

# Git Parameter Plugin

## Plugin Information

View Git Parameter [on the plugin site](#) for more information.

**Adds ability to choose branches, tags or revisions from git repository configured in project.**

- [Plugin Info](#)
- [Basic configuration](#)
  - [Project configuration](#)
  - [Build with Parameters form](#)
- [Example pipeline script](#)
  - [Branch type - Basic usage](#)
  - [Tag type](#)
  - [Branch Tag type](#)
  - [Revision type](#)
  - [Pull Request type](#)
- [Options:](#)
  - [Parameter Type](#)
  - [Branch](#)
  - [Branch Filter](#)
  - [Tag Filter](#)
  - [Sort Mode](#)
  - [Default Value](#)
  - [Selected Value](#)
  - [Use repository](#)
  - [Quick Filter](#)
  - [List Size](#)
- [Global configuration](#)
- [Error handling](#)
- [Releases schedule](#)
- [Changelog](#)

## Plugin Info

This plugin allows you to assign git branch, tag, pull request or revision number as parameter in your builds.

### **Important!**

There is no need to set up anything special in plugin settings.  
*This plugin will read GIT SCM configuration from your projects.*  
This plugin used directly the [Git Plugin](#) and [Git Client Plugin](#).

## Basic configuration

### [Project configuration](#)

**General** | Source Code Management | Build Triggers | Build Environment | Build | Post-build Actions

This project is parameterized

**Git Parameter**

Name:

Description:

[Safe HTML] [Preview](#)

Parameter Type:

[Advanced...](#)

[Add Parameter](#)

Throttle builds

Disable this project

Execute concurrent builds if necessary

Restrict where this project can be run

[Advanced...](#)

---

### Source Code Management

None

Git

Repositories

Repository URL:

Credentials:  [Add](#)

[Advanced...](#)

[Add Repository](#)

Branches to build

Branch Specifier (blank for 'any'):

[Add Branch](#)

Git executable:

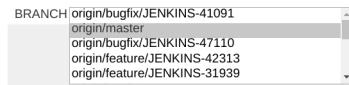
Repository browser:

Additional Behaviours: [Add](#)

Build with Parameters form

## Project git\_parameter

This build requires parameters:



If you selected revisions as a type of presented data and working space is empty, the plug needs clone the repository, before it can display the contents. This may take some time if you have a slow connection or repository is large.

Build

## Example pipeline script

Important! version 0.9.4 or later

### Branch type - Basic usage

#### Pipeline: Branch type - Basic usage (Declarative Pipeline)

```
// Using git without checkout
pipeline {
  agent any
  parameters {
    gitParameter branchFilter: 'origin/(.*)', defaultValue: 'master', name: 'BRANCH', type: 'PT_BRANCH'
  }
  stages {
    stage('Example') {
      steps {
        git branch: "${params.BRANCH}", url: 'https://github.com/jenkinsci/git-parameter-plugin.git'
      }
    }
  }
}
```

#### Pipeline: Branch type - Basic usage (Scripted Pipeline)

```
properties([
  parameters([
    gitParameter(branch: '',
      branchFilter: 'origin/(.*)',
      defaultValue: 'master',
      description: '',
      name: 'BRANCH',
      quickFilterEnabled: false,
      selectedValue: 'NONE',
      sortMode: 'NONE',
      tagFilter: '*',
      type: 'PT_BRANCH')
  ])
])
node {
  git branch: "${params.BRANCH}", url: 'https://github.com/jenkinsci/git-parameter-plugin.git'
}
```

Important settings:

- It should be set a **default** value because initial build must get this information
- Using **git** should be set a **branchFilter** as *'origin/(.\*)'* (origin is a remote server name)

