

Triggering Analysis on Hudson Job

Table of Contents

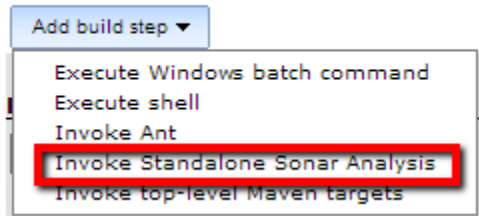
- [Triggering a SonarQube Analysis with the SonarQube Scanner](#)
- [Triggering a SonarQube Analysis with Maven](#)

For each job, configure an analysis using either the Build Step or the Post Build Action. You do not need both.

Triggering a SonarQube Analysis with the SonarQube Scanner

Go to the *Build* section, click on *Add build step* and choose *Invoke Standalone Sonar Analysis*.

Build



Configure the SonarQube analysis. You can either point to an existing *sonar-project.properties* file or set the analysis properties directly in the *Project properties* field:

Build

Invoke Standalone Sonar Analysis

Sonar Installation:

JDK:

Sonar Runner:

Path to project properties:

Project properties:

JVM Options:

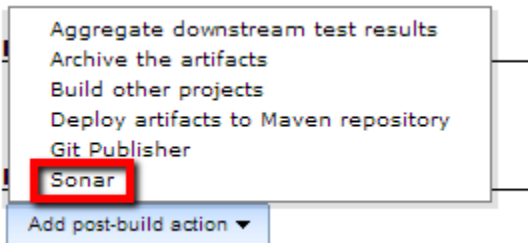
Parameters:

Properties	Description
Sonar Installation	Mandatory. Which SonarQube installation to use. To manage SonarQube installations, see Configuring the SonarQube Scanner for Hudson .
JDK	Optional. The JDK to use for the SonarQube analysis (by default, it is inherited from the job).
Sonar Runner	Mandatory. Which SonarQube Runner to use to launch the analysis. To manage SonarQube Runners, see Configuring Hudson SonarQube Plugin .
Path to project properties	Optional. Relative path of the project properties file.

Project properties	<p>Mandatory if 'Path to project properties' is not set. The properties of the SonarQube analysis. The properties defined here will override the ones defined in the project properties file.</p> <pre> # required metadata sonar.projectKey=my:project sonar.projectName=My project sonar.projectVersion=1.0 # path to source directories (required) sonar.sources=srcDir1,srcDir2 # path to test source directories (optional) sonar.tests=testDir1,testDir2 # path to project binaries (optional), for example directory of Java bytecode sonar.binaries=binDir # optional comma-separated list of paths to libraries. Only path to JAR file and path to directory of classes are supported. sonar.libraries=path/to/library.jar,path/to/classes/dir # The value of the property must be the key of the language. sonar.language=cobol # Additional parameters sonar.my.property=value </pre> <p>See Analyzing with SonarQube Runner for detailed information on configuring a SonarQube analysis with the SonarQube Runner.</p>
JVM_OPTIONS	The JVM parameters used when executing the SonarQube Runner.

Triggering a SonarQube Analysis with Maven

On a Maven job, go to the 'Post-build Actions' section and click on 'Add post-build action':



Then configure this post-build action:

- Choose the *Sonar Installation*.

Post-build Actions



- Fill in the 'Advanced...' parameters as needed:

Post-build Actions

Sonar

Sonar Installation ?

Branch ?

Language ?

JDK ?

MAVEN_OPTS ?

Additional properties ?

MAVEN_OPTS env var to provide, if not set the plugin will use the MAVEN_OPTS defined by the maven builder config.

Additional properties to be passed to the mvn executable (example: -Dsome.property=some.value).

Don't use global triggers configuration ?

Delete

Parameters:

Properties	Description
Sonar Installation	Mandatory. SonarQube installation to use. To manage SonarQube installations, see Configuring Hudson SonarQube Plugin .
Branch	Optional. sonar.branch property. See Analysis Parameters for more information.
Language	Mandatory. sonar.language property.
MAVEN_OPTS	Optional. To specify JVM options needed when launching Maven as an external process. See MAVEN_OPTS documentation . Shell-like environment variable expansions work in this field, by using the \${VARIABLE} syntax .
JDK	Optional. The JDK to use for the SonarQube analysis (by default, it is inherited from the job).
Additional properties	Optional. Additional properties can be set. See Analysis Parameters for detailed documentation on these additional properties.
Don't use global triggers configuration	If checked, overrides the default trigger actions set at SonarQube installation level.