Parameterized Trigger Plugin

This plugin lets you trigger new builds when your build has completed, with various ways of specifying parameters for the new build. You can add multiple configurations: each has a list of projects to trigger, a condition for when to trigger them (based on the result of the current build), and a parameters section.

There is also a Parameterized Remote Trigger Plugin in case you want to trigger a build on a different/remote Jenkins Master.

The parameters section can contain a combination of one or more of the following:

- a set of predefined properties
- properties from a properties file read from the workspace of the triggering build
- the parameters of the current build
- Subversion revision: makes sure the triggered projects are built with the same revision(s) of the triggering build. You still have to make sure those projects are actually configured to checkout the right Subversion URLs.
- Restrict matrix execution to a subset: allows you to specify the same combination filter expression as you use in the matrix project configuration and further restricts the subset of the downstream matrix builds to be run.

The parameter section is itself pluggable, and other plugins can contribute other sources of parameters.

This triggering mechanism can be used both as a post-build step or as a build step, in which case you can also block for the completion of the triggered builds. This lets you create a "function call" like semantics.

*** YOU MUST DEFINE THE PARAMETER IN DOWNSTREAM JOBS VIA "This project is parameterized". For example, if job1 passes ABC=123 to job2 then in job2 mark the job as "This project is parameterized" and "Add Parameter" named "ABC". ***

Usage as a Build step

When using the "Trigger/Call builds on another project" item.

If the trigger is configured with the "Block until the triggered projects finish their builds" enabled, the following Environment variables are made available for further build steps

Env variables for future build steps

- LAST_TRIGGERED_JOB_NAME="Last project started"
- TRIGGERED_BUILD_NUMBER_<project name>="Last build number triggered"
  from version 2.17 onwards
- TRIGGERED_JOB_NAMES="Comma separated list of all triggered projects"
- TRIGGERED_BUILD_NUMBERS_<project name>="Comma separated list of build numbers triggered"
- TRIGGERED_BUILD_RESULT_<project name>="Last triggered build result of project"
- TRIGGERED_BUILD_RESULT_<project name>_<build number>="Result of triggered build for build number"
- TRIGGERED_BUILD_RUN_COUNT_<project name>="Number of builds triggered for the project"

From 2.17 onwards

All Project names have characters not a-zA-Z or 0-9 replaced by _(multiple characters are condensed into a single _).

Note that with the BuildStep a variable can be used for the project name, I.E. ${projectName}.

Please submit bugs and feature requests to the issue tracker and not (only) in the comments.

Use of the plugin in a Matrix job

Post build task

When using the trigger parameterized build as a post build task for a matrix job the triggering will be be done once when all of the different matrix configurations have completed.

In this case some of the Environment variables may not be resolvable as passing them to downstream jobs will fail. You also cannot use a variable for the downstream project name. If this functionality is needed, the BuildStep must be used.
Environment variables that should be available are the default shell ones (<yourserver:port>/env-vars.html) and ones defined as Parameters. Variables added by the other plugins as buildwrappers may not be available.

Build step

When using the trigger parameterized build as a buildstep it will be called for every different configuration, so if triggering another project with no parameters it will be done the same number of times as you have configurations, possible causing the triggered job to run more than once.

However this also allows you to trigger other jobs with parameters relating to the current configuration, i.e. triggering a build on the same node with the same JDK.

Plugins contributing additional parameter types to this plugin

- **Git Plugin** — This plugin allows use of Git as a build SCM, including repository browsers for several providers. A recent Git runtime is required (1.7.9 minimum, 1.8.x recommended). Interaction with the Git runtime is performed by the use of the Git Client Plugin, which is only tested on official git client. Use exotic installations at your own risk.

- **NodeLabel Parameter Plugin** — This plugin adds two new parameter types to job configuration - node and label, this allows to dynamically select the node where a job/project should be executed.

Backward compatibility with version 2.22

- Since Parameterized Trigger 2.23, there are cases that Parameterized Trigger fails to trigger downstream builds that can be successfully triggered with Parameterized Trigger <= 2.22.
  - This is caused by the new behavior introduced in Parameterized Trigger 2.23. It gets to pass parameter values not directly to the downstream build, but to parameter definitions of downstream projects. This enables parameter definitions perform its specific process, for example, selecting nodes with NodeLabel Parameter Plugin.
  - Example: There is a project with a choice parameter with choices A, B, C. When you triggered that project with parameter value D, it fails with following output in the upstream:

```plaintext
[output]
```
java.lang.IllegalArgumentException: Illegal choice: D
   at hudson.model.ChoiceParameterDefinition.checkValue(ChoiceParameterDefinition.java:72)
   ...

