–

Step-by-step walkthrough for configuring and executing parallel builds

**Jenkins scripted pipeline**

PREPARED BY:

PREPARED FOR:

**Version:** 1.0  
**Department:** Architect Department  
**Delivery:** End of Month

Table of Contents

[**1.** Executive Summary 3](#_Toc209520458)

[2. Parallel Build & Test 3](#_Toc209520459)-6

[3. Example – Cross Plateform 7](#_Toc209520461)

# Executive Summary

**This document provides a comprehensive guide to setting up and executing parallel builds in a development environment. It includes detailed steps, annotated screenshots, and explanations to help developers streamline their build processes and improve efficiency.**

# Parallel Build & Test

**Step 1: Create a New Project**

* Go to **Jenkins Dashboard → New Item**
* Enter project name, e.g. ParallelBuildDemo
* Select **Pipeline** and click **OK**

# A screenshot of a computer AI-generated content may be incorrect.

**Step 3: Define Pipeline Script**

* Scroll down to **Pipeline** section.
* Choose **Pipeline script** option.
* Paste the pipeline code from your document (either Windows Parallel Build or Cross-Platform Build).

pipeline {

agent any

stages {

stage('Build & Test in Parallel') {

parallel {

stage('Build') {

steps {

echo "Building the application..."

bat 'echo Simulating Build && powershell -Command "Start-Sleep -Seconds 5"'

echo "Build done."

}

}

stage('Unit Tests') {

steps {

echo "Running unit tests..."

bat 'echo Running Unit Tests && ping -n 4 127.0.0.1 > nul'

echo "Unit tests completed."

}

}

stage('Integration Tests') {

steps {

echo "Running integration tests..."

bat 'echo Running Integration Tests && powershell -Command "Start-Sleep -Seconds 4"'

echo "Integration tests completed."

}

}

}

}

stage('Deploy') {

steps {

echo "Deploying to environment..."

bat 'echo Deployment step executed'

}

}

}

}

**A screenshot of a computer

AI-generated content may be incorrect.**

**Step 4: Save and Build**

* **Click Save.**
* **From the left menu, click Build Now.**

A screenshot of a computer

AI-generated content may be incorrect.

**Step 5: Monitor Console Output**

* After the job starts, click on the build number → **Console Output**.
* You will see logs for:
  + Build
  + Unit Tests
  + Integration Tests
  + Deploy

**Step 6: View Stages (Blue Ocean / Stage View)**

* If you have **Blue Ocean plugin** or **Pipeline Stage View**, you’ll see the stages executed in **parallel boxes**.
* Example: Build, Unit Tests, Integration Tests run side-by-side, then Deploy runs after them.

A screenshot of a computer

AI-generated content may be incorrect.

# Example – Cross Plateform

pipeline {

agent none // each stage picks its own agent

stages {

stage('Build on multiple OS') {

parallel {

stage('Linux Build') {

agent { label 'linux' }

steps {

echo "Building on Linux..."

bat 'echo Linux build running && sleep 5'

}

}

stage('Windows Build') {

agent { label 'windows' }

steps {

echo "Building on Windows..."

bat 'echo Windows build running && powershell -Command "Start-Sleep -Seconds 5"'

}

}

}

}

stage('Deploy') {

agent any

steps {

echo "Deploying after parallel builds..."

}

}

}

}

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.