### Parameter type

- PT\_TAG
- PT\_BRANCH
- PT\_BRANCH\_TAG

- PT\_REVISION
- PT\_PULL\_REQUEST

**Important: If you need use other type (other than branch) parameter, you must use git within *checkout***

## Tag type

### Pipeline: Tag type

```
// Using git within checkout
pipeline {
  agent any
  parameters {
    gitParameter name: 'TAG',
                 type: 'PT_TAG',
                 defaultValue: 'master'
  }
  stages {
    stage('Example') {
      steps {
        checkout([$class: 'GitSCM',
                 branches: [[name: "${params.TAG}"]],
                 doGenerateSubmoduleConfigurations: false,
                 extensions: [],
                 gitTool: 'Default',
                 submoduleCfg: [],
                 userRemoteConfigs: [[url: 'https://github.com/jenkinsci/git-parameter-plugin.git']]
                ])
      }
    }
  }
}
```

## Branch Tag type

### Pipeline: Branch Tag type

```
pipeline {
  agent any
  parameters {
    gitParameter name: 'BRANCH_TAG',
                 type: 'PT_BRANCH_TAG',
                 defaultValue: 'master'
  }
  stages {
    stage('Example') {
      steps {
        checkout([$class: 'GitSCM',
                 branches: [[name: "${params.BRANCH_TAG}"]],
                 doGenerateSubmoduleConfigurations: false,
                 extensions: [],
                 gitTool: 'Default',
                 submoduleCfg: [],
                 userRemoteConfigs: [[url: 'https://github.com/jenkinsci/git-parameter-plugin.git']]
                ])
      }
    }
  }
}
```

## Revision type

### Pipeline: Revision type

```
pipeline {
  agent any
  parameters {
    gitParameter name: 'REVISION',
                 type: 'PT_REVISION',
                 defaultValue: 'master'
  }
  stages {
    stage('Example') {
      steps {
        checkout([$class: 'GitSCM',
                  branches: [[name: "${params.REVISION}"]],
                  doGenerateSubmoduleConfigurations: false,
                  extensions: [],
                  gitTool: 'Default',
                  submoduleCfg: [],
                  userRemoteConfigs: [[url: 'https://github.com/jenkinsci/git-parameter-plugin.git']]
                ])
      }
    }
  }
}
```

## Pull Request type

### Pipeline: Pull Request type

```
pipeline {
  agent any
  parameters {
    gitParameter name: 'PULL_REQUESTS',
                 type: 'PT_PULL_REQUEST',
                 defaultValue: '1',
                 sortMode: 'DESCENDING_SMART'
  }
  stages {
    stage('Example') {
      steps {
        checkout([$class: 'GitSCM',
                  branches: [[name: "pr/${params.PULL_REQUESTS}/head"]],
                  doGenerateSubmoduleConfigurations: false,
                  extensions: [],
                  gitTool: 'Default',
                  submoduleCfg: [],
                  userRemoteConfigs: [[refspec: '+refs/pull/*:refs/remotes/origin/pr/*', url: 'https://github.com/jenkinsci/git-parameter-plugin.git']]
                ])
      }
    }
  }
}
```

## Options:

### Parameter Type

#### Name using in pipeline

type: 'PT\_TAG' or 'PT\_BRANCH' or 'PT\_BRANCH\_TAG' or 'PT\_REVISION' or 'PT\_PULL\_REQUEST'

Explains about PT\_TAG or PT\_BRANCH or PT\_BRANCH\_TAG:

Plugin using [git ls-remote](#) command to get remote tags or branches, this solution was implemented in

[JENKINS-40232](#) - Getting issue details...

STATUS

. This has been done for performance reasons. The plugin does not have to fetch the repository before getting information

In code plugin using `getRemoteReferences` from `GitClient`, look implementation in [CliGitAPIImpl](#)

```
package org.jenkinsci.plugins.gitclient
//...

public interface GitClient {
//...
    Map<String, ObjectId> getRemoteReferences(String remoteRepoUrl, String pattern, boolean headsOnly,
boolean tagsOnly) throws GitException, InterruptedException;
//...
}
```

## Branch

Name using in pipeline

branch

## Branch Filter

Name using in pipeline

branchFilter

## Tag Filter

Name using in pipeline

tagFilter

## Sort Mode

Name using in pipeline

sortMode: 'NONE' or 'ASCENDING\_SMART' or 'DESCENDING\_SMART' or 'ASCENDING' or 'DESCENDING'

You can select the following sorting options for tags/revision/branches/branches\_or\_tags/pull requests

- none
- descending
- ascending
- ascending smart
- descending smart

For the smart variants the compare treats a sequence of digits as a single character. Contributed by Graeme Hill.

