

# Dealing with Data Driven Test Combinatorial Complexity using Jenkins Job DSL

---



# Contents

---

- I Motivation
- II Jenkins Job DSL Plugin Introduction
- III Summary

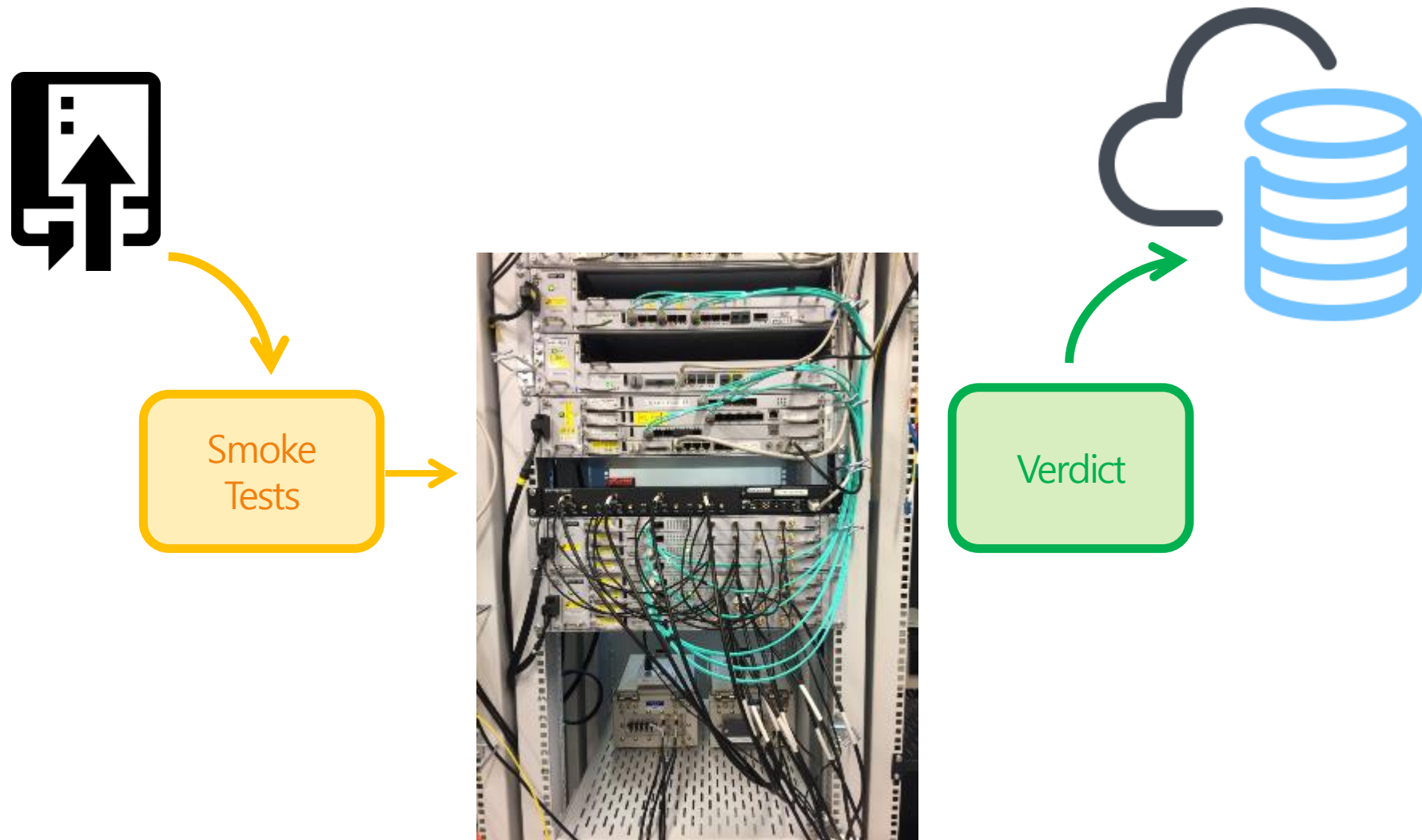




---

# Motivation

# Original Task



# Solution: Create a Jenkins Job

The screenshot shows the Jenkins web interface for a project named "SmokeTestJob". The breadcrumb navigation at the top reads "Jenkins > SmokeTestJob >".

**Project SmokeTestJob**

**Left-hand menu:**

- Status (info icon)
- Changes (clock icon)
- Workspace (folder icon)
- Build with Parameters (play icon)
- Delete Project (trash icon)
- Configure (gear icon)
- Rebuild Last (play icon)
- Job Config History (wrench icon)
- Rename (clock icon)

**Main content area:**

- Workspace:** A folder icon labeled "Workspace".
- Recent Changes:** A clock icon labeled "Recent Changes".
- Permalinks:** A section header.

**Build History:**

- Build History (gear icon) [trend](#) ^
- Search input: find [x]
- RSS for all [RSS icon]
- RSS for failures [RSS icon]

# Let's Extend Our Task a Little



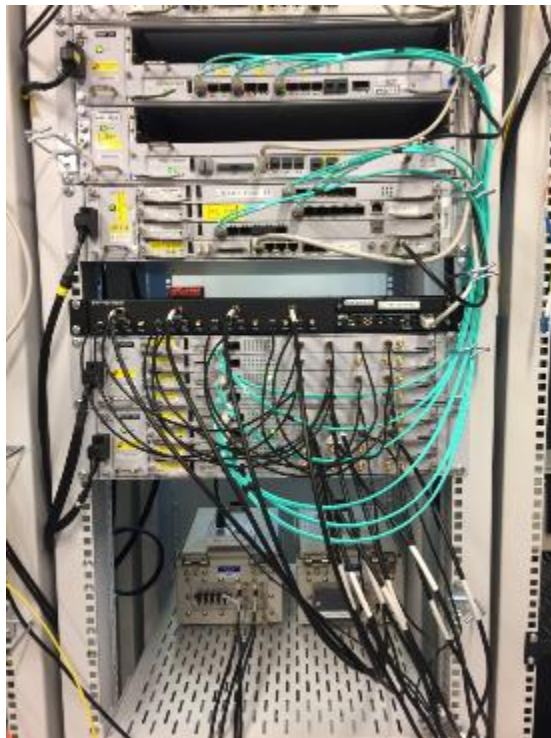
Smoke Tests  
(Conf1)