- This is taken as a designated behavior.
  - As those failures are ones designed by parameter definitions. For example, the choice parameter is designed not to accept unexpected values.
  - You will face same problem when you triggered those builds with Jenkins CLI or Remote access API.
- It is recommended to fix your project configuration to have parameter definitions not fail.
  - For example, use EnvInject Plugin to process values or use Extensible Choice Parameter plugin which provides a choice parameter accepting edited values.
- As backward compatibility, you can make it work without fix by setting Java system property hudson.plugins.parameterizedtrigger.
  ProjectSpecificParametersActionFactory.compatibility_mode to true.
  - It can be done with launching Jenkins as followings:

    ```
    java -Dhudson.plugins.parameterizedtrigger.ProjectSpecificParametersActionFactory.compatibility_mode=true -jar jenkins.war
    ```

- In RedHat Linux systems, you can modify /etc/sysconfig/jenkins as following:

    ```
    -JENKINS_JAVA_OPTIONS="-Djava.awt.headless=true"
    +JENKINS_JAVA_OPTIONS="-Djava.awt.headless=true -Dhudson.plugins.parameterizedtrigger.
      ProjectSpecificParametersActionFactory.compatibility_mode=true"
    ```

- In Debian Linux systems, you can modify /etc/defaults/jenkins as following:

    ```
    -JAVA_ARGS="-Djava.awt.headless=true"  # Allow graphs etc. to work even when an X server is present
    +JAVA_ARGS="-Djava.awt.headless=true -Dhudson.plugins.parameterizedtrigger.
      ProjectSpecificParametersActionFactory.compatibility_mode=true"  # Allow graphs etc. to work even when an X server is present
    ```

- In Windows system, you can modify jenkins.xml placed in the installed folder.

    ```
    <arguments>-Xrs -Xmx256m -Dhudson.lifecycle=hudson.lifecycle.WindowsServiceLifecycle -jar "%BASE\jenkins.war" --httpPort=8080</arguments>
    +<arguments>-Xrs -Xmx256m -Dhudson.lifecycle=hudson.lifecycle.WindowsServiceLifecycle -Dhudson.
      plugins.parameterizedtrigger.ProjectSpecificParametersActionFactory.compatibility_mode=true -jar "%BASE\jenkins.war" --httpPort=8080</arguments>
    ```

## Changelog

### 2.35.1 (July 12, 2017)
- Prevent NullPointerException when Parameterized Trigger starts multiple builds and rejects some of them due to the missing Job/Build permission (JENKINS-45471)

### 2.35 (July 10, 2017)
- Fix security issue

### 2.34 (Jun 19, 2017)
- Update Jenkins core minimal requirement to 1.642.3
- Trigger config: Handle missing ancestor job when selecting Downstream Projects in non-job items (JENKINS-32527)

### 2.33 (Feb 26, 2017)
- Allow passing current build parameters to the projects being triggered via Parameterized Post-Build Trigger (PR #99)
- Allow setting downstream projects to be triggered from child projects (PR #106)
- Prevent error when triggering a list of project with a trailing comma (PR #111)
- Prevent NullPointerException in the case of race condition with project enabling/disabling (PR #110)
- Ensure that FileParameter file is closed after its reading when constructing parameters (PR #102)
2.32 (Jul 26, 2016)
- Log a message instead of throwing a NullPointerException when not being able to load a build's workspace (PR #105)

2.31 (Dec 16, 2015)
- Convert to use the new plugin parent pom (JENKINS-34474)

2.30 (Dec 16, 2015)
- Fix the sub project list visualization if a user has Item.DISCOVER without Item.READ (JENKINS-31727)
- Minor fixes and improvements (PR #91)

2.29 (Sep 10, 2015)
- Workflow compatibility fixes
- Extra unit tests for JENKINS-30040

2.28 (Mar 7, 2015)
- Support firing downstream workflow jobs via trigger (issue #26050)

2.27 (Mar 7, 2015)
- Allow using non-default artifact manager for properties file (issue #28980)

