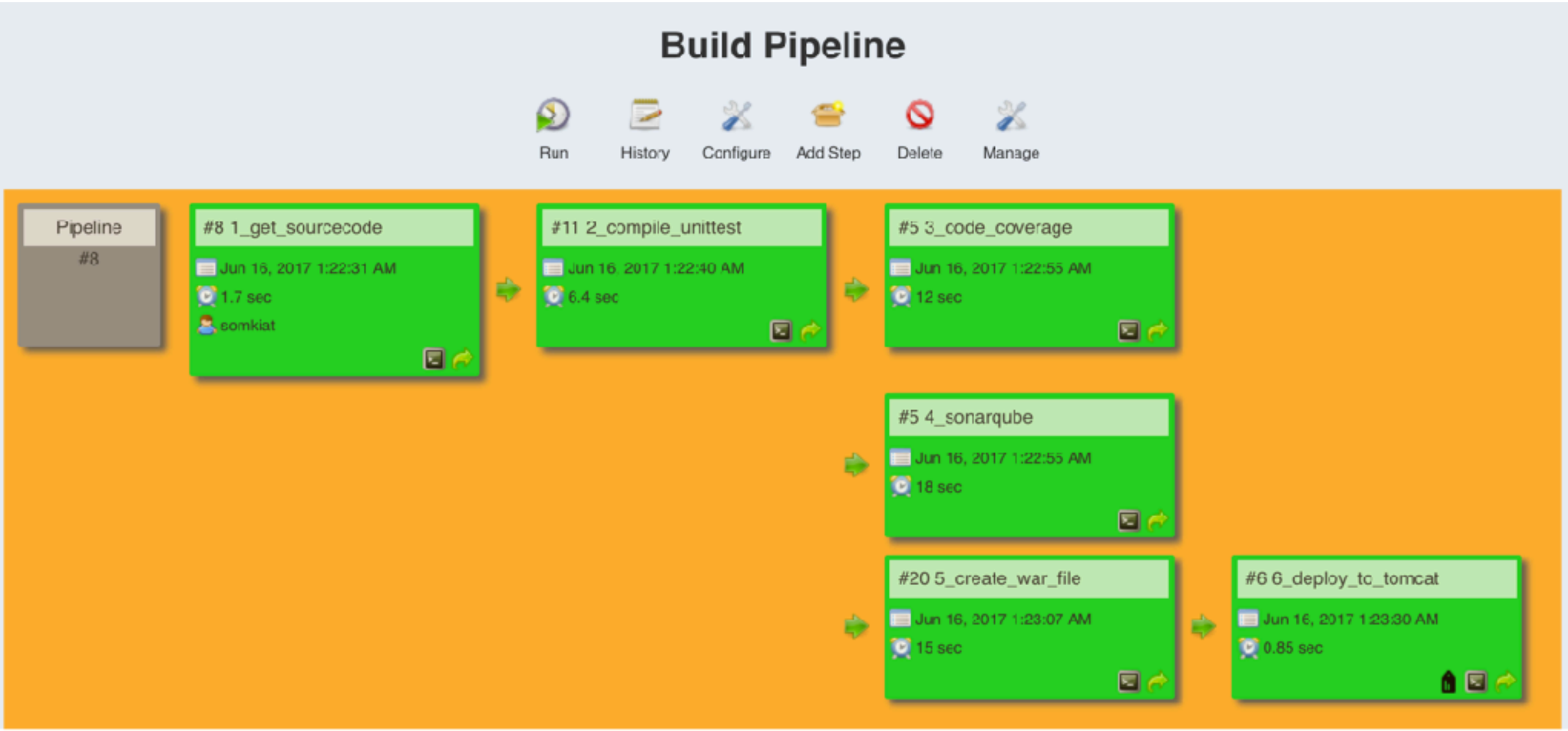
Tomcat Web Application Manager

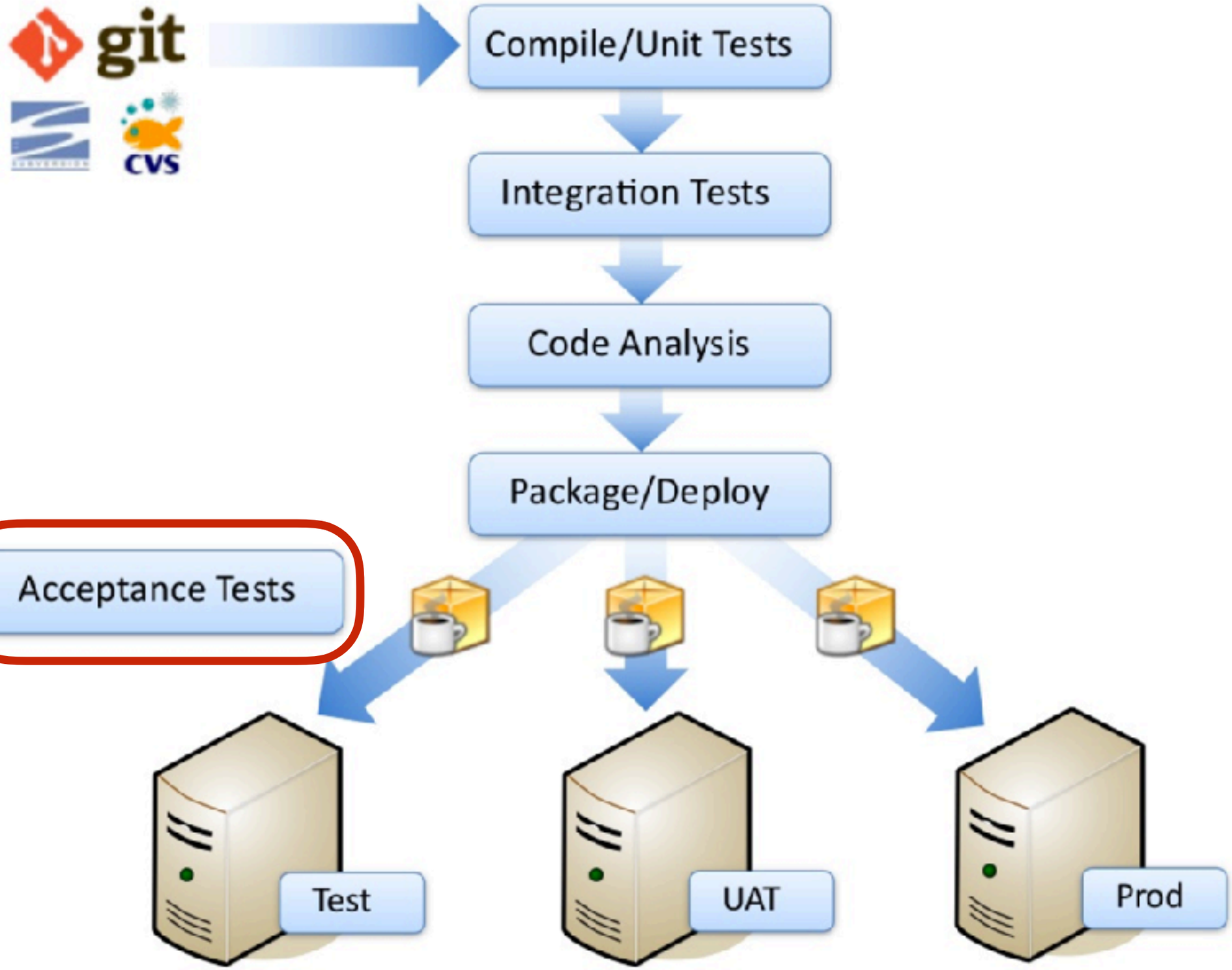
Message:	OK

Manager			
List Applications	HTML Manager Help	Manager Help	Server Status

Applications					
Path	Version	Display Name	Running	Sessions	Commands
/	*None specified*	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/demo	*None specified*	Grade Converter	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	*None specified*	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

My build pipeline

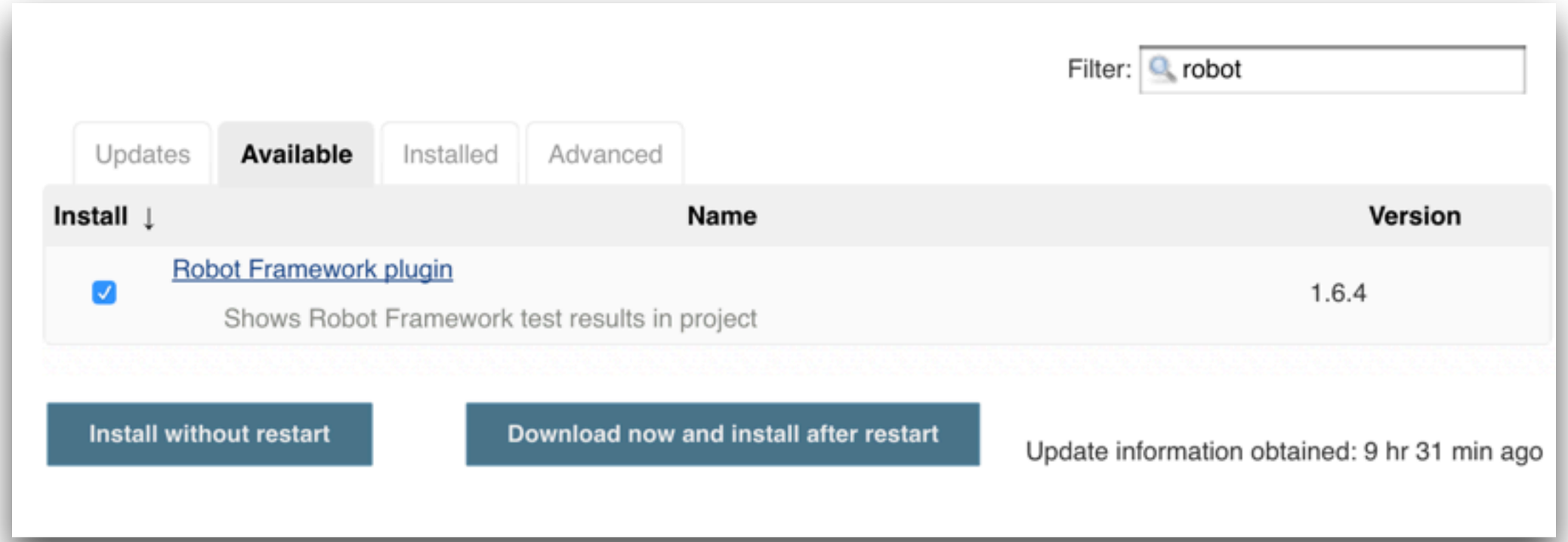



git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
Acceptance Tests
Test
UAT
Prod

6

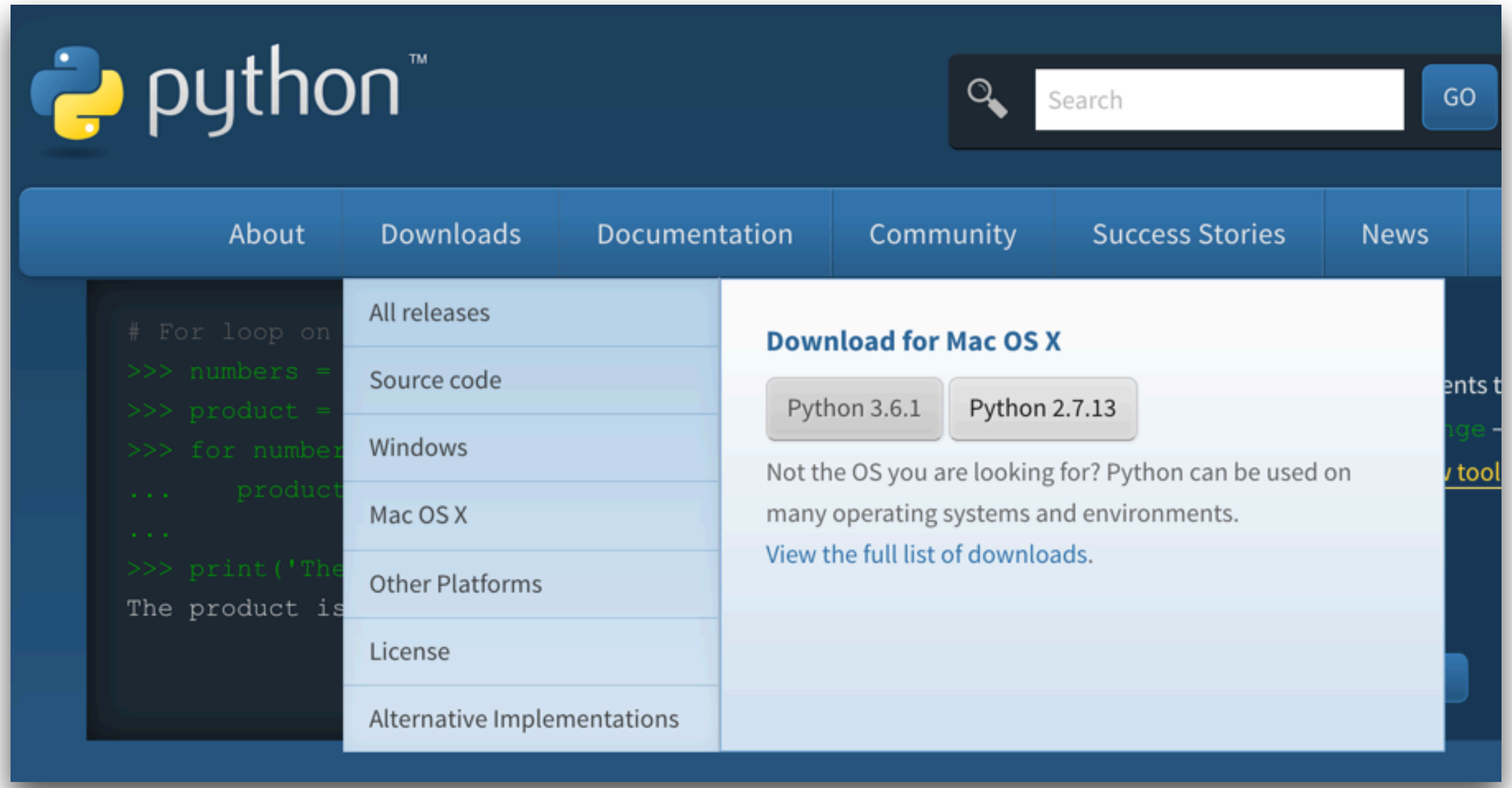
Acceptance test with Robotframework

Install Robot Framework plugin



Install Python 2.7.13

https://www.python.org/



Try ...

$python

$pip

Configuration for Windows

$set **PYTHON_HOME**=c:/Python27

$set **PATH**=.;%PYTHON_HOME%;

%PYTHON_HOME%/Scripts;%PATH%

Configuration for Mac

$easy_install pip

Try again !!

$python

$pip

Install Robot framework

$pip install **robotframework**

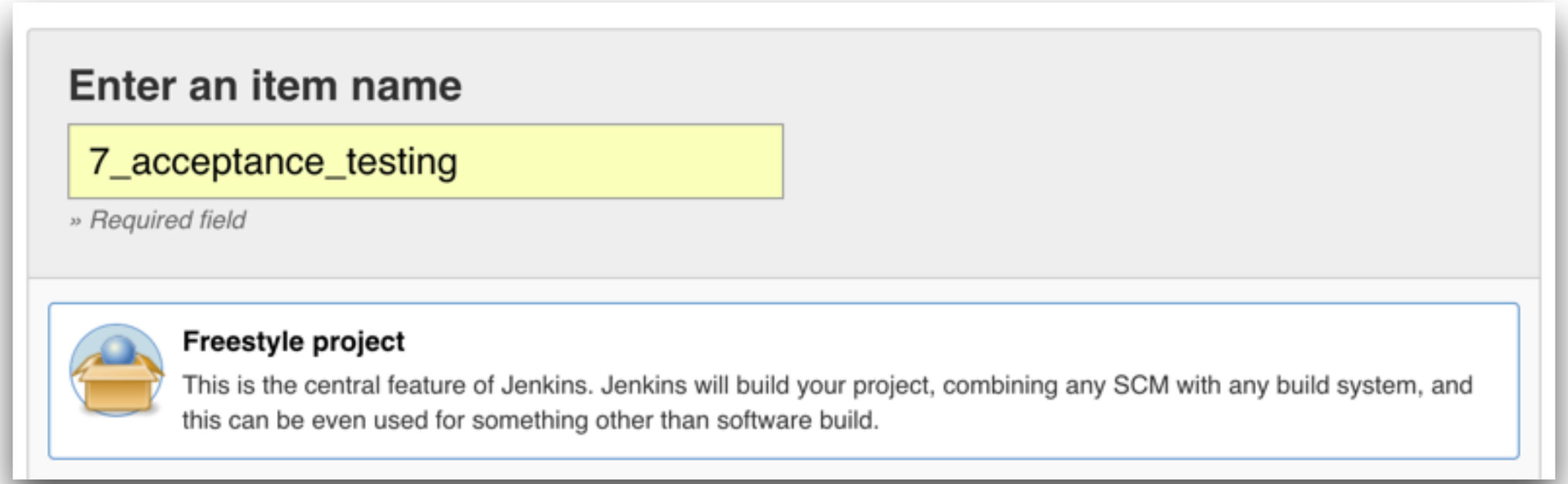
$pip install **robotframework-selenium2library**