## Default Value

Name using in pipeline
defaultValue

In release 0.9.9 or later good to set a default value, because this value is using in the initial build (in Pipeline). Default value is returned when some error occurred on getting data.

Default Value	<input type="text"/>	
 Default Value is required. Example origin/master		

## Selected Value

Name using in pipeline
selectedValue: 'NONE' or 'TOP' or 'DEFAULT'

## Use repository

Name using in pipeline
useRepository

**Remember:** You don't set a git repository into the plugin, this plugin using git repositories which are defined in project in SCM section!

If in the task are defined multiple repositories, this option specifies which the repository is taken into account on getting data.

*If the option is not defined, is taken a first defined repository.*

This option is a regular expression, which is compared to the 'Repository URL'.

You can define the multiple SCM for few way, you can use [Multiple SCMs Plugin](#), specified many 'Repository URL' in one SCM or define them in pipeline.

Consider an example based on two repositories:

- <https://github.com/klimas7/exampleA.git>
- <https://github.com/klimas7/exampleB.git>

### Pipeline: Complex example

```
pipeline {
  agent any
  parameters {
    gitParameter branchFilter: 'origin.*/(.*)', defaultValue: 'master', name: 'BRANCH_A', type:
'PT_BRANCH', useRepository: '.*exampleA.git'
    gitParameter branchFilter: 'origin.*/(.*)', defaultValue: 'master', name: 'BRANCH_B', type:
'PT_BRANCH', useRepository: '.*exampleB.git'
  }
  stages {
    stage('Example') {
      steps {
        git branch: "${params.BRANCH_A}", url: 'https://github.com/klimas7/exampleA.git'
        git branch: "${params.BRANCH_B}", url: 'https://github.com/klimas7/exampleB.git'
      }
    }
  }
}
```

After initial run you get

## Pipeline MultiSCM\_pipeline

This build requires parameters:

BRANCH\_A exA\_branch\_1  
exA\_branch\_2  
master

If you selected revisions as a type of presented data :

BRANCH\_B exB\_branch\_1  
exB\_branch\_2  
master

If you selected revisions as a type of presented data :

Build

Example when 'Use repository' is not set:

### Pipeline: Use repository is not set

```
pipeline {
  agent any
  parameters {
    gitParameter branchFilter: 'origin.*/(.*)', defaultValue: 'master', name: 'BRANCH', type: 'PT_BRANCH'
  }
  stages {
    stage('Example') {
      steps {
        git url: 'https://github.com/klimas7/exampleA.git'
        git url: 'https://github.com/klimas7/exampleB.git'
      }
    }
  }
}
```

After initial run you get



# Pipeline MultiSCM\_pipeline\_v2

This build requires parameters:

BRANCH

- exA\_branch\_1
- exA\_branch\_2
- master

If you selected revisions as a type of presented data and work

Build

## Quick Filter

Name using in pipeline

quickFilterEnabled

## List Size

Name using in pipeline

listSize

## Global configuration

Release 0.9.9 or later

Git Parameter

Show 'need to clone' information

The message should be displayed:

'If working space is empty, the plug needs clone the repository, before it can display the contents. This may take some time if you have a slow connection or repository is large.'

The message is displayed under the selection field. Only for revisions as a parameter type.

```
REVISION 463eda98 2019-02-16 22:11 klimas7 <klimas7@gmail.com> [maven-release-plugin] prepare for next development iteration
39579f2a 2019-02-16 22:11 klimas7 <klimas7@gmail.com> [maven-release-plugin] prepare release git-parameter-0.9.8
a09d59c9 2019-02-16 21:57 klimas7 <klimas7@gmail.com> Release 0.9.8
ed069232 2019-02-16 21:57 Boguslaw Klimas <klimas7@gmail.com> Merge pull request #77 from jenkinsci/feature/JENKINS-54359 JENKINS-54359
3959c399 2019-02-16 21:50 klimas7 <klimas7@gmail.com> [maven-release-plugin] prepare for next development iteration
```