Smoke Tests  
(Conf2)



Smoke Tests  
(ConfN)



Verdict1



Verdict2



VerdictN



# Naïve Solution 1: Just Copy the Original Job



If you want to create a new item from other existing, you can use this option:



Copy from

SmokeTestJob

S	W	Name ↓	Last Success	Last Failure	Last Duration
		SmokeTestJob	41 sec - #1	N/A	0.25 sec
		SmokeTestJob_P1_AA_P2_11_P3_A1	21 sec - #1	N/A	32 ms
		SmokeTestJob_P1_AA_P2_11_P3_B2	16 sec - #1	N/A	27 ms
		SmokeTestJob_P1_AA_P2_11_P3_C3	12 sec - #1	N/A	27 ms
		SmokeTestJob_P1_BB_P2_22_P3_B2	9 sec - #1	N/A	27 ms
		SmokeTestJob_P1_CC_P2_33_P3_C3	4.8 sec - #1	N/A	29 ms

# What Could Have Possibly Go Wrong?

---



## Preconditions:

1. Each configuration controls 3 parameters
2. Each parameter can take one of the 4 valid values
3. 90% of possible configurations are equally important

**Total configuration number: 60**

Adding a new testbed (different H/W type) doubles the number of jobs



# Solution 1 Drawbacks

---

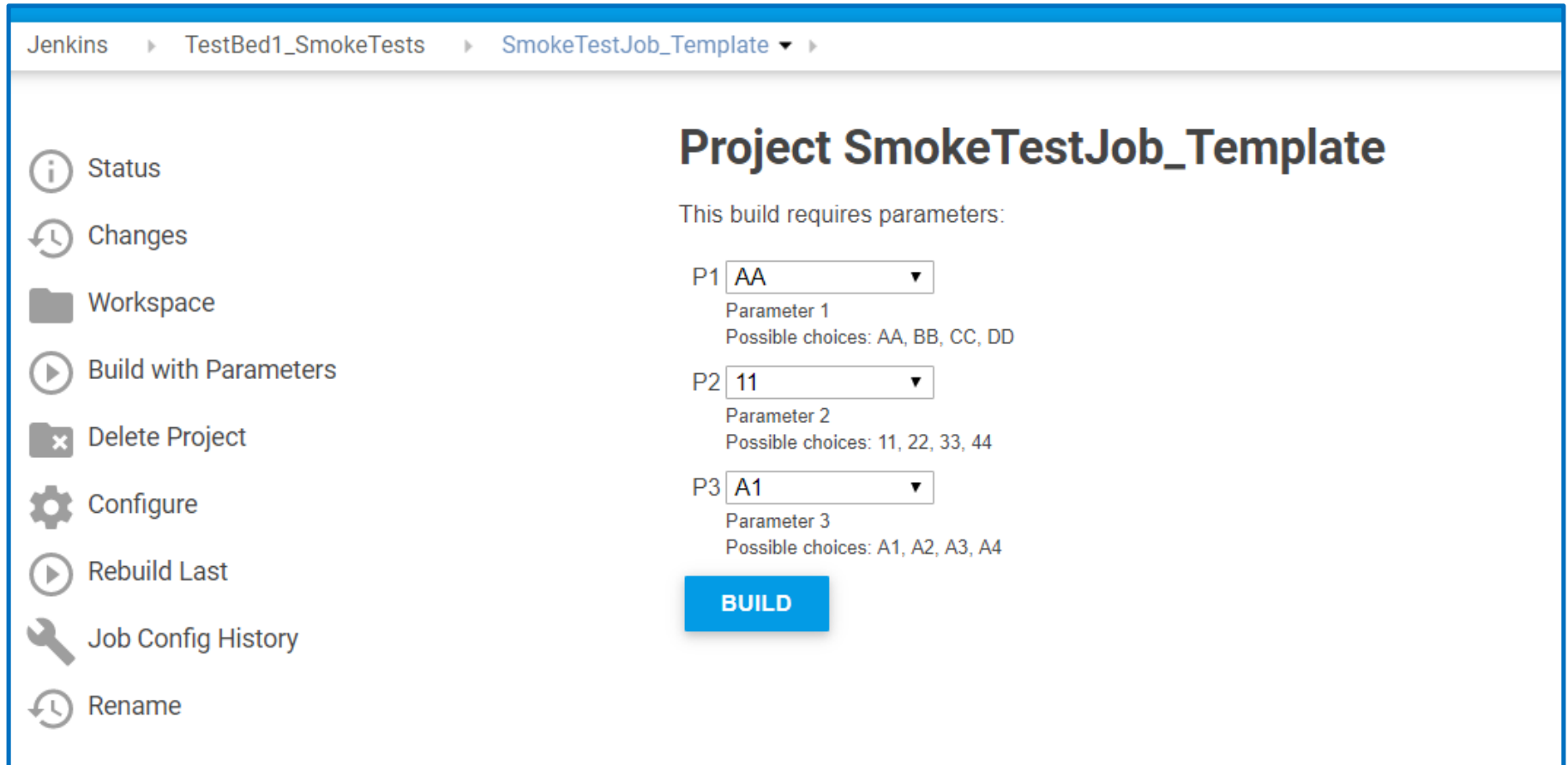


1. New job creation is tedious and error prone
2. Need to change all jobs if additional parameter or execution step is required
3. Huge effort to recover the jobs in case of Jenkins host failure

**Maintenance nightmare!**

# Let's Improve It: MultiJob Step with Parameters

## 1. Create parametrized template job



Jenkins > TestBed1\_SmokeTests > SmokeTestJob\_Template

### Project SmokeTestJob\_Template

This build requires parameters:

P1    
Parameter 1  
Possible choices: AA, BB, CC, DD

P2    
Parameter 2  
Possible choices: 11, 22, 33, 44

P3    
Parameter 3  
Possible choices: A1, A2, A3, A4

**BUILD**

- Status
- Changes
- Workspace
- Build with Parameters
- Delete Project
- Configure
- Rebuild Last
- Job Config History
- Rename

# Let's Improve It: MultiJob Job with Parameters

2. Create MultiJob Project and add each configuration as a separate phase job

The screenshot displays the configuration interface for a MultiJob Phase. The main window is titled "MultiJob Phase" and contains the following sections:

- Phase name:** "Smoke Test For Testbed1"
- Phase jobs:**
  - Job:**
    - Job name:** "SmokeTestJob\_Template"
    - Job alias:** "SmokeTestJob\_Template\_AA\_11\_A1"
    - Kill the phase on:** "Never (ignore the job result and continue the phase)"
    - Build only if SCM changes:**
    - Disable:**
    - Abort all other job:**
    - Aggregated Test Results:**
    - Current job parameters:**
    - Enable retry:**
    - Enable condition:**
  - Predefined parameters:**
    - Parameters:** "P1=AA", "P2=11", "P3=A1"

An "ADVANCED..." button is located at the bottom right of the configuration area.