2.26 (Mar 7, 2015)
- Trim file names in property file lists
- Updated the version of depending subversion-plugin to 1.38 with svnkit-1.3.6-jenkins-2 that supports TLS. (JENKINS-25772)
- Add FAILURE_OR_BETTER Trigger Threshold. (JENKINS-26029)
- Rely on archived artifacts to trigger (parameterized) build fallback to workspace for backward compatibility. (JENKINS-25192)
- Added export of triggered builds and jobs to XML/JSON API. (JENKINS-26031)
- Updated required core version to 1.580.1 - the plugin requires now Java 6 (the same as Jenkins Core itself)

2.25 (Jun 1, 2014)
- FIXED: useMatrixChild in FileBuildParameters cannot be configured at all. (JENKINS-22705)
- Support absolute paths in "Parameters from properties file". (JENKINS-23084)
- Allow using parameter files even if no workspace exists. (JENKINS-22229)
- FIXED: Checking project names in configuration page works wrong with Cloudbees Template plugin. (JENKINS-22856)

2.24 (Mar 16, 2014)
- This is a release to provide backward compatibility with Parameterized Trigger Plugin <= 2.22 (JENKINS-22125).
  - See #Backward compatibility with version 2.22 for details.

2.23 (Mar 09, 2014)
- Now attempts to convert parameter strings to the type defined in the project
  - When the parameter definition is subclass of SimpleParameterDefinition, Parameterized Trigger plugin tries to create a parameter value by calling SimpleParameterDefinition#createValue, which is used when triggering builds via CLI.
- Avoid NPE by checking the getLastSuccessfulBuild() for null before calling resolveProject() (JENKINS-20932)
- Supports property files with non-ascii characters. This feature only works properly in Java 1.6, but Java 1.5 is still supported. (JENKINS-19990, JENKINS-20651)
- Matrix project support for "Parameters from properties file". (JENKINS-21013)
- Added license notice (MIT) (JENKINS-21270)

2.22 (Dec 13, 2013)
- Fixed projects.size to projects.size() in groovy script. This caused exception with IBM Java 6. (JENKINS-20719)
- expand variable in project names looking for unresolved names
- Triggered subprojects displayed in build status pages now preserves the triggered order.

2.21 (Oct 06, 2013)
- Supported hierarchical job model (cloudbees folders)
- Avoids NPE when used as a build step and the triggered job is deleted or renamed (JENKINS-19793)

2.20 (Aug 26, 2013)
- update/delete project references in conditional buildsteps (JENKINS-18967)
- Removed unnecessary plugin dependency (related to JENKINS-19152)
2.19 (Aug 11, 2013)
- Bugfix: Check existence of all subproject(s) before launching
- List configured and triggered subproject from project view (subprojects are grouped into static, dynamic and unresolved)
- set upstream builds to original builds of promotions (JENKINS-17751)

2.18 (Jun 2, 2013)
- Added a new parameter factory type that takes multiple BLOB files and trigger one build each.

2.17 (Feb 26, 2013)
Fixed Issues
- Parameterized trigger from a multi-configuration project triggers downstream jobs twice (JENKINS-11669)
- Allow specifying Git SHA1 commit id for next build(s) as Predefined parameter (JENKINS-13852)
- Counter/File Parameter Factories now can prevent triggering, skip adding parameters or fail the build if errors occur. (JENKINS-13872)
- A parameterized buildstep in a matrix project causes IllegalArgumentException (JENKINS-14278)
- Unusable environment variable TRIGGERED_BUILD_NUMBER_jobname (JENKINS-14545)
- It's not possible to specify multiple property files in parametrised trigger (JENKINS-15834)
- Boolean parameter becomes string (JENKINS-15920)
- Define implicit parameter: "build them on the same node" (JENKINS-16334)

2.16 (Oct 09, 2012)
- Error validating project name in form (JENKINS-15130)
- clarify error when no parameter set and triggerWithNoParameter unchecked (unfiled)

2.15 (May 23, 2012)
- Allow each parameter type to be selected only once (requires Jenkins core version 1.463!). (JENKINS-8916)
- Fix triggering projects based on variables (JENKINS-13875)

2.14 (Apr 27, 2012)
- Fixed infinite loop, when "Block until the triggered projects finish their builds" option is used and triggered projects are disabled. (JENKINS-12923)
- Added a new parameter definition in the trigger that lets the upstream controls which subset of the downstream matrix job to build.

2.13 (Feb 9, 2012)
- Builds triggered as build steps will now appear as downstream builds. (JENKINS-11082, JENKINS-9263, JENKINS-5184)
- Info on builds kicked off as blocking build steps exposed. (JENKINS-11345)
- When multiple builds are launched as blocking build steps, all should be listed as "waiting to complete", and should be listed in launched order. (JENKINS-11111, JENKINS-11117)
- Fixed a compatibility with hierarchical jobs (such as the CloudBees folder plugin.)
- Added a mechanism to repeatedly invoke jobs for each file that matches GLOB (pull #11)

2.12 (Oct 30, 2011)
- Added a mechanism to repeatedly invoke the same project with different set of parameters.
- Improved the default value of the blocking configuration.
- Added hyperlinks to console output

2.11 (Aug 6, 2011)
- Added “unstable or worse” condition (JENKINS-9858).

2.10 (Jul 10, 2011)
- Implemented JENKINS-10028: Textbox for downstream projects should be validated as with the "Build other projects" text box.
- Implemented JENKINS-8788: When renaming a job, the parameterized trigger should reflect the new name if the job appears as a parameterized triggered job.

2.9 (Jul 10, 2011)
- Fix subversion plugin dependency.
- Multiple triggers between the same upstream/downstream project combo will now work, though this will also depend on Jenkins core 1.414. (JENKINS-8985)
• Implemented JENKINS-9217: Added a new option to not run further build steps if triggered builds fail or are unstable
• Implemented JENKINS-9391: Allow triggering projects based on variables

2.8 (Mar 27, 2011)
• Prevent IndexOutOfBoundsException, when no projects are specified.
• Added new "Trigger build without parameters" option for post-action trigger builds.

2.7 (Mar 1, 2011)
• Rerelease 2.6 to properly set required Jenkins version.

2.6 (Feb 17, 2011)
• Improved the progress report to show what jobs are being executed when used as a builder.
• Fixed NPE when the child build is called concurrently by other jobs.

2.5 (Feb 12, 2011)
• Added a mechanism to call other builds synchronously and map their results.

2.4 (Jul 29, 2010)
• Fix passing of parameters when a maven project is the upstream job. (JENKINS-8141)
• Add support for multiconfiguration(matrix) projects. (JENKINS-5687)
• Fix variable expansion in "predefined parameters" so backslashes in variable values are not lost. Add help text about backslash handling. (JENKINS-6004)
• File parameters are currently not reusable, so omit these when "current build parameters" is selected. (JENKINS-6777)
• Update trigger settings when other jobs are renamed or deleted. (JENKINS-6483)
• Add an "always trigger" option. (JENKINS-6656)

2.3 (Jan 16, 2010)
• NOTE: This release is built against Jenkins 1.341 but works with Jenkins 1.319 or higher.
• Implement Jenkins API so connected jobs show in Upstream/Downstream Projects list on job pages with Jenkins 1.341 or higher. (JENKINS-5184)
• Merge together parameter lists from multiple sources to avoid multiple Parameters links on build pages (JENKINS-5143) and failure to pass some parameters further along the downstream chain (JENKINS-4587).

2.2 (Jan 11, 2010)
• Restore compatibility with Java 5. (JENKINS-5231)
• Rollback change for JENKINS-5184, as this caused builds to be triggered twice. (JENKINS-5232)

2.1 (Jan 9, 2010)
• Implement Jenkins API so connected jobs show in Upstream/Downstream Projects list on job pages. (JENKINS-5184)

2.0 (Aug 10, 2009)
Major refactoring. Now supports any combination of projects to build, result condition and set of parameter sources.
Should be backward compatible for configuration, except the 'batch condition' which was removed.

1.6 (Jul 18, 2009)
• Removed downstream projects parameters duplication.

1.5 (Jul 5, 2009)
• Added batch condition: a batch/shell script can be used to decide if project[s] built should be triggered.
• If downstream project parameters values are not specified explicitly their default values are used now (empty values was used in previous versions).

1.4 (Jun 11, 2009)
• Environment variables (including current build parameters) can now be used in the downstream build parameters and in parameters file name.

1.3 (Feb 28, 2009)
• Trigger another build when the current is unstable

1.2 (Feb 27, 2009)
• Fixes incompatibility with Jenkins 1.285. Jenkins 1.286 is a minimum requirement for this version.
## 1.0, 1.1 (Feb 9, 2009)
- Initial release

<table>
<thead>
<tr>
<th>type</th>
<th>priority</th>
<th>key</th>
<th>summary</th>
</tr>
</thead>
</table>

⚠️ Can't show details. Ask your admin to whitelist this Jira URL.

View these issues in Jira