

How to - Integrate multi branch pipeline in jenkins

How to setup Jenkins Multi branch Pipeline jobs:

Prerequisite

1. Jenkins (2.x version)
2. Jenkins Multibranch Pipeline is installed.

The Multi branch Pipeline allows you to automatically create a pipeline for each branch on your repository with the help of Jenkinsfile.

Jenkinsfile is stored along with your source code inside a Source Code Management (SCM) eg. GitHub, BitBucket, etc.

The Workflow Multi branch feature (provided by the workflow plugin) provides the following key abilities:

1. Automatic Workflow job creation in Jenkins per new branch in the repo.
2. Build specific to that child-branch and its unique scm change and build history.

The Multibranch Pipeline project type enables you to implement different Jenkinsfiles for different branches of the same project. In a Multibranch Pipeline project, Jenkins automatically discovers, manages and executes Pipelines for branches which contain a Jenkinsfile in source control.

Creating a Multi branch Pipeline job:

Configure Jenkins:

1. Open **Manage Jenkins -> Configure System Bitbucket Endpoints**

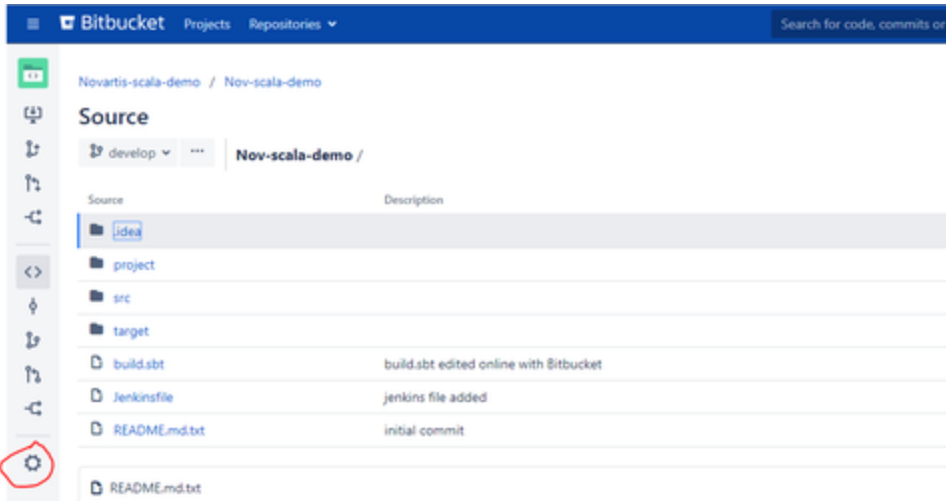
The screenshot shows the 'Bitbucket Endpoints' configuration page in Jenkins. The 'Bitbucket Server' section is expanded, showing fields for Name, Server URL, Credentials, Custom Jenkins Root URL, and Webhook implementation to use. A red circle highlights the 'Add' button in the Bitbucket Server section.

2. Add server in Bit bucket endpoint section.

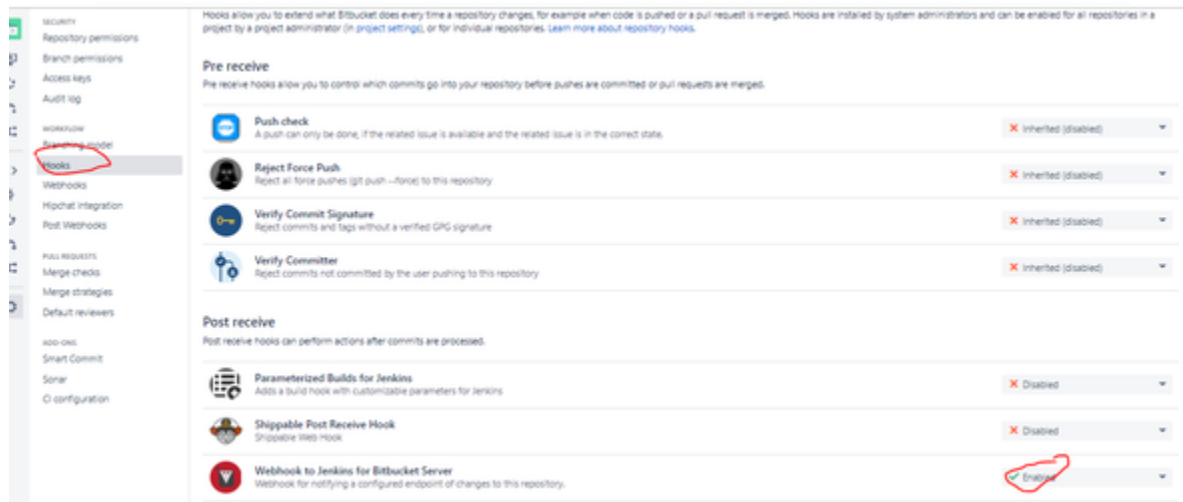
3. The add-on Post Webhooks for Bitbucket can be installed to work with any version of Bitbucket Branch Source or Bitbucket Server.