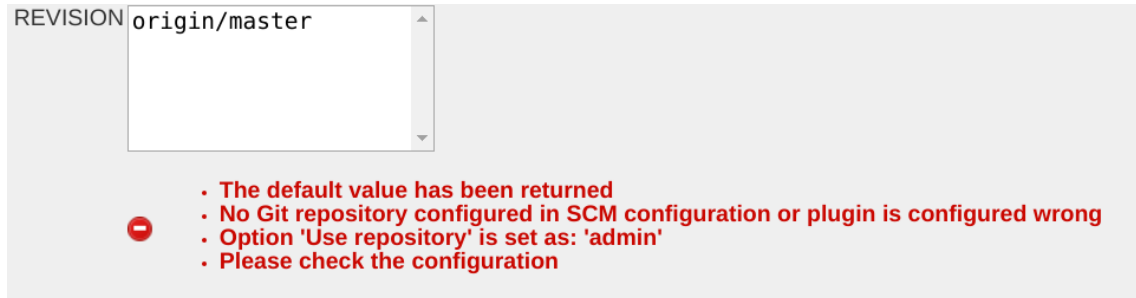
If working space is empty, the plug needs clone the repository, before it can display the contents. This may take some time if you have a slow connection or repository is large.

## Error handling

Release 0.9.9 or later

If an error occurred while retrieving data, the default value is returned.  
Additional information is provided below, along with the cause of the error.

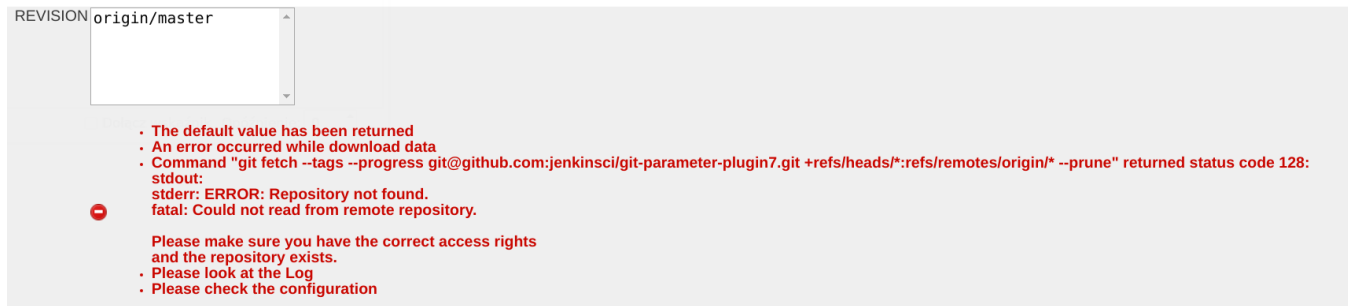
Examples:



REVISION origin/master

- The default value has been returned
- No Git repository configured in SCM configuration or plugin is configured wrong
- Option 'Use repository' is set as: 'admin'
- Please check the configuration

This error will occurred when repository is not configured or 'Use repository' option no match with any repository.



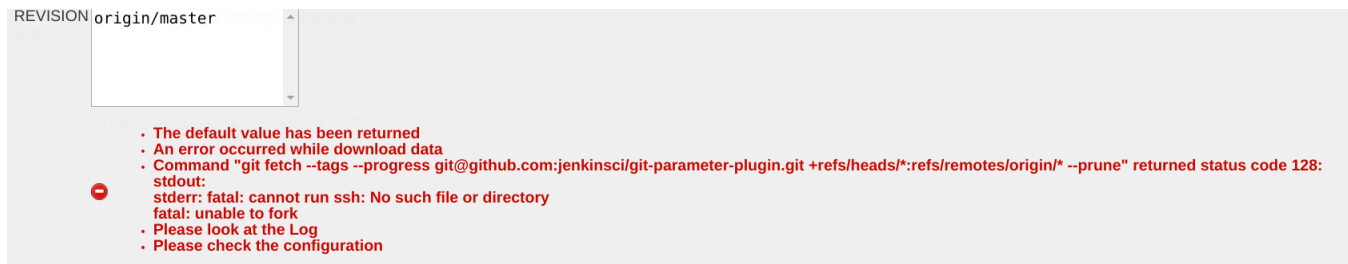
REVISION origin/master

- The default value has been returned
- An error occurred while download data
- Command "git fetch --tags --progress git@github.com:jenkinsci/git-parameter-plugin.git +refs/heads/\*:refs/remotes/origin/\* --prune" returned status code 128:  
stdout:  
stderr: ERROR: Repository not found.  
fatal: Could not read from remote repository.