# MultiJob Project Final Configuration



S	W	Job	Last Success	Last Failure	Last Duration
—	⊗	<b>TestBed1_SmokeTestMultiJob</b>	N/A	N/A	N/A
—	⊗	<i>Smoke Test For Testbed1</i>			
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_A1)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_B2)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_C3)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_D4)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_BB_11_A1)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_BB_22_A1)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_BB_33_A1)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_BB_44_A1)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_CC_11_A1)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_CC_11_B2)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_CC_11_C3)	N/A	N/A	N/A
—	⊗	SmokeTestJob_Template (SmokeTestJob_Template_CC_11_B4)	N/A	N/A	N/A

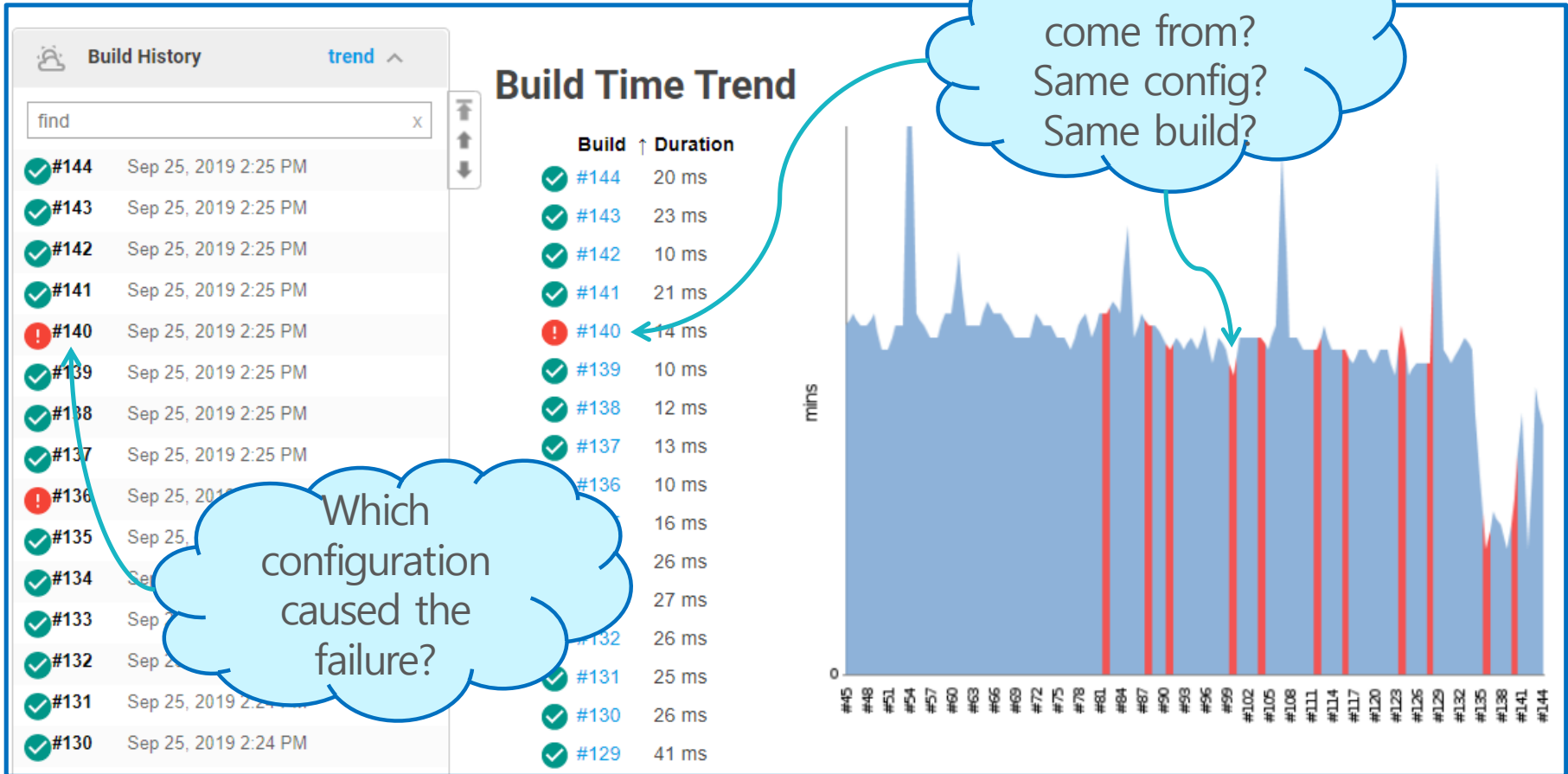
# Are We There Yet?

Let's execute all the jobs

S	W	Job	Last Success	Last Failure	Last Duration	Console	Built On
!	☁	<b>TestBed1_SmokeTestMultiJob</b>	1 day 5 hr	11 sec	3.1 sec		Jenkins
!	⚙	<b>Smoke Test For Testbed1</b>					
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_A1)			36 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_B2)	11 sec	N/A	29 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_C3)	11 sec	N/A	27 ms		Jenkins
!	⚙	SmokeTestJob_Template (SmokeTestJob_Template_AA_11_D4)	1 day 5 hr	11 sec	28 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_BB_11_A1)	11 sec	N/A	28 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_BB_22_A1)	11 sec	N/A	28 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_BB_33_A1)	11 sec	N/A	27 ms		Jenkins
!	⚙	SmokeTestJob_Template (SmokeTestJob_Template_BB_44_A1)	1 day 5 hr	11 sec	26 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_CC_11_A1)	11 sec	N/A	27 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_CC_11_B2)	11 sec	N/A	26 ms		Jenkins
✓	⚙	SmokeTestJob_Template (SmokeTestJob_Template_CC_11_C3)	11 sec	N/A	27 ms		Jenkins

Let's debug this failure

# Hard to Debug, Unreliable Execution Trend



# Hard to Update the Configurations



How to change the configuration AA\_11\_D4?