Set up in the repository settings

Go to the repository's settings.



Click Hooks in the left-hand navigation and enable for Post-Receive Webhooks item and save.



then give Jenkins instance URL.

WEBHOOK TO JENKINS FOR BITBUCKET SERVER

Jenkins instance(s) Add another Jenkins instance

Jenkins URL*

URL to the Jenkins instance. Example: https://jenkins.example.com

Jenkins Endpoint

Select notification type for Jenkins instance. Git and Bitbucket Source plugins are supported.

General settings

☐ Skip SSL Certificate Validation
When connecting to Jenkins, allow all certificates to be accepted, including self-signed certs

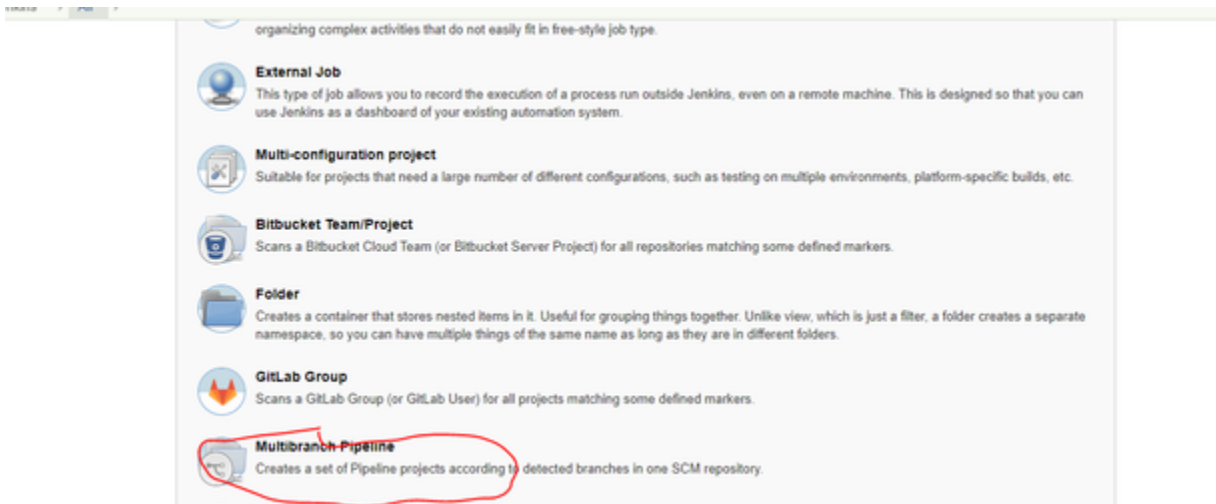
☐ Omit the Trigger Build Button
Do not display the Trigger Build Button on the pull request overview page

Configuration check

Test the configured connection with Jenkins.

Add a job on Jenkins

Step 1: Create a new item or job,click multi branch pipeline



Step 2: Select branch source option, click on add source and select Bitbucket option.



Step 3 :Add the details

Server - The server to connect to. The list of servers is configured in the Manage Jenkins » Configure Jenkins » Bitbucket Endpoints screen. The list of servers can include both Bitbucket Cloud as well as Bitbucket Server instances.

Credentials: Credentials used to scan branches

Owner : Specify the name of the Bitbucket Team or Bitbucket User Account.If using Bitbucket server ,Use project key.

Repository name : The repository to scan.

Branch Sources

Bitbucket

Server: Bitbucket Cloud (https://bitbucket.org)

Credentials: none

Owner:

Repository Name:

Behaviors:

Discover branches

Strategy: Exclude branches that are also filed as PRs

Discover pull requests from origin

Strategy: Merging the pull request with the current target branch revision

Discover pull requests from forks

Strategy: Merging the pull request with the current target branch revision

Trust: Forks in the same account

Add

Step 4 :select discover option option and select All branches

Bitbucket

Server: Bitbucket-server (http://novartis.devops.altimetrik.io:7990)

Credentials: admin/*****

Owner: NOVS

Repository Name: nov-scala-demo

Behaviors:

Discover branches

Strategy: All branches

Add

Property strategy: All branches get the same properties

Add property

Step 4 :Under Build configuration give the jenkinsfile path

Build Configuration

Mode: by Jenkinsfile

Script Path: Jenkinsfile

Commit sample jenkinsfile in repository.

```

1  @library("my_sample_demo") _
2
3  pipeline {
4    agent none
5    stages {
6      stage("Poll SCM") {
7        steps {
8          echo 'hello world Poll SCM'
9          pollSCM "Poll SCM"
10         }
11      }
12      stage("Build") {
13        steps {
14          echo 'hello world Build'
15          build "Build"
16        }
17      }
18      stage("Test") {
19        steps {
20          echo 'hello world Test'
21          testing "Test"
22        }
23      }
24      stage("Deploy") {
25        steps {
26          echo 'hello world Deploy'
27          deployBuild "Deploy"
28        }
29      }
30    }
31  }
32

```

Scan the project for every 1 minute.

Scan Multibranch Pipeline Triggers

- ☐ Build when another project is promoted
- ☒ Periodically if not otherwise run

Interval

1 minute

Once done it will scan all branches available in the repository.

Output log.

[View as plain text](#)
[Multibranch Pipeline Events](#)
[Delete Multibranch Pipeline](#)
[People](#)
[Build History](#)
[Project Relationship](#)
[Check File Fingerprint](#)
[Bitbucket](#)
[Rename](#)
[Config Files](#)
[Pipeline Syntax](#)
[Credentials](#)

```

Looking up MDVS/nov-scala-demo for branches
Checking branch feature/test from MDVS/nov-scala-demo
'Jenkinsfile' found
Met criteria
No changes detected: feature/test (still at 37d0d28279d256ed0cc0ddeb3145facc3766f8e)
Checking branch feature/pipelinecode from MDVS/nov-scala-demo
'Jenkinsfile' found
Met criteria
No changes detected: feature/pipelinecode (still at 9d546739dd5ee8ed06df6cf8291802f904edfb82)
Checking branch demo from MDVS/nov-scala-demo
'Jenkinsfile' found
Met criteria
No changes detected: demo (still at af90b1b5187e73970bd1d998897b9630de6eb7f)
Checking branch master from MDVS/nov-scala-demo
'Jenkinsfile' found
Met criteria
No changes detected: master (still at 13e3c0f1700425277b7d7981de7de6e477d39737)
Checking branch develop from MDVS/nov-scala-demo
'Jenkinsfile' found
Met criteria
No changes detected: develop (still at a16acc7113d100b0f6e2ef7431eb38cf9796cca3)

5 branches were processed
[Thu Oct 03 09:47:01 UTC 2019] Finished branch indexing. Indexing took 0.32 sec
Finished: SUCCESS

```

Build Queue

0 builds in the queue.

Build Executor Status

- Up
- Status
- Configure
- Scan Multibranch Pipeline Now
- Scan Multibranch Pipeline Log
- Multibranch Pipeline Events
- Delete Multibranch Pipeline
- People
- Build History
- Project Relationship
- Check File Fingerprint
- Bitbucket
- Rename
- Config Files
- Pipeline Syntax
- Crash/Status

nov-scala-demo

Folder name: Test-multibranch

Branches (5)		Pull requests (0)	Tags (0)			
S	W	Name	Last Success	Last Failure	Last Duration	
		default	55 min - #2	N/A	0.4 sec	
		feature/demo	53 min - #1	N/A	1.6 sec	
		feature/onlinecode	53 min - #1	N/A	1.5 sec	
		feature/test	52 min - #1	N/A	1.5 sec	
		master	54 min - #1	N/A	1.6 sec	

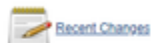
Icon: [S](#) [M](#) [L](#)

Legend RSS for all RSS for failures RSS for just latest builds

- Up
- Status
- Changes
- Build Now
- View Configuration
- Full Stage View
- Bitbucket
- Pipeline Syntax

Branch feature/demo

Full project name: Test-multibranchfeature%2Fdemo



Stage View

Average stage times: (Average full run time: ~1s)		Poll SCM	Build	Test	Deploy
		118ms	111ms	123ms	89ms
<div> <div>#1</div> <div>Oct 03 15:10</div> <div>No Changes</div> </div>		118ms	111ms	123ms	89ms

Permalinks

- Last build (#1): 58 min ago
- Last stable build (#1): 58 min ago

Build History [trend](#)

find

#1 Oct 3, 2019 9:46 AM

RSS for all RSS for failures