

# How to Integrate JMeter into Jenkins

Hannah Inman - 19 Sep 2016

Tweet Like 1 Share 46 G+

This is a guest post by Dmitri Tikhansi from BlazeMeter.

Continuous integration (CI) ([http://info.blazemeter.com/how-to-build-testing-into-continuous-integration-pipeline-webinar?utm\\_source=BM&utm\\_medium=resources&utm\\_campaign=webinar-how-to-build-testing-into-your-ci-pipeline?utm\\_source=cloudbees&utm\\_medium=BM\\_blog&utm\\_campaign=integrate-jmeter-jenkins&utm\\_campaign=integrate-jmeter-jenkins](http://info.blazemeter.com/how-to-build-testing-into-continuous-integration-pipeline-webinar?utm_source=BM&utm_medium=resources&utm_campaign=webinar-how-to-build-testing-into-your-ci-pipeline?utm_source=cloudbees&utm_medium=BM_blog&utm_campaign=integrate-jmeter-jenkins&utm_campaign=integrate-jmeter-jenkins)), test automation and “shifting left” are becoming the standard for DevOps, developers and QA engineers. But despite the importance of performance, and the understanding that systems are complex and it can be challenging to identify and fix bottlenecks in a short time, load testing is still not an integral part of the CI workflow.

Jenkins easily enables users to integrate load testing into its workflow. By using Jenkins as part of the CI process and to trigger jobs by commits, users are taking advantage of automation and process speed-up capabilities.

## Advantages of Integrating JMeter into Jenkins

JMeter (<http://jmeter.apache.org/>) is one of the most popular open-source load testing systems. By integrating JMeter into Jenkins, users can enjoy:

- Unattended test executions right after software build and deploy
- Automatic build failures in case of performance degradation
- Easy access to test reports that show application metric trends - all tests are in one place and available to anybody with the correct permissions
- Automated routine work of test configuration, execution and baseline results analysis. Users' hands and minds are free for more important, complex and interesting tasks.

## How to Integrate JMeter into Jenkins

1. Store JMeter results as XML (recommended because they are easier to use) or CSV
2. Specify the command to run your test in the Execute Shell Batch Command section
3. Check the Console Output tab to verify the execution was successful
4. Find your files in the project's workspace
5. Specify Build Parameters

## How to Use the Performance Plugin

To view JMeter reports on Jenkins, you can use the Performance plugin. (<https://wiki.jenkins-ci.org/display/JENKINS/Performance+Plugin>)

1. Install the Performance plugin (<https://wiki.jenkins-ci.org/display/JENKINS/Performance+Plugin>)
2. Configure the plugin

The Performance plugin can be added as a “Post-build Action”. When the JMeter test is finished, the plugin will:

- Collect the data
- Conditionally fail the build if the error threshold is exceeded
- Build or update the performance trend chart for the project

The screenshot shows the 'Publish Performance test result report' configuration page in Jenkins. The page is divided into several sections:

- Performance report:** Includes an 'Add a new report' button and a 'Select mode:' section with radio buttons for 'Relative Threshold' (selected) and 'Error Threshold'.
- Build result:** Includes a checkbox for 'Fail build when result files are not present'.
- Use Error thresholds on single build:** Includes input fields for 'Unstable' and 'Failed' thresholds, both set to '0'.
- Average response time threshold:** Includes a large text area for setting the threshold.
- Use Relative thresholds for build comparison:** Includes input fields for 'Unstable % Range' and 'Failed % Range', both set to '0.0'. It also has radio buttons for 'Compare with previous Build number' (selected) and 'Compare with Build number'.
- Compare based on:** Includes a dropdown menu set to 'Average Response Time'.
- Performance display:** Includes checkboxes for 'Performance Per Test Case Mode' (selected) and 'Show Throughput Chart'.

Configuration options explained:

<https://www.cloudbees.com/blog/how-integrate-jmeter-jenkins>

- **Performance report** - For **JMeter** you will need to upload a file in XML format.
- **Select mode** - The choices are in the **Relative Threshold** and **Error Threshold**. The **Relative Threshold** compares the difference from the previous test, and if it exceeds the defined value the build will be marked as failed/unstable. The **Error Threshold** marks the the build as unstable or failed if the amount of errors will exceed the specified value.
- **Build result** - If the JMeter test doesn't generate the output jtl file(s) - the build will be failed.
- **Use error thresholds on single build** - Define error thresholds for the current build.
- **Average response time threshold** - Set the maximum acceptable value of the Average Response Time metric.
- **Use relative thresholds for build comparison** - Set the percentage difference of errors. The "source" build can be either the previous build or "known good" build which is used as a baseline.
- **Performance per Test Case Mode** - If you need to see separate graphs for each single test case on the Performance Trend chart you can trigger the behavior by this option.
- **Show Throughput Chart** - Set whether to display "Throughput" trend chart on project dashboard or not.

Jenkins can also be integrated with BlazeMeter ([https://a.blazemeter.com/app/sign-up?utm\\_source=cloudbees&utm\\_medium=BM\\_blog&utm\\_campaign=integrate-jmeter-jenkins&utm\\_campaign=integrate-jmeter-jenkins](https://a.blazemeter.com/app/sign-up?utm_source=cloudbees&utm_medium=BM_blog&utm_campaign=integrate-jmeter-jenkins&utm_campaign=integrate-jmeter-jenkins)) and Taurus ([http://gettaurus.org/?utm\\_source=cloudbees&utm\\_medium=BM\\_blog&utm\\_campaign=integrate-jmeter-jenkins&utm\\_campaign=integrate-jmeter-jenkins](http://gettaurus.org/?utm_source=cloudbees&utm_medium=BM_blog&utm_campaign=integrate-jmeter-jenkins&utm_campaign=integrate-jmeter-jenkins)), for faster and easier results and more analysis options.

Congratulations on adding performance tests to your continuous integration process!

#### Blog Categories:

Blog Posts about Jenkins (/blog-categories/jenkins)

## Add new comment

Your name

E-mail

Homepage

Subject

Comment \*

By submitting this form, you accept the Mollom privacy policy (<https://www.mollom.com/web-service-privacy-policy>).

✓ SAVE

## Related Content

### Women in DevOps: Nicole Munro (/blog/women-devops-nicole-munro)

Hannah Inman - 26 Jan 2018

Tweet

Like 0

Share

25

G+

Read more (/blog/women-devops-nicole-munro) Hannah Inman's blog (/blogs/hannah-inman)

## Now on DevOps Radio: What's In Your DevOps Wallet? Featuring Capital One's Brock Beatty (/blog/now-devops-radio-what%E2%80%99s-your-devops-wallet-featuring-capital-one%E2%80%99s-brock-beatty)

Hannah Inman - 30 Jan 2018

Tweet Like 0 Share G+

Read more (/blog/now-devops-radio-what%E2%80%99s-your-devops-wallet-featuring-capital-one%E2%80%99s-brock-beatty)  
Hannah Inman's blog (/blogs/hannah-inman)

## Now on DevOps Radio: Open Source, OpenShift, Open Sesame, Featuring Red Hat's Ari LiVigni (/blog/now-devops-radio-open-source-openshift-open-sesame-featuring-red-hat%E2%80%99s-ari-livigni)

Hannah Inman - 09 Jan 2018

Tweet Like 0 Share G+

Read more (/blog/now-devops-radio-open-source-openshift-open-sesame-featuring-red-hat%E2%80%99s-ari-livigni)  
Hannah Inman's blog (/blogs/hannah-inman)

### Subscribe to our Blog Posts

Enter email

SUBSCRIBE

(https://chrome.google.com/webstore/detail/rss-subscription-extensio/nlbjncdgjeocebhnmkbbdbekmmmcbfjd)

Tweet Like 1 Share 46 G+

### Recent Blogs