Try to run

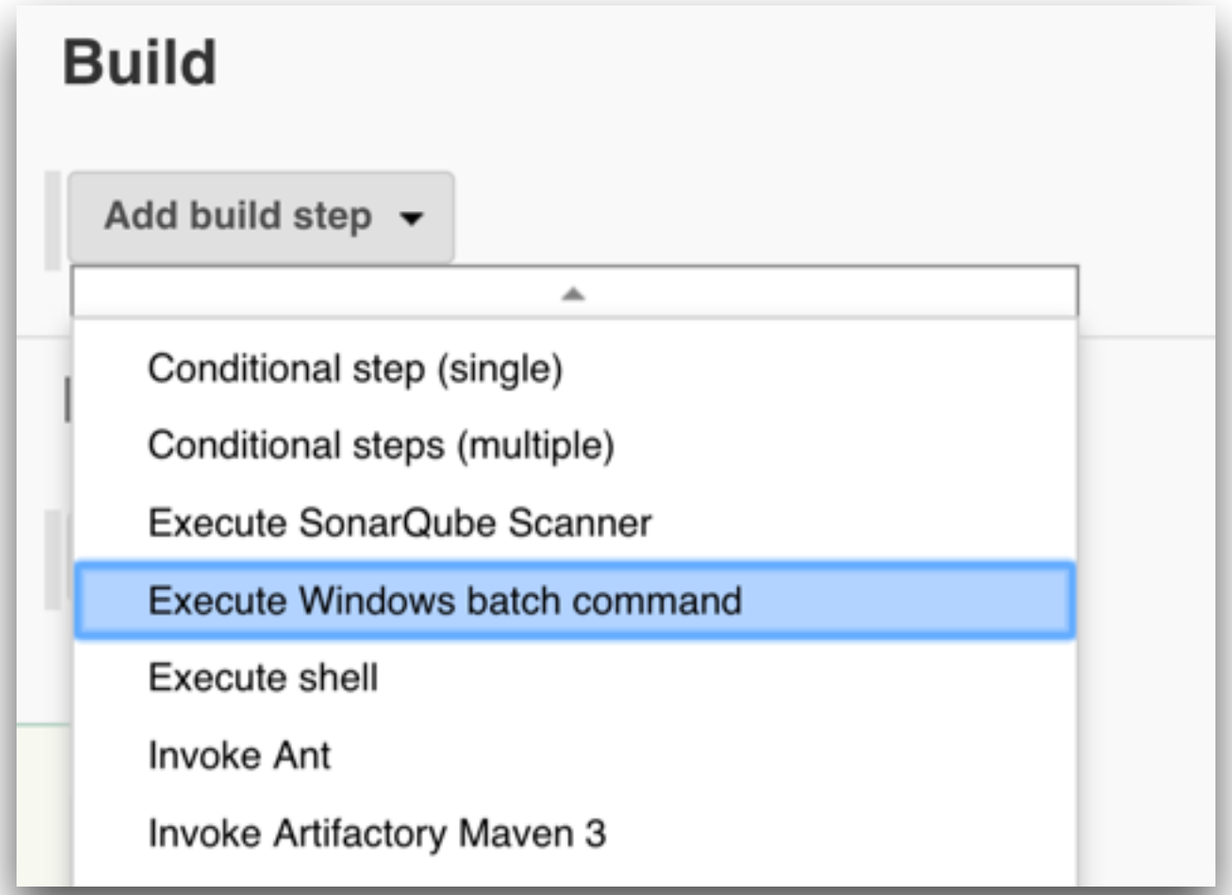
$pybot

```
[ ERROR ] Expected at least 1 argument, got 0.

Try --help for usage information.
```

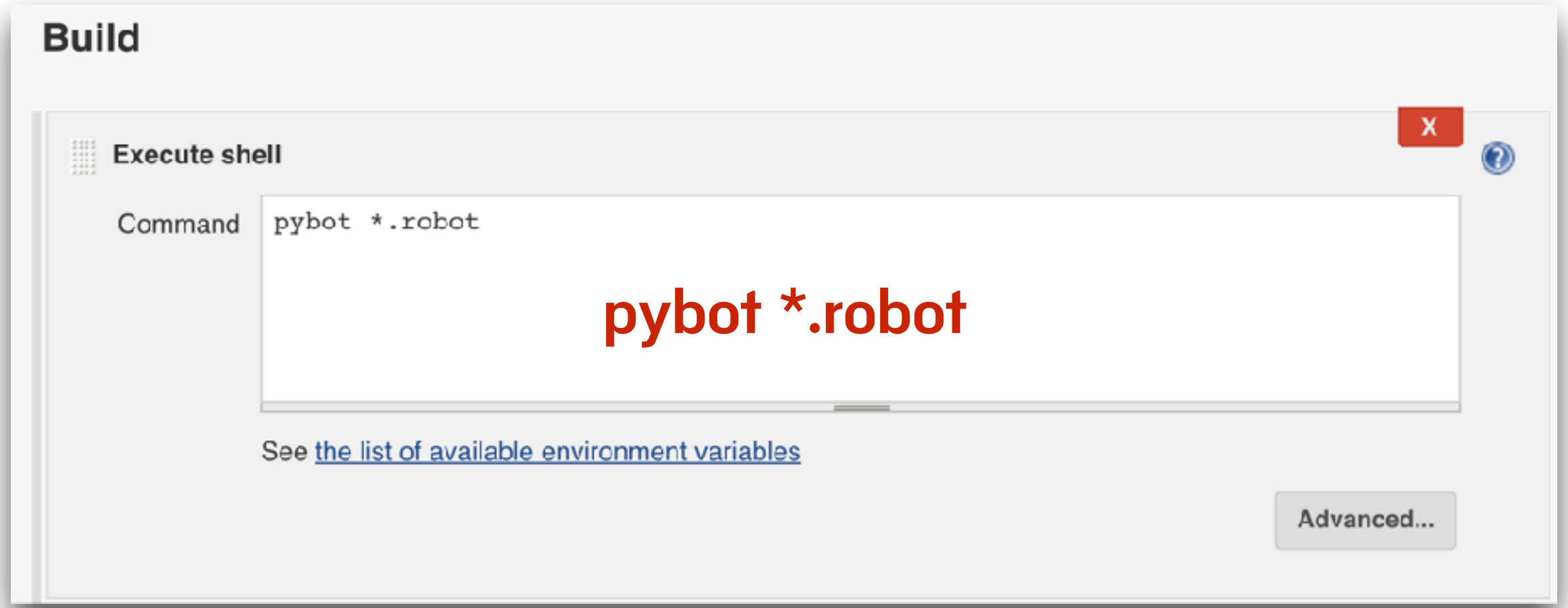
Crete new job on Jenkins



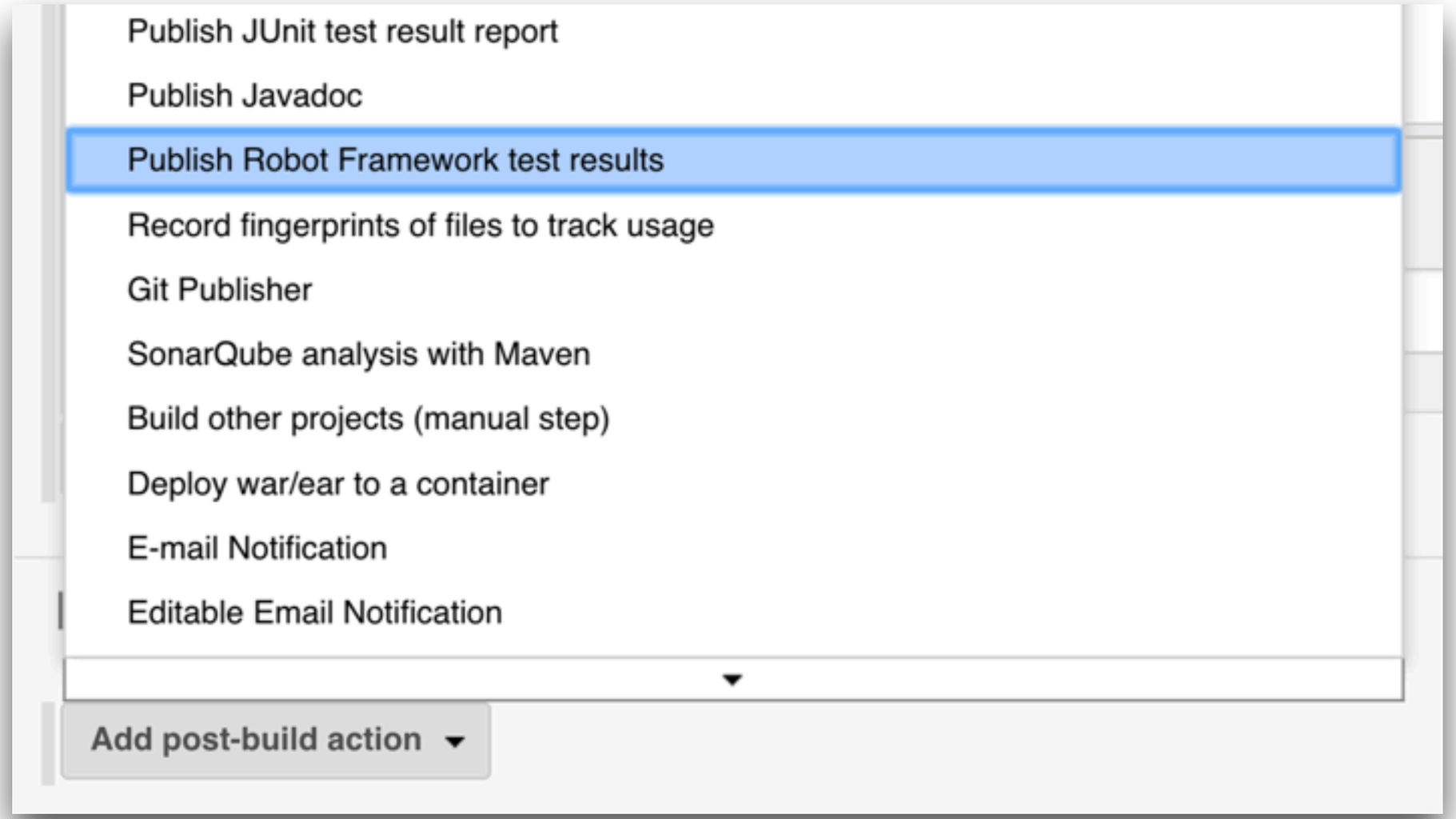
Config job to run robot



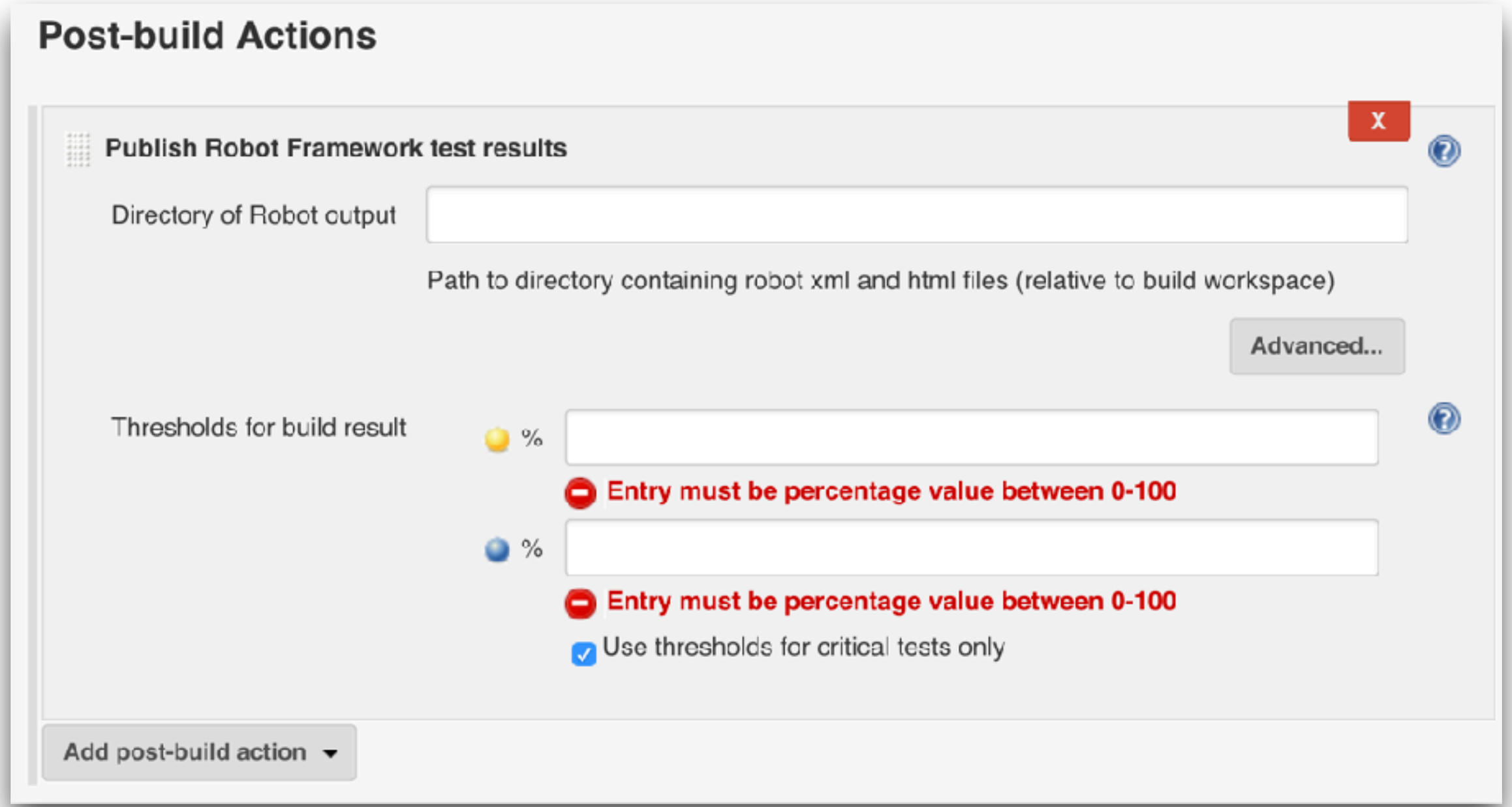
Config job to run robot



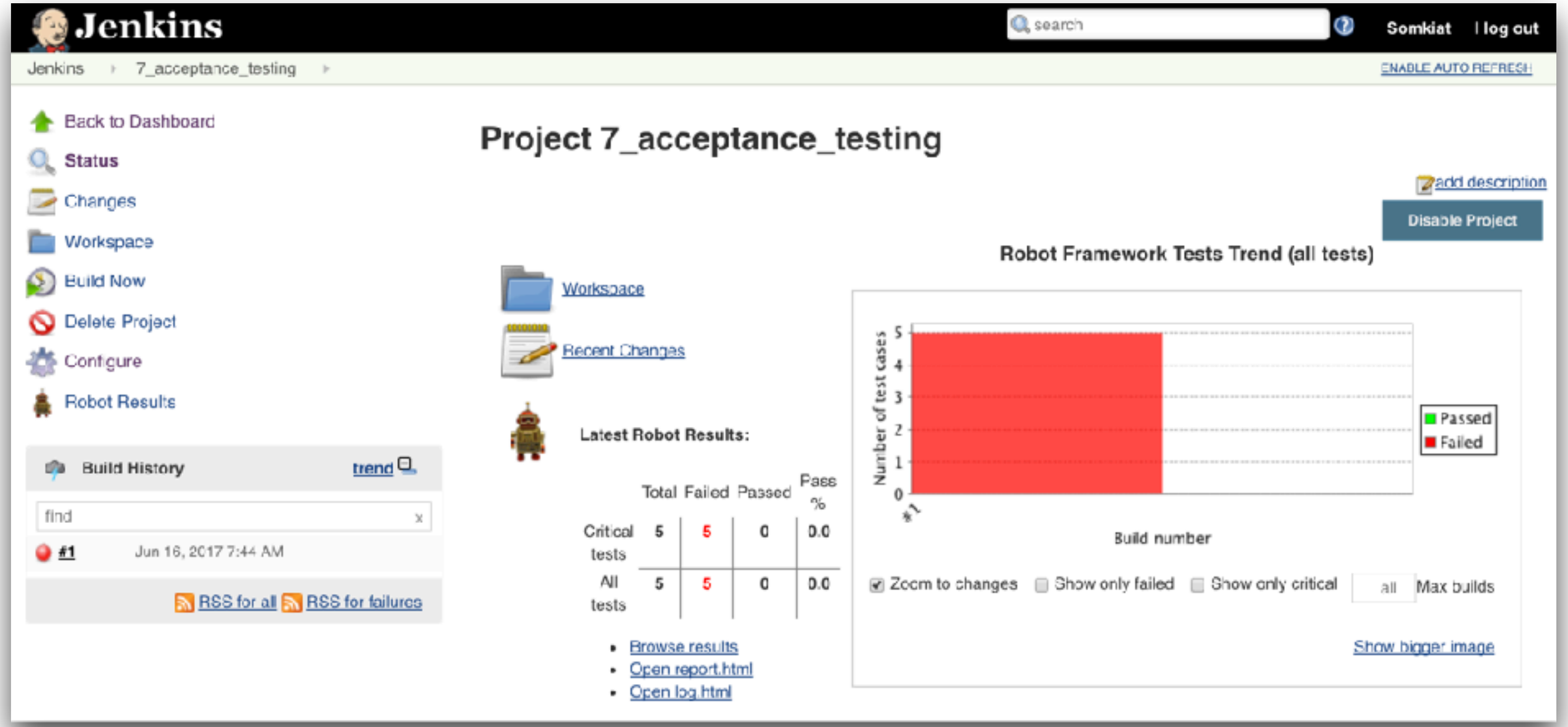
Add report in Post-build



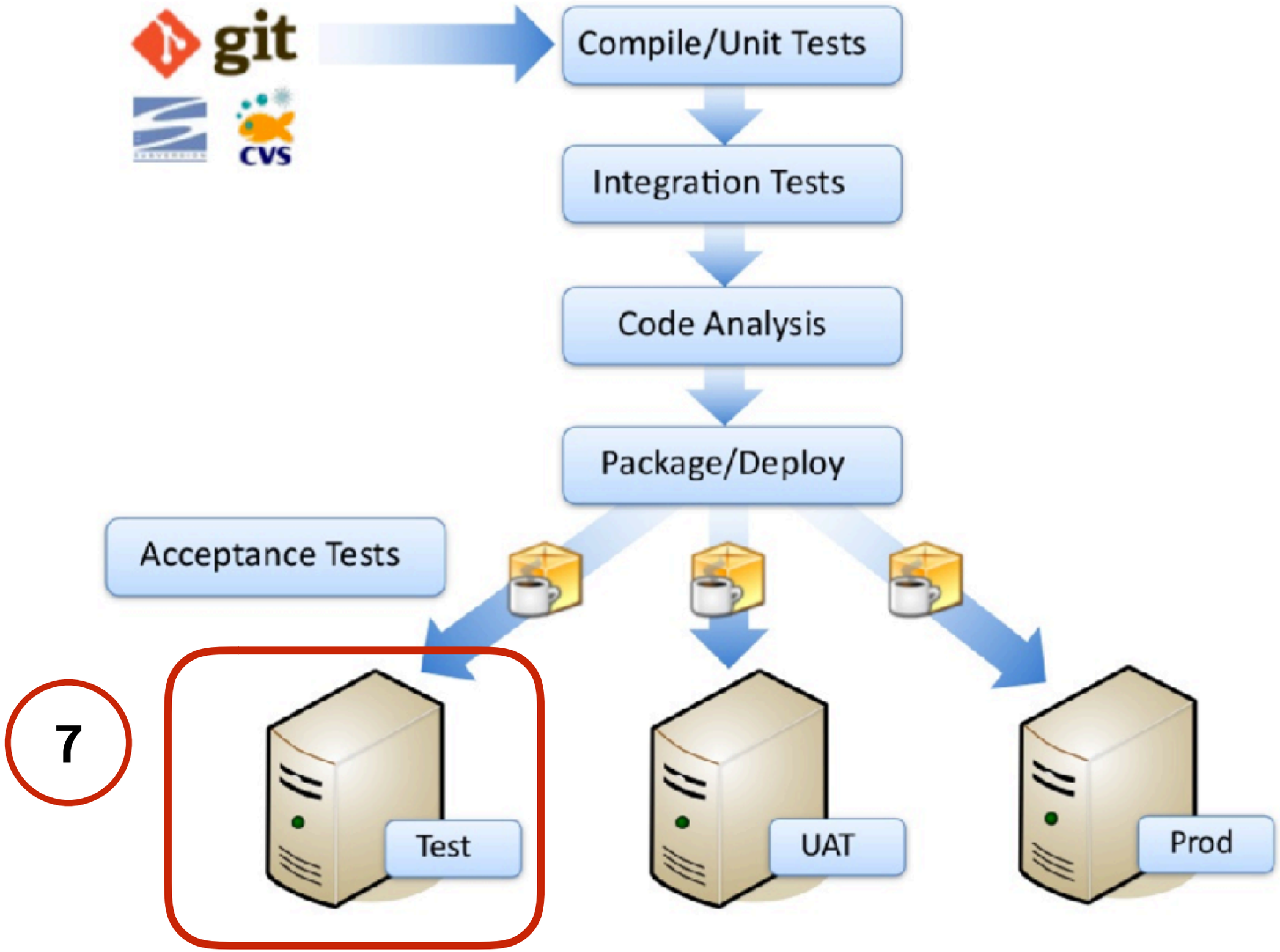
Add report in Post-build



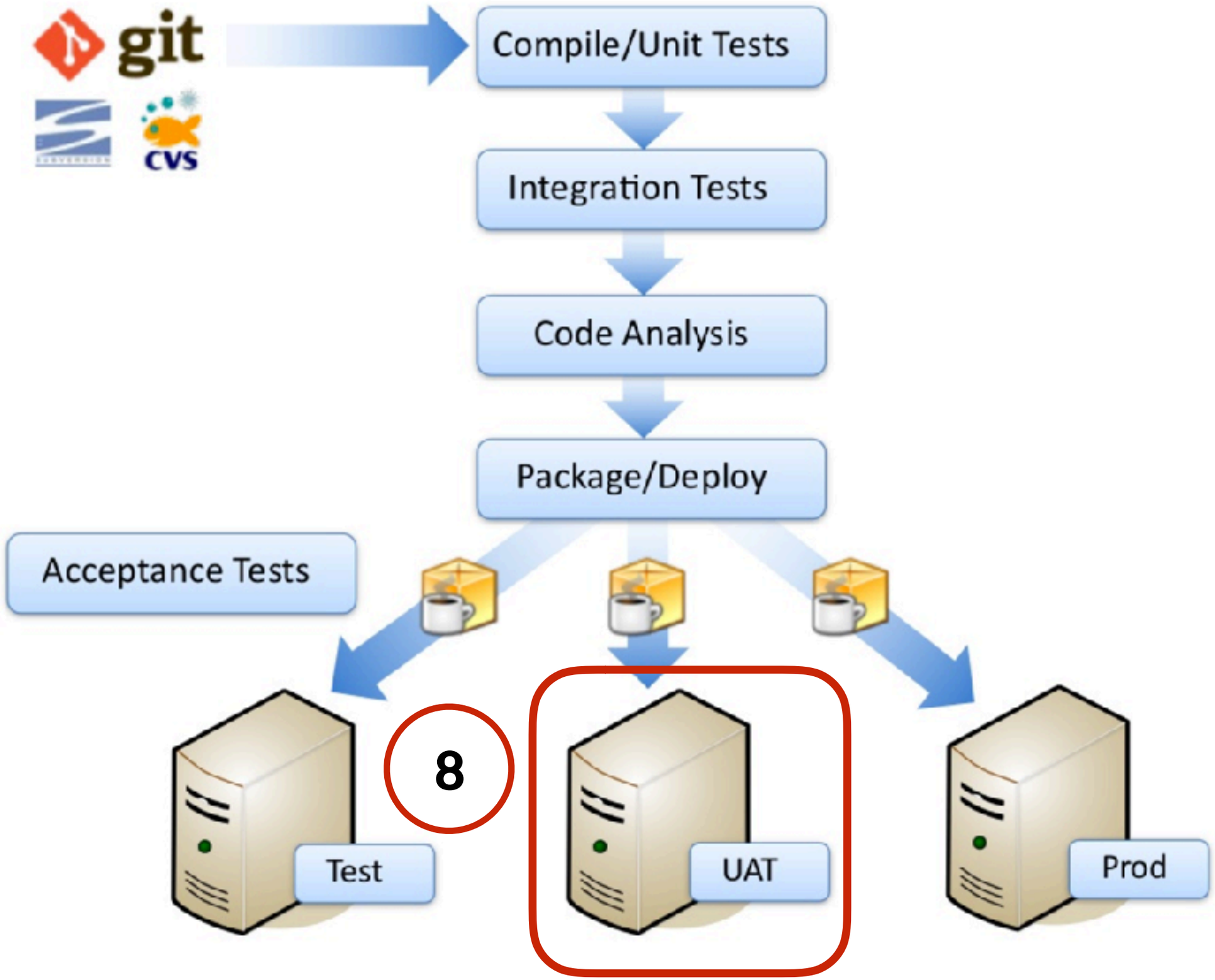
Try to build !!


Jenkins
search
Somkiat
| log out
Jenkins
7_acceptance_testing
ENABLE AUTO REFRESH
Back to Dashboard
Status
Changes
Workspace
Build Now
Delete Project
Configure
Robot Results
Build History
trend
find
#1
Jun 16, 2017 7:44 AM
RSS for all
RSS for failures
Project 7_acceptance_testing
add description
Disable Project
Workspace
Recent Changes
Latest Robot Results:
Total
Failed
Passed
Pass %
Critical tests
5
5
0
0.0
All tests
5
5
0
0.0
Browse results
Open report.html
Open log.html
Robot Framework Tests Trend (all tests)
Number of test cases
Build number
Passed
Failed
Zoom to changes
Show only failed
Show only critical
all
Max builds
Show bigger image

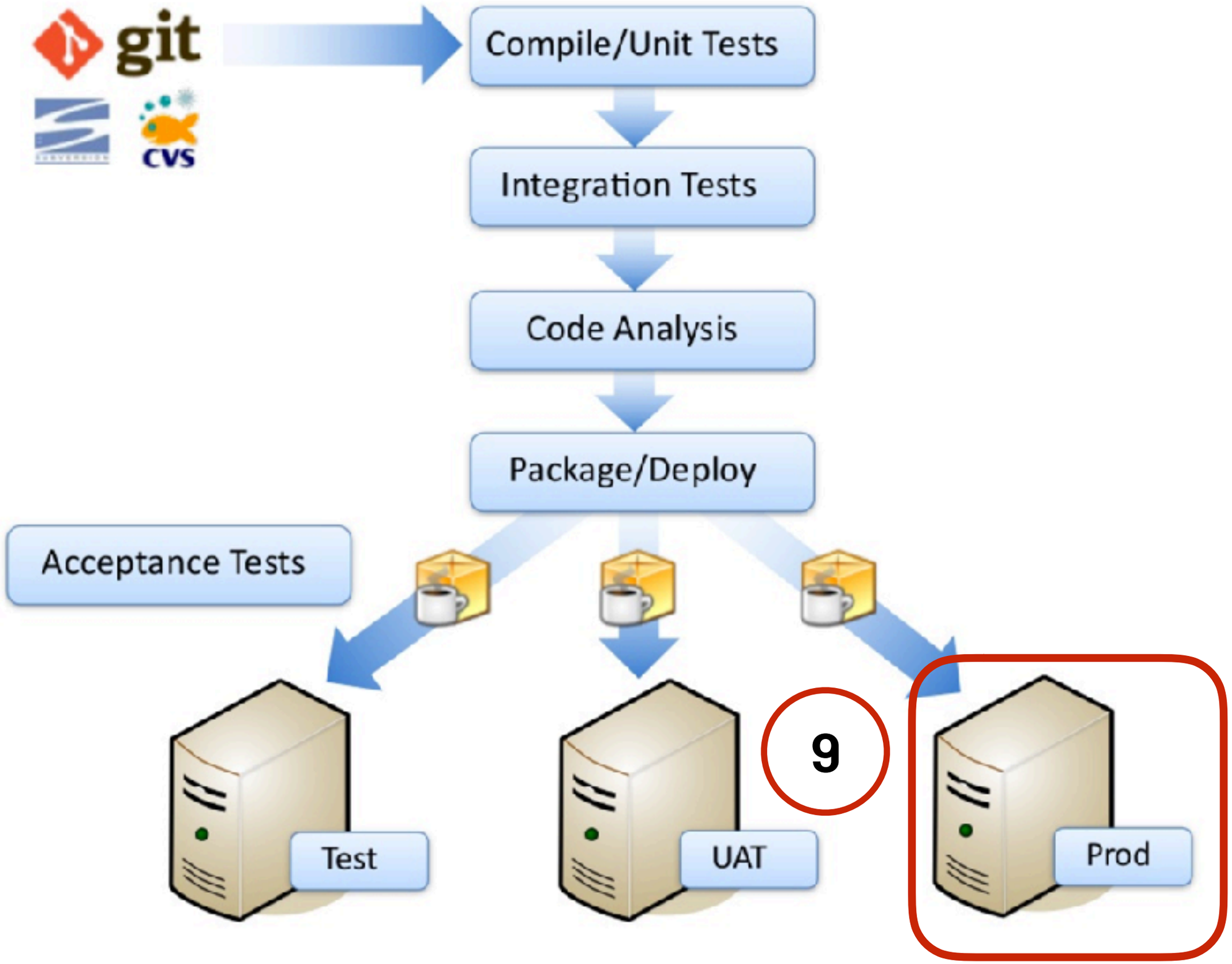
My build pipeline


git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
Acceptance Tests
Test
UAT
Prod

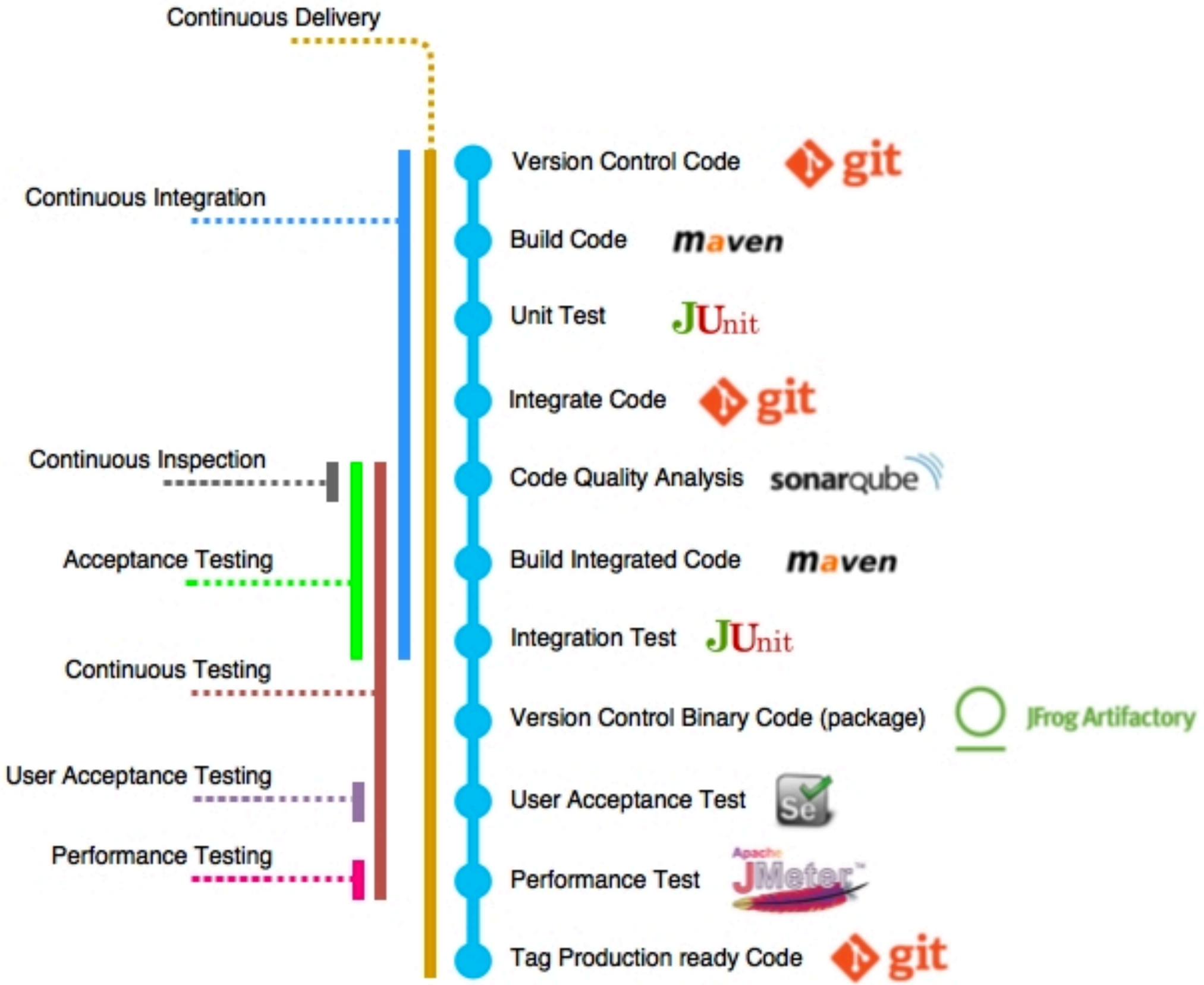
Deploy to Test ?


git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
Acceptance Tests
8
Test
UAT
Prod

Deploy to UAT ?


git
CVS
Compile/Unit Tests
Integration Tests
Code Analysis
Package/Deploy
Acceptance Tests
Test
UAT
Prod
9

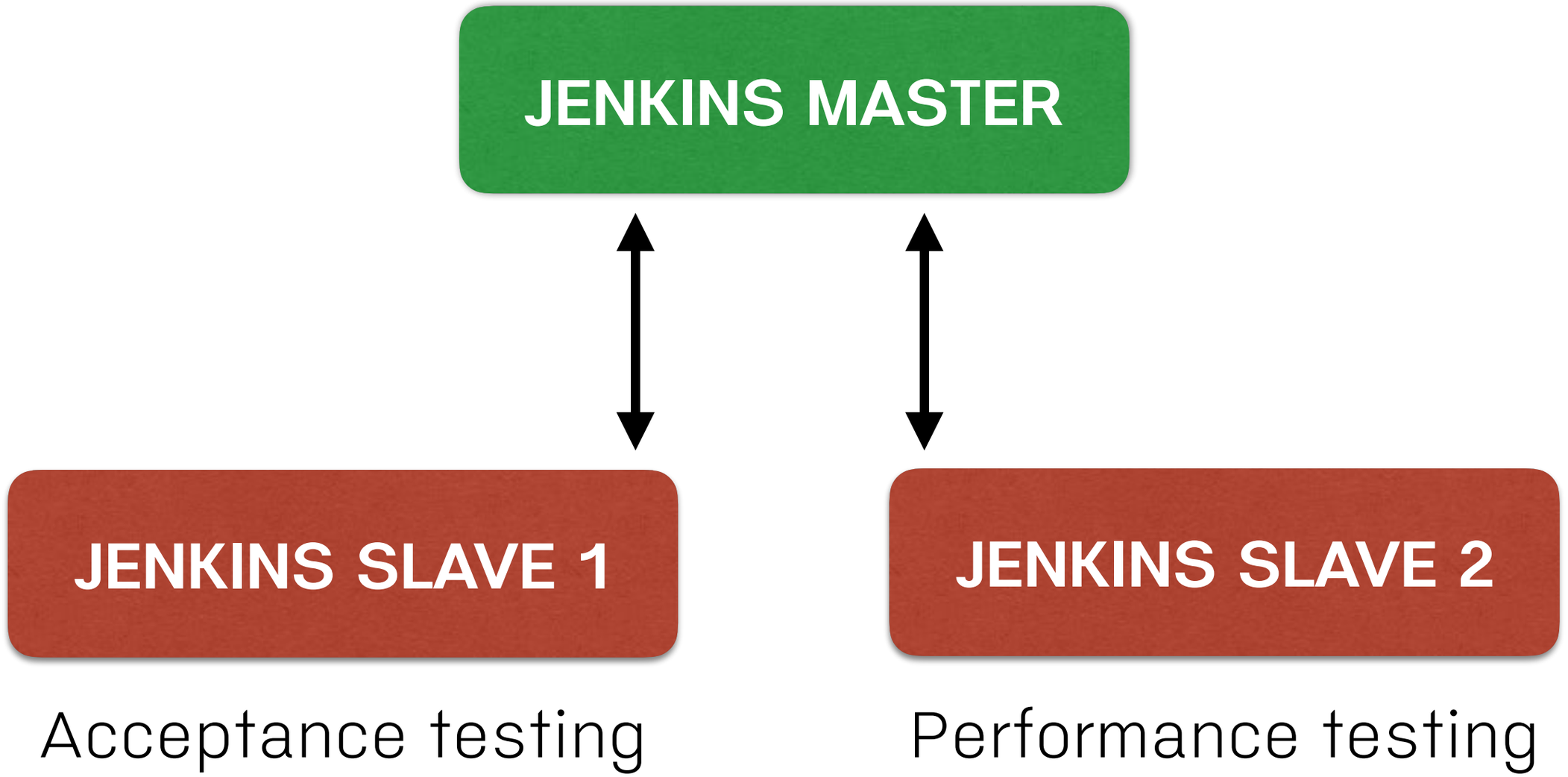
Deploy to Production ?


Continuous Delivery
Continuous Integration
Continuous Inspection
Acceptance Testing
Continuous Testing
User Acceptance Testing
Performance Testing
Version Control Code
git
Build Code
maven
Unit Test
JUnit
Integrate Code
git
Code Quality Analysis
sonarqube
Build Integrated Code
maven
Integration Test
JUnit
Version Control Binary Code (package)
JFrog Artifactory
User Acceptance Test
Performance Test
Apache JMeter
Tag Production ready Code
git

What’s next ?

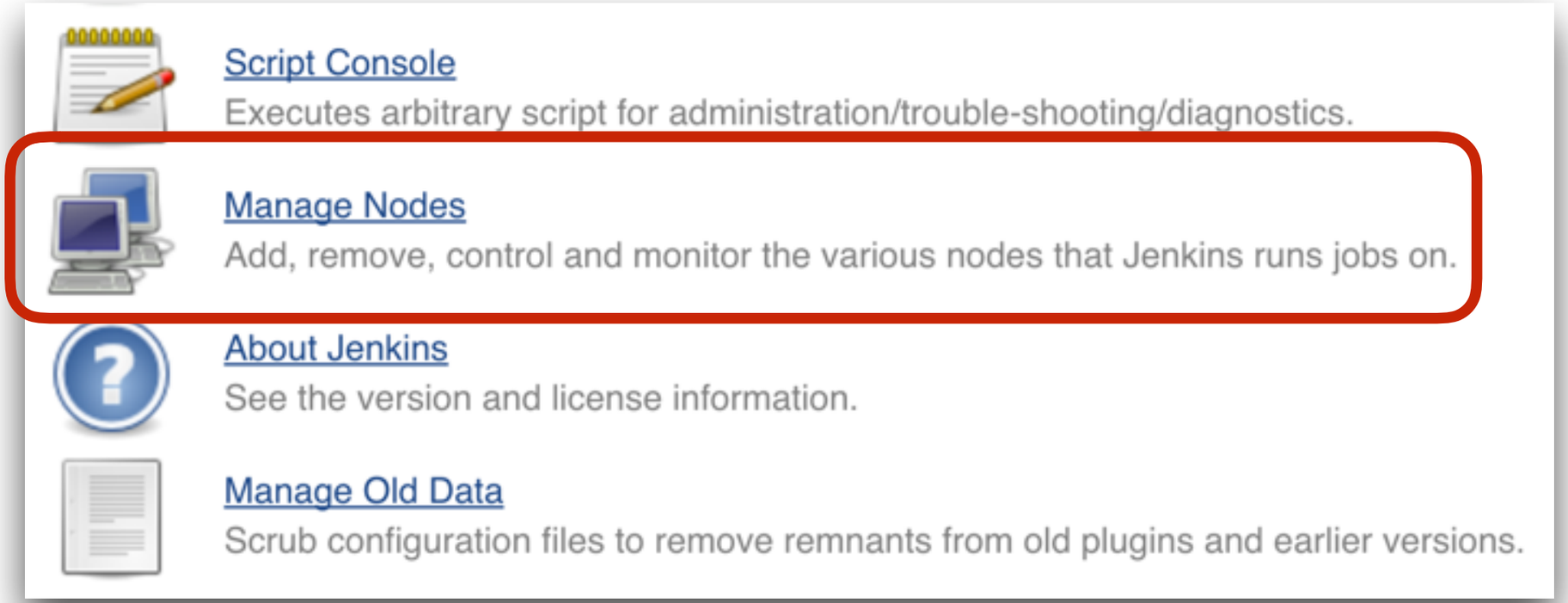
Performance test with JMeter

Jenkins master-slave



Add slave node

Manage Jenkins -> Manage Nodes

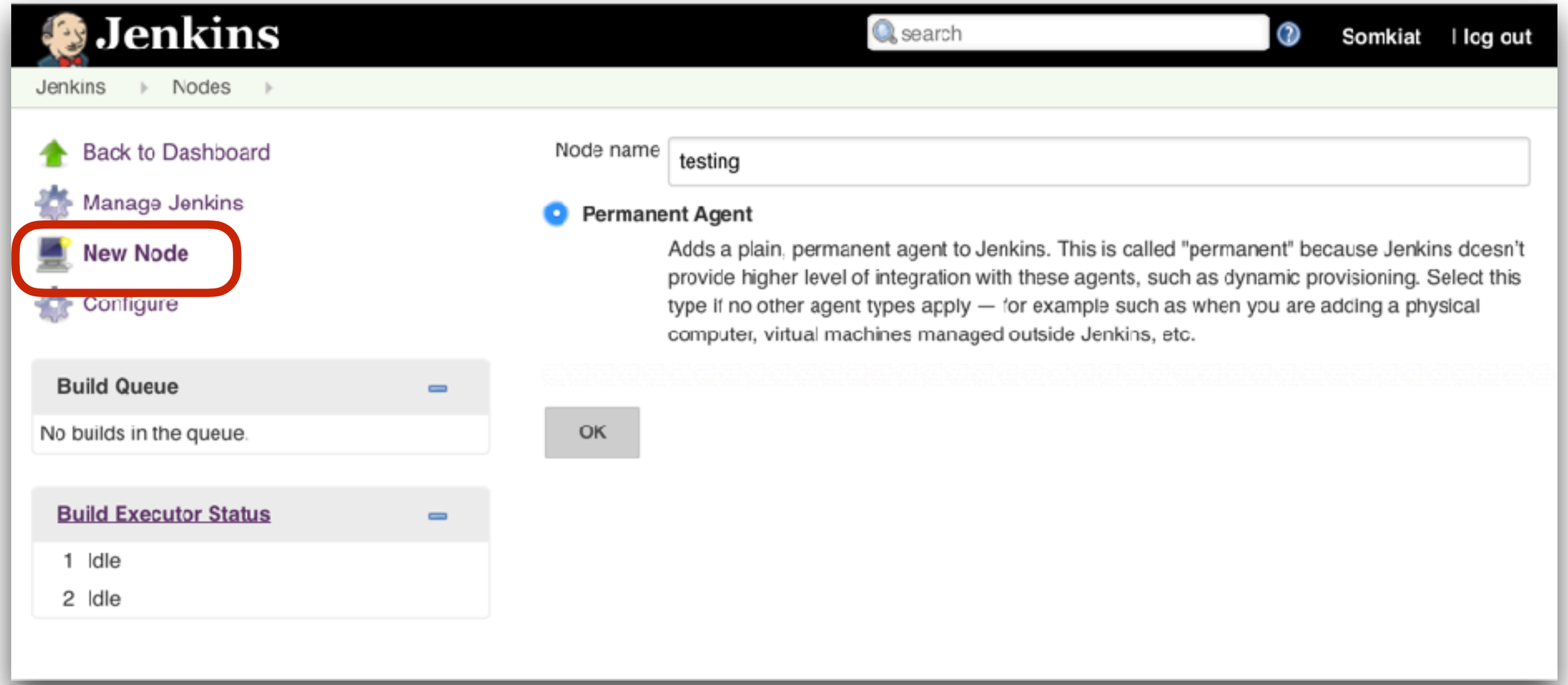


Current node

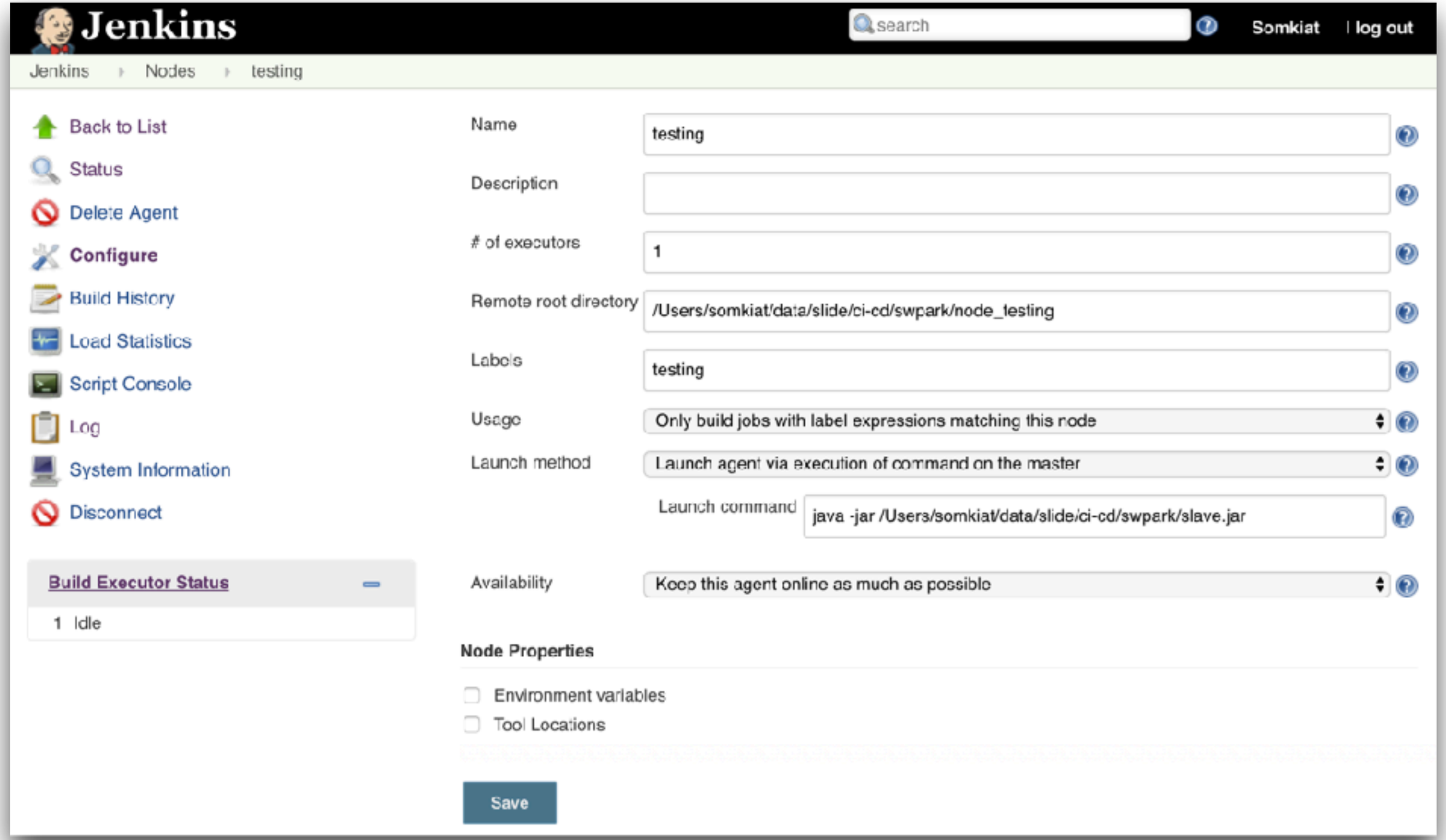
S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Mac OS X (x86_64)	In sync	21.31 GB	1.26 GB	21.31 GB	0ms
	Data obtained	**22 min**	**22 min**	**22 min**	**22 min**	**22 min**	**22 min**

Refresh status

New node


Jenkins
search
Somkiat
log out
Jenkins
Nodes
Back to Dashboard
Manage Jenkins
New Node
Configure
Node name
testing
Permanent Agent
Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins dcesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.
Build Queue
No builds in the queue.
Build Executor Status
1 Idle
2 Idle
OK

Edit node


Jenkins
search
Somkiat
log out
Jenkins
Nodes
testing
Back to List
Status
Delete Agent
Configure
Build History
Load Statistics
Script Console
Log
System Information
Disconnect
Build Executor Status
1 Idle
Name
testing
Description
# of executors
1
Remote root directory
/Users/somkiat/data/slide/ci-cd/swpark/node_testing
Labels
testing
Usage
Only build jobs with label expressions matching this node
Launch method
Launch agent via execution of command on the master
Launch command
java -jar /Users/somkiat/data/slide/ci-cd/swpark/slave.jar
Availability
Keep this agent online as much as possible
Node Properties
Environment variables
Tool Locations
Save

List of nodes

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time	
	master	Mac OS X (x86_64)	In sync	21.28 GB	1.02 GB	21.28 GB	0ms	
	testing	Mac OS X (x86_64)	In sync	21.28 GB	1.02 GB	21.28 GB	4025ms	
	Data obtained	**1 min 27 sec**	**1 min 27 sec**	**1 min 27 sec**	**1 min 26 sec**	**1 min 27 sec**	**1 min 27 sec**	

Refresh status