Please make sure you have the correct access rights and the repository exists.

- Please look at the Log
- Please check the configuration

This error will occurred when repository not exists or URL is wrong.



REVISION origin/master

- The default value has been returned
- An error occurred while download data
- Command "git fetch --tags --progress git@github.com:jenkinsci/git-parameter-plugin.git +refs/heads/\*:refs/remotes/origin/\* --prune" returned status code 128:  
stdout:  
stderr: fatal: cannot run ssh: No such file or directory  
fatal: unable to fork

- Please look at the Log
- Please check the configuration

This error will occurred when is no ssh command on Jenkins master.

## Releases schedule

?

## Changelog

### 16/02/2019 Version 0.9.9

- [JENKINS-55769](#): Tag match filter shows more entries than direct command (git tag -l "\$tagFilter")
- [JENKINS-55770](#): Intrusive and misleading warning text from the parameter selection display
- [JENKINS-54359](#): Change error handling

### 09/10/2018 Version 0.9.6

- [JENKINS-53911](#): Duplicate entries in list, if git repo is checked out twice.
- [JENKINS-52533](#): Display commit message on the build action
- [JENKINS-45925](#): Show git commit message when building with parameter "revision"

### 17/09/2018 Version 0.9.5

- [JENKINS-51041](#), [JENKINS-50510](#), [JENKINS-45419](#): Fixed use repository option

## 16/08/2018 Version 0.9.4

- [JENKINS-52273](#): Support git-parameter-plugin in declarative pipeline
- [JENKINS-52132](#): Description with HTML support

## 20/06/2018 Version 0.9.3

- [JENKINS-51521](#): Git parameter does not show branch list in case deleteDir() is present in pipeline
- [JENKINS-51476](#): Git parameter plugin is not retrieving revision number

## 16/04/2018 Version 0.9.2

- [JENKINS-50776](#): Default selected item doesn't always honor exactly Default Value
- [JENKINS-49727](#): Add optional parameter to specify the number of items the list will display

## 18/02/2018 Version 0.9.1

- [JENKINS-45419](#): 'Use Repository' setting does not find other remote urls if multiple repos are added to job
- [PR #55](#): Add complete French support

## 02/11/2017 Version 0.9.0

- [JENKINS-47110](#): Retrieving Git references do not work with variable in Repository URL
- [PR #54](#): Help improvement: complete English translation, and reorder items
- [JENKINS-47078](#): IndexOutOfBoundsException for pipeline job
- [JENKINS-39530](#): Add support to Pipeline projects

## 04/09/2017 Version 0.8.1

- [JENKINS-46216](#): Null Pointer exception when no default parameter provided
- [JENKINS-45577](#): [Git Parameter Plugin] Parameter does not support the definition from CLI
- [JENKINS-46624](#): fix remote name
- [JENKINS-46185](#): Set browser focus to filter after the QuickFilter has been filled
- [JENKINS-46038](#): Extend list of supported type with pull request
- [JENKINS-26799](#): Multiple SCMs plugin support part 3 (Work fine when selected revisions)
- [JENKINS-42313](#): Default Value not honoured

## 02/06/2017 Version 0.8.0

- [JENKINS-26799](#): Multiple SCMs plugin support part 2
- [JENKINS-40523](#): Include Jenkins Project Name in Log message
- [JENKINS-40232](#): Git Parameter Plugin doesn't need to clone

## 23/01/2017 Version 0.7.2

- [JENKINS-41091](#): git-parameter:0.7.1 breaks the multi-line parameters in rebuild

## 11/27/2016 Version 0.7.1

- [JENKINS-39366](#): Add support for a rebuild-plugin
- [JENKINS-26799](#): Multiple SCMs plugin support

## 09/12/2016 Version 0.7.0

- [JENKINS-37555](#): Better support for internationalization
- [JENKINS-37595](#): Add support for polish localization
- [JENKINS-37370](#): Retrieving Git references do not work with variable in "Repository URL"
- [JENKINS-37953](#): Add support to ProxySCM
- [JENKINS-37738](#): Update dependency in plugin and cleanup in pom
- [JENKINS-34876](#): Git Parameters not working for Pipeline projects and Jenkinsfile from SCM

## 08/06/2016 Version 0.6.2

- [JENKINS-36833](#): Race Condition Populating Multiple Tag Parameters
- [JENKINS-36934](#): No return value passed to the url
- [JENKINS-31939](#): The top value is better to be chosen by default of to have such option (part 3)

## 07/19/2016 Version 0.6.1

- [JENKINS-31939](#): The top value is better to be chosen by default of to have such option (part 2)

## 07/06/2016 Version 0.6.0

- [JENKINS-36104](#): Add Repo SCM support (Derron Hu)
- [JENKINS-16290](#): git parameter plugin doesn't support Jenkins slave setup with git repos checked out only on slaves
- [JENKINS-35363](#): Git parameter filter doesn't work

## 05/03/2016 Version 0.5.1

- [JENKINS-34425](#): Git parameter plugin skips one build number while populating tags
- [JENKINS-34544](#): NPE After update to 0.5.0
- [JENKINS-34574](#): After cleaned workspace doesn't show branches

## 04/02/2016 Version 0.5.0

### 1. User interface

- [JENKINS-27435](#): Quick branch filter (Thank Bruno P. Kinoshita for inspiration)
- [JENKINS-33963](#): Branch filter does not save the value (part of the work Joe Hansche)

### 2. Refactor/fix/feature

- [JENKINS-33361](#): Long release number in branch, tag or revision name
- [JENKINS-33084](#): Git Parameter plugin should prune stale remote branches (@darashenka)
- [JENKINS-31939](#): The top value is better to be choosed by default of to have such option
- [JENKINS-33831](#): Revision Parameter Type: ArrayIndexOutOfBoundsException
- [JENKINS-33912](#): Refactoring Test Case

## 01/16/2015 Version 0.4.0

- Possibility to select branch, tag or branch (Alban Dericbourg)
- Keep complex logics in jelly as less as possible (Yestin Sun)
- support folders (Nicolas De Loof)
- Minimized pom.xml as suggested by Jesse Glick
- Removed LICENSE.txt as suggested by Jesse Glick

## 05/14/14 Version 0.3.2

### User visible changes are:

- Updated help texts for configuration and when selecting your tag/revision
- Runs a fetch each time the user enters the "Build with parameter".
- Run clone when fetch fails on workspace empty (Gabor Liptak)
- Merging SortMode from graeme-hill
- With an empty workspace the tags are calculated after we made a checkout. This may take a long time.

### Changes relevant to developer

- Added MIT-LICENSE.txt to use the same license as Jenkins-CI.
- Added Contributors.textile
- Display month not minutes in date. Add HH.mm. Display only first 8 chars of SHA1 (Niklaus Giger)
- Add backup pluginRepository (Gabor Liptak)
- Use GitTool to query configured location of the git executable (gliptak)
- Upgrade to git 2.2.0. (christ66)
- Build against latest stable Jenkins-CI version 1.554.1
- New co-maintainer/developer Niklaus Giger (id ngiger)
- Version 0.3 and 0.3.1 never made it to the distribution, because of problems with the release mechanism.

## 02/21/12 Version 0.2

- Corrected error - plugin wasn't showing anything after change of main Git Plugin
- Corrected major dis-functionality - plugin now it showing revisions only from correct job/project.
- Adding support for choosing branch from which revisions/tags are returned

## 11/01/11 Version 0.1

- Initial Release

Adds ability to choose branches, tags or revisions from git repositories configured in project.