The screenshot displays a configuration management interface for a 'Smoke Test For Testbed1'. It features four vertically stacked job configuration panels. Each panel is titled 'Job' and contains a 'Job name' input field with the value 'SmokeTestJob\_Template'. To the right of each input field is a question mark icon. Below each input field is a circular refresh icon and a button labeled 'ADVANCED...'. A red square with a white 'X' is positioned in the top right corner of each job panel, indicating a configuration error or a failed update. The overall interface is light gray with white text and buttons.

# Hard to Re-run the Failing Configuration



Jenkins ▾ ▸ SmokeTestJob\_Template ▸

## Project SmokeTestJob\_Template

This build requires parameters:

P1  ▾  
Parameter 1  
Possible choices: AA, BB, CC, DD

P2  ▾  
Parameter 2  
Possible choices: 11, 22, 33, 44

P3  ▾  
Parameter 3  
Possible choices: A1, A2, A3, A4

**BUILD**

What should we select here?

- Status
- Build
- Delete Project
- Configure
- Rebuild Last
- Job Config History
- Rename





# Debugging Nightmare

# Who Could Help Us?

---





---

# Jenkins Job DSL Plugin

# Introducing Jenkins DSL



---

From <https://wiki.jenkins.io>

The job-dsl-plugin allows the programmatic creation of projects using a DSL. Pushing job creation into a script allows you to automate and standardize your Jenkins installation, unlike anything possible before.

# Installation



Jenkins > Plugin Manager

Manage Jenkins  
Update Center

Filter:

Updates Available Installed Advanced

Install	Name	Version
<input checked="" type="checkbox"/>	<b>Job DSL</b> This plugin allows Jobs and Views to be defined via DSLs	1.76
<input type="checkbox"/>	<b>XML Job to Job DSL</b> Use this plugin to convert your jobs into DSL Groovy scripts	0.1.10

**INSTALL WITHOUT RESTART** **DOWNLOAD NOW AND INSTALL AFTER RESTART**  
**CHECK NOW**

# Create First Seed Job



## Enter an item name

» *Required field*



### Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.



### Maven project

Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.



### Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

# Create First Seed Job



## Build

**Process Job DSLs**

See [Job DSL API](#) for syntax reference.

Use the provided DSL script

DSL Script

```
1 job('Jenkins_Job_DSL_Plugin/DSL_First_Job') {
2     steps {
3         shell('echo Hello Jenkins Job DSL')
4     }
5 }
```

Look on Filesystem

Use Groovy Sandbox:

Action for existing jobs and views:  Ignore changes

Action for existing jobs and views managed by another seed job:  Fail if seed collision

Action for removed jobs:

Action for removed views:

# Generate a Job from the Seed Job



Jenkins > Jenkins\_Job\_DSL\_Plugin > Jenkins\_First\_Job\_Seed

## Project Jenkins\_First\_Job\_Seed

Full project name: Jenkins\_Job\_DSL\_Plugin/Jenkins\_First\_Job\_Seed

- Up
- Status
- Changes
- Workspace
- Build Now
- Delete Project
- Configure
- Rebuild Last
- Move
- Job Config History
- Job DSL API Reference
- Rename

**Workspace**

**Recent Changes**

**Generated Items:**

- [Jenkins\\_Job\\_DSL\\_Plugin » DSL\\_First\\_Job](#)

### Permalinks

- Last build (#2), 19 sec ago
- Last stable build (#2), 19 sec ago
- Last successful build (#2), 19 sec ago
- Last failed build (#1), 3 min 28 sec ago
- Last unsuccessful build (#1), 3 min 28 sec ago
- Last completed build (#2), 19 sec ago

**Build History** trend ^

find X

✓ #2 Sep 25, 2019 4:40 PM



# Generate a Job from the Seed Job



The screenshot displays the Jenkins web interface for a job named 'DSL\_First\_Job'. The breadcrumb navigation at the top reads 'Jenkins > Jenkins\_Job\_DSL\_Plugin > DSL\_First\_Job'. The main heading is 'Project DSL\_First\_Job', with the full project name 'Jenkins\_Job\_DSL\_Plugin/DSL\_First\_Job' listed below it. A left-hand sidebar contains navigation options: 'Up', 'Status', 'Changes', 'Workspace', 'Build', 'Delete', 'Config', 'Rebuild', 'Move', 'Job C', and 'Renam'. The 'Build' step configuration is highlighted with a blue border. It features a title 'Build' and a sub-section 'Execute shell' with a red 'X' icon and a help icon. The 'Command' field contains the text 'echo Hello Jenkins Job DSL'. Below the command field is a link: 'See the list of available environment variables'. At the bottom of the configuration area are two buttons: 'ADD BUILD STEP' and 'ADVANCED...'.

# Let's Add Some Parameters



```
1 job('Jenkins_Job_DSL_Plugin/SmokeTestJob_P1_AA_P2_11_P3_A1') {
2   environmentVariables {
3     envs(P2: "11")
4   }
5   parameters {
6     stringParam("P1", "AA", "First parameter, string")
7     textParam("P3", "A1", "Third parameter, text")
8   }
9   steps {
10    shell('echo Parametrized Job: P1=$P2, P2=$P2, P3=$P3')
11  }
12 }
13 job('Jenkins_Job_DSL_Plugin/SmokeTestJob_P1_AA_P2_11_P3_B2') {
14   environmentVariables {
15     envs(P2: "11")
16   }
17   parameters {
18     stringParam("P1", "AA", "First parameter, string")
19     textParam("P3", "B2", "Third parameter, text")
20   }
21   steps {
22     shell('echo Parametrized Job: P1=$P2, P2=$P2, P3=$P3')
23   }
24 }
25
```

# ... and Generate the Jobs

**Project Jenkins\_Parametrized\_Job\_Seed**

Full project name: Jenkins\_Job\_DSL\_Plugin/Jenkins\_Parametrized\_Job\_Seed

Workspace

Recent Changes

Generated Items:

- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_AA\_P2\_11\_P3\_A1
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_AA\_P2\_11\_P3\_B2
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_AA\_P2\_11\_P3\_C3
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_AA\_P2\_11\_P3\_D4
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_BB\_P2\_11\_P3\_A1
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_BB\_P2\_11\_P3\_B2
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_BB\_P2\_11\_P3\_C3
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_BB\_P2\_11\_P3\_D4
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_CC\_P2\_11\_P3\_A1
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_CC\_P2\_11\_P3\_B2
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_CC\_P2\_11\_P3\_C3
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_CC\_P2\_11\_P3\_D4
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_DD\_P2\_11\_P3\_A1
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_DD\_P2\_11\_P3\_B1
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_DD\_P2\_11\_P3\_C3
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_DD\_P2\_11\_P3\_D4

**Build History** trend ^

Build	Timestamp	Status
#3	Sep 26, 2019 9:47 AM	Success
#2	Sep 26, 2019 9:45 AM	Success
#1	Sep 26, 2019 9:42 AM	Failure

# What's Inside?

The image shows a Jenkins configuration page for parameters and a corresponding pipeline script. Annotations connect the UI elements to the code.

**String Parameter Configuration:**

- Name: P1
- Default Value: AA
- Description: First parameter, string
- Options: [Plain text] Preview,  Trim the string

**Multi-line String Parameter Configuration:**

- Name: P3
- Default Value: B2
- Description: Third parameter, text
- Options: [Plain text] Preview

**Properties Content:** P2=11

**Pipeline Code:**

```
20 job('Jenkins_Job_PSL_Plugin/SmokeTestJob_P1_AA_P2_11_P3_B2') {
21   environmentVariables {
22     envs(P2: "11")
23   }
24   parameters {
25     stringParam("P1", "AA", "First parameter, string")
26     textParam("P3", "B2", "Third parameter, text")
27   }
28   steps {
29     shell('echo Parametrized Job: P1=$P2, P2=$P2, P3=$P3')
30   }
31 }
```

Annotations: A blue arrow points from the 'Default Value' field of the String Parameter to the 'stringParam' line in the code. A green arrow points from the 'Default Value' field of the Multi-line String Parameter to the 'textParam' line in the code. An orange arrow points from the 'envs(P2: "11")' line in the code to the 'Properties Content' box.

# Now DRY It a Little...



```
1 job('Jenkins_Job_DSL_Plugin/SmokeTestJob_P1_AA_P2_11_P3_A1') {
2   environmentVariables {
3     envs(P2: "11")
4   }
5   parameters {
6     stringParam("P1", "AA", "First parameter, string")
7     textParam("P3", "A1", "Third parameter, text")
8   }
9   steps {
10    shell('echo Parametrized Job: P1=$P2, P2=$P2, P3=$P3')
11  }
12 }
13 job('Jenkins_Job_DSL_Plugin/SmokeTestJob_P1_AA_P2_11_P3_B2') {
14   environmentVariables {
15     envs(P2: "11")
16   }
17   parameters {
18     stringParam("P1", "AA", "First parameter, string")
19     textParam("P3", "B2", "Third parameter, text")
20   }
21   steps {
22    shell('echo Parametrized Job: P1=$P2, P2=$P2, P3=$P3')
23  }
24 }
25
```

Can we improve it?

# Use a Template Job

```
1 def script_steps = '''echo Parametrized Job: P1=$P2, P2=$P2, P3=$P3
2 pip install -r requirements.txt
3 python --version
4 '''
5
6 job('Jenkins_Job_DSL_Plugin/SmokeTestJob_Template') {
7     scm {
8         git {
9             remote {
10                url("https://github.com/olekkravchenko/mbt_robot_listener")
11                branch("*/master")
12            }
13        }
14    }
15    steps {
16        virtualenv {
17            pythonName('Python2.7')
18            clear(false)
19            systemSitePackages(true)
20            command(script_steps)
21        }
22    }
23 }
24 job('Jenkins_Job_DSL_Plugin/SmokeTestJob_P1_AA_P2_11_P3_A1') {
25     using("Jenkins_Job_DSL_Plugin/SmokeTestJob_Template")
26     environmentVariables {
27         envs(P2: "11")
28     }
29     parameters {
30         stringParam("P1", "AA", "First parameter, string")
31         textParam("P3", "A1", "Third parameter, text")
32     }
33 }
34
35
```

# Build Results



## ✓ Console Output

```
Started by user srpol  
Running as SYSTEM  
[EnvInject] - Loading node environment variables.
```

Jenkins » Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_P1\_AA\_P2\_11\_P3\_A1

### Project SmokeTestJob\_P1\_AA\_P2\_11\_P3\_A1

Full project name: Jenkins\_Job\_DSL\_Plugin/SmokeTestJob\_P1\_AA\_P2\_11\_P3\_A1

- Up
- Status
- Changes
- Workspace
- Build with Parameters
- Delete Project
- Configure
- Rebuild Last
- Job Config History
- Rename

- Workspace
- Recent Changes
- Seed job: Jenkins\_Job\_DSL\_Plugin » Jenkins\_Parametrized\_Job\_Seed
- Template job: Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_Template

# What's Generated?

The screenshot shows the Jenkins configuration page for a job named 'SmokeTestJob\_P1\_AA\_P2\_11\_P3\_A1'. The 'Source Code Management' tab is active, showing 'Git' selected as the SCM. The repository URL is 'https://github.com/olekkravchenko/mbt\_robot\_listener' and the branch is specified as '\*/master'. The 'Build' section shows a 'Virtualenv Builder' step configured with 'Python2.7' as the Python version, 'Clear' unchecked, 'Nature' set to 'Shell', and a command to echo job parameters, install requirements, and check the Python version.

```
6 job('Jenkins_Job_DS_Plugin/SmokeTestJob_Template') {
7   scm {
8     git {
9       remote {
10        url("https://github.com/olekkravchenko/mbt_robot_listener")
11        branch("*/master")
12      }
13    }
14  }
15  steps {
16    virtualenv {
17      pythonName('Python2.7')
18      clear(false)
19      systemSitePackages(true)
20      command(script_steps)
21    }
22  }
23 }
```



# Bringing It All Together...



```
1 def parametrized_jobs_folder = 'Jenkins_Job_DSL_Plugin'
2 def parametrized_jobs_list = [
3   'SmokeTestJob_P1_AA_P2_11_P3_A1',
4   'SmokeTestJob_P1_AA_P2_11_P3_B2',
5   'SmokeTestJob_P1_AA_P2_11_P3_C3',
6   'SmokeTestJob_P1_AA_P2_11_P3_D4',
7   'SmokeTestJob_P1_BB_P2_11_P3_A1',
8   'SmokeTestJob_P1_BB_P2_11_P3_B2',
9   'SmokeTestJob_P1_BB_P2_11_P3_C3',
10  'SmokeTestJob_P1_BB_P2_11_P3_D4',
11 ]
12 multiJob("${parametrized_jobs_folder}/SmokeTestJob_Container"){
13   parameters{
14     stringParam("MULTIJOB_PARAM1", "MJP1", "Demo parameter for MultiJob")
15   }
16   steps {
17     phase("Run all smoke test configurations"){
18       parametrized_jobs_list.each{
19         phaseJob("${it}") {
20           killPhaseCondition('NEVER')
21           abortAllJobs(false)
22           parameters{
23             predefinedProp('PHASE_JOB_PARAM1', 'ABC')
24           }
25         }
26       }
27     }
28     executionType('SEQUENTIALLY')
29     continuationCondition('ALWAYS')
30   }
31 }
```

Groovy array

Groovy loop

# And the Amazing Result Is...



S	W	Job	Last Success
—	⊗	SmokeTestJob_Container	N/A
—	⊗	<i>Run all smoke test configurations</i>	
—	⊗	SmokeTestJob_P1_AA_P2_11_P3_A1	N/A
—	⊗	SmokeTestJob_P1_AA_P2_11_P3_B2	N/A
—	⊗	SmokeTestJob_P1_AA_P2_11_P3_C3	N/A
—	⊗	SmokeTestJob_P1_AA_P2_11_P3_D4	N/A
—	⊗	SmokeTestJob_P1_BB_P2_11_P3_A1	N/A
—	⊗	SmokeTestJob_P1_BB_P2_11_P3_B2	N/A
—	⊗	SmokeTestJob_P1_BB_P2_11_P3_C3	N/A
—	⊗	SmokeTestJob_P1_BB_P2_11_P3_D4	N/A
—	⊗	SmokeTestJob_P1_CC_P2_11_P3_A1	N/A
—	⊗	SmokeTestJob_P1_CC_P2_11_P3_B2	N/A
—	⊗	SmokeTestJob_P1_CC_P2_11_P3_C3	N/A
—	⊗	SmokeTestJob_P1_CC_P2_11_P3_D4	N/A
—	⊗	SmokeTestJob_P1_DD_P2_11_P3_A1	N/A
—	⊗	SmokeTestJob_P1_DD_P2_11_P3_C3	N/A
—	⊗	SmokeTestJob_P1_DD_P2_11_P3_D4	N/A

# Most Important Parts in MultiJob Seed



**Build**

**MultiJob Phase**

Phase name: Run all smoke test configurations

Phase jobs

**Job**

Job name: SmokeTestJob\_P1\_AA\_P2\_11\_P3\_A1

Job alias:

Kill the phase on: Never (ignore the job result and continue th

Build only if SCM changes

Disable

Abort all other job

Aggregated Test Results

Current job parameters

Enable retry

Enable condition

**Predefined parameters**

ADD JOBS...

```
12 multiJob("${parametrized_jobs_folder}/SmokeTestJob_Container"){
13   parameters{
14     stringParam("MULTIJOB_PARAM1", "MJP1", "Demo parameter for MultiJob")
15   }
16   steps {
17     phase("Run all smoke test configurations"){
18       parametrized_jobs_list.each{
19         phaseJob("${it}") {
20           killPhaseCondition('NEVER')
21           abortAllJobs(false)
22           parameters{
23             predefinedProp('PHASE_JOB_PARAM1', 'ABC')
24           }
25         }
26       }
27     executionType('SEQUENTIALLY')
28     continuationCondition('ALWAYS')
29   }
30 }
31 }
```

Job execution type: Running phase jobs sequentially

Continuation condition to next phase when jobs' statuses are: Always

# What's Missing?

## Trigger



```
20 multiJob("${parametrized_jobs_folder}/SmokeTestJob_Container"){
21   triggers {
22     folderContentTrigger {
23       cronTabSpec("H/2 * * * *")
24       path('${MAIL_INBOX}')
25       includes("**/*.msg")
26       excludes("dump/mail/")
27       excludeCheckLastModificationDate(false)
28       excludeCheckContent(false)
29       excludeCheckFewerOrMoreFiles(false)
30     }
31   }
```

[FSTrigger] - Monitor folder

Path

Includes   
Fileset includes setting that specifies the includes files, such as \*\*/\*.xml,\*\*/\*.c. Fileset patterns are comma separated. Basedir of the fileset is relative to the workspace root. If no value is set, all files are used.

Excludes

Exclude check lastModification date

Exclude check content

Exclude check fewer or more files

Schedule

Would last have run at Thursday, September 26, 2019 2:58:51 PM UTC; would next run at Thursday, September 26, 2019 2:58:51 PM UTC.

# What's Missing?

## Result Publisher

```
50 publishers {
51   archiveJUnit {
52     junit {
53       pattern('smoke_test_result_combined.xml')
54     }
55   }
56   failedThresholds {
57     unstable(10)
58     unstableNew(20)
59     failure(30)
60     failureNew(40)
61   }
62   skippedThresholds {
63     unstable(5)
64     unstableNew(10)
65     failure(15)
66     failureNew(15)
67   }
68   thresholdMode(ThresholdMode.PERCENT)
69 }
```

### Post-build Actions

#### Publish xUnit test result report

Report Type: **JUnit**

Pattern:

[See the list of available environment variables](#)

- Skip if there are no test files
- Fail the build if test results were not updated this run
- Delete temporary JUnit files
- Stop and set the build status to failed if there are errors when processing a result file

ADD ▾

#### Thresholds

##### Failed Tests

Build Status	Total	New	Total	New
Thresholds:	<input type="text" value="10"/>	<input type="text" value="20"/>	<input type="text" value="30"/>	<input type="text" value="40"/>

Configure the build status. A build is considered as unstable or failure if the new or total number of failed tests exceeds the specified thresholds.

##### Skipped Tests

Build Status	Total	New	Total	New
Thresholds:	<input type="text" value="5"/>	<input type="text" value="10"/>	<input type="text" value="15"/>	<input type="text" value="15"/>

# It's a Little Bit Messy



Jenkins\_Job\_DSL\_Plugin

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration
⊖	⚙️	DSL_First_Job	N/A	N/A	N/A
✅	☁️	Jenkins_First_Job_Seed	4 days 22 hr - #2	4 days 22 hr - #1	1.7 sec
✅	⚙️	Jenkins_Parametrized_Job_Seed	4 days 3 hr - #21	4 days 3 hr - #12	0.86 sec
✅	📁	Jenkins_Parametrized_MultiJob_Seed	3 days 23 hr - #12		
✅	⚙️	SmokeTestJob_Container	2 min 43 sec - #1		
✅	⚙️	SmokeTestJob_P1_AA_P2_11_P3_A1	2 min 42 sec - #1		
✅	⚙️	SmokeTestJob_P1_AA_P2_11_P3_B2	2 min 30 sec - #3		
✅	⚙️	SmokeTestJob_P1_AA_P2_11_P3_C3	2 min 20 sec - #1	N/A	9.2 sec
✅	⚙️	SmokeTestJob_P1_AA_P2_11_P3_D4	2 min 10 sec - #1	N/A	10 sec
✅	⚙️	SmokeTestJob_P1_BB_P2_11_P3_A1	1 min 57 sec - #1	N/A	7.6 sec
✅	⚙️	SmokeTestJob_P1_BB_P2_11_P3_B2	1 min 47 sec - #1	N/A	9 sec
✅	⚙️	SmokeTestJob_P1_BB_P2_11_P3_C3	1 min 37 sec - #1	N/A	8.5 sec
✅	⚙️	SmokeTestJob_P1_BB_P2_11_P3_D4	1 min 27 sec - #1	N/A	8.9 sec

Does it even belong together?

# Let's Organize It Better

## Create a Separate Folder

```
1 folder('SmokeTestJobs_Testbed1') {  
2     displayName('Smoke test jobs for Testbed1')  
3     description('Folder that contains all jobs executed on Testbed1')  
4 }  
5
```

The screenshot shows the Jenkins web interface. The breadcrumb path is 'Jenkins > Smoke test jobs for Testbed1'. On the left sidebar, there are navigation options: Up, Status, Configure, New Item, Delete Folder, People, Build History, Move, and Job Config History. The main content area displays a folder icon and the name 'Smoke test jobs for Testbed1'. Below the name, it shows 'Folder name: SmokeTestJobs\_Testbed1' and 'Folder that contains all jobs executed on Testbed1'. There is a '+ All' button. At the bottom, there is a 'Welcome to Jenkins!' message and a light blue box with the text 'Please create new jobs to get started.' Two arrows originate from the code block above: a green arrow points from the folder name in the code to the folder name in the UI, and an orange arrow points from the description in the code to the description in the UI.













# Let's Organize It Better

## Create a Separate View

```
1 def parametrized_jobs_folder = 'Jenkins_Job_DSL_Plugin'
2 def parametrized_jobs_list = [
3   'SmokeTestJob_P1_AA_P2_11_P3_A1',
4   'SmokeTestJob_P1_AA_P2_11_P3_B2',
5   'SmokeTestJob_P1_AA_P2_11_P3_C3',
6   'SmokeTestJob_P1_AA_P2_11_P3_D4',
7 ]
8 listView("${parametrized_jobs_folder}/SmokeTestJobsView") {
9   description('All generated jobs for Smoke Tests')
10  filterBuildQueue()
11  filterExecutors()
12  jobs {
13    parametrized_jobs_list.each {
14      name("${it}")
15    }
16  }
17  jobFilters {
18    status {
19      status(Status.STABLE)
20    }
21  }
22  columns {
23    status()
24    weather()
25    name()
26    lastSuccess()
27    lastFailure()
28    lastDuration()
29    buildButton()
30  }
31 }
```

All generated jobs for Smoke Tests

 edit description

S	W	Name ↓	Last Success	Last Failure	Last Duration	
		<a href="#">SmokeTestJob_P1_AA_P2_11_P3_A1</a>	1 day 18 hr - #8	N/A	1.8 sec	
		<a href="#">SmokeTestJob_P1_AA_P2_11_P3_B2</a>	1 day 18 hr - #10	N/A	1.8 sec	
		<a href="#">SmokeTestJob_P1_AA_P2_11_P3_C3</a>	1 day 18 hr - #8	N/A	1.8 sec	
		<a href="#">SmokeTestJob_P1_AA_P2_11_P3_D4</a>	1 day 18 hr - #8	N/A	1.8 sec	
		<a href="#">SmokeTestJob_P1_BB_P2_11_P3_A1</a>	1 day 18 hr - #8	N/A	1.8 sec	
		<a href="#">SmokeTestJob_P1_BB_P2_11_P3_B2</a>	1 day 18 hr - #8	N/A	1.7 sec	



# What Did We Achieve?

## Can isolate particular configuration failure

The screenshot displays the Jenkins web interface. On the left, a table lists jobs with their status, icons, and last success/failure times. The job 'SmokeTestJob\_Container' is highlighted with a red exclamation mark icon, indicating a failure. An orange arrow points from this job to the detailed view of 'Project SmokeTestJob\_P1\_BB\_P2\_11\_P3\_A1' on the right. This view includes a sidebar with navigation options like 'Status', 'Changes', 'Workspace', and 'Build History'. The 'Build History' section shows a list of builds, with build #9 marked as failed (red exclamation mark) and build #8 as successful (green checkmark). The main content area shows project details, including workspace, recent changes, and upstream projects.

S	W	Job	Last Success	Last Failure
!	☁	SmokeTestJob_Container	1 day 19 hr	2 min 51 sec
-	⚙	Run all smoke test configurations		
✓	⚙	SmokeTestJob_P1_AA_P2_11_P3_A1	2 min 50 sec	N/A
✓	⚙	SmokeTestJob_P1_BB_P2_11_P3_A1		
✓	⚙	SmokeTestJob_P1_CC_P2_11_P3_A1		
✓	⚙	SmokeTestJob_P1_DD_P2_11_P3_A1		
✓	⚙	SmokeTestJob_P1_EE_P2_11_P3_A1		
!	🔗	SmokeTestJob_P1_FF_P2_11_P3_A1		
✓	⚙	SmokeTestJob_P1_GG_P2_11_P3_A1		
✓	⚙	SmokeTestJob_P1_HH_P2_11_P3_A1		

**Project SmokeTestJob\_P1\_BB\_P2\_11\_P3\_A1**  
Full project name: Jenkins\_Job\_DSL\_Plugin/SmokeTestJob\_P1\_BB\_P2\_11\_P3\_A1

**Build History**

Build	Time	Status
#9	Oct 2, 2019 10:48 AM	Failed
#8	Sep 30, 2019 3:48 PM	Successful
#7	Sep 30, 2019 3:43 PM	Successful

**Upstream Projects**

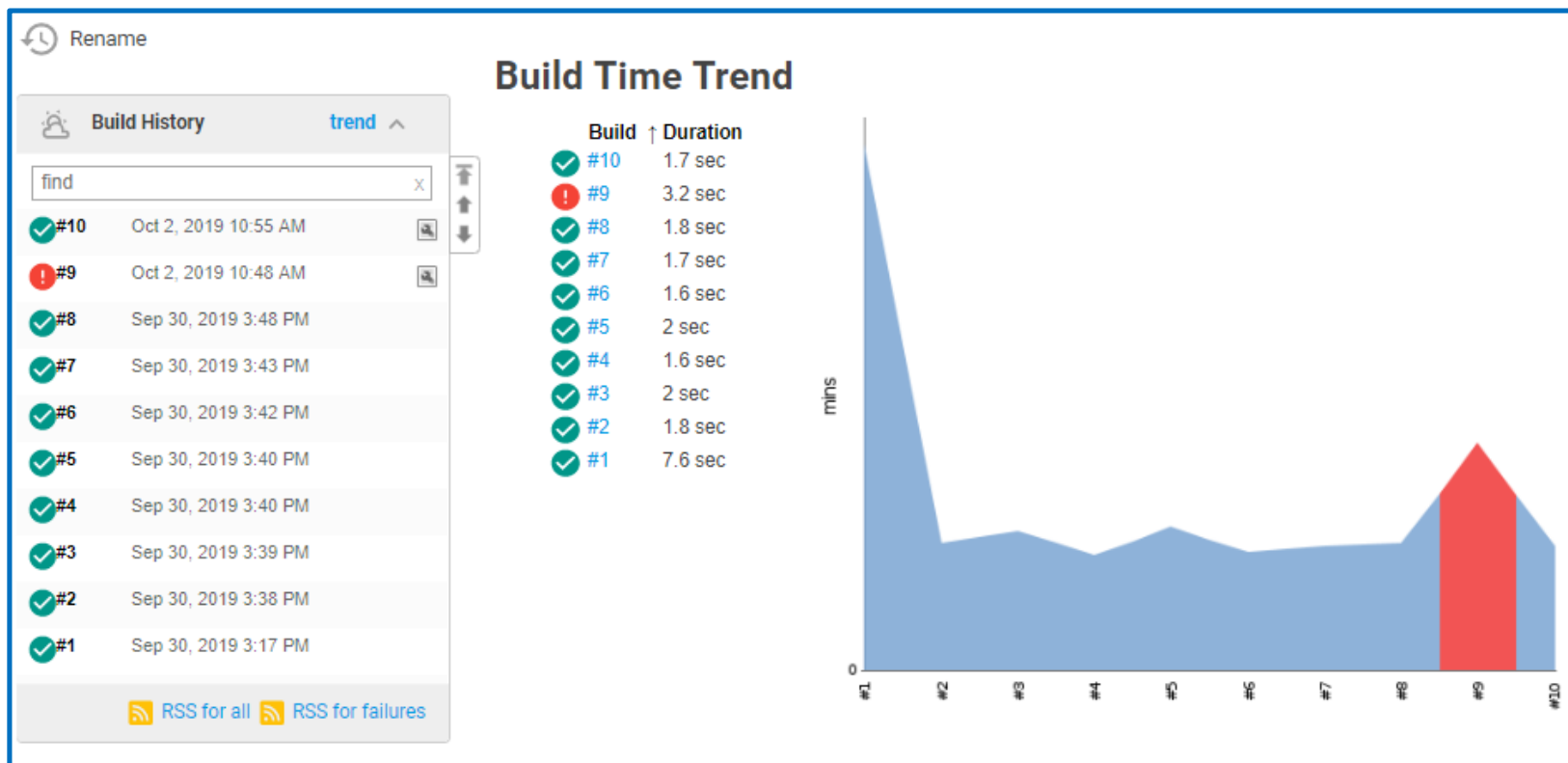
- Jenkins\_Job\_DSL\_Plugin » SmokeTestJob\_Container

**Permalinks**

- Last build (#9), 5 min 14 sec ago
- Last stable build (#8), 1 day 19 hr ago
- Last successful build (#8), 1 day 19 hr ago
- Last failed build (#9), 5 min 14 sec ago
- Last unsuccessful build (#9), 5 min 14 sec ago

# What Did We Achieve?

Execution result history is separate for each job



# What Did We Achieve?

Can easily modify jobs steps or configuration at once or add a new parametrized job

```
1 def parametrized_jobs_folder = 'Jenkins_Job_DSL_Plugin'
2 def script_steps = '''echo Parametrized Job: P1=$P2, P2=$P2, P3=$P3, P4=$P4
3 pip install -r requirements.txt
4 python --version
5 '''
6
7 job("${parametrized_jobs_folder}/SmokeTestJob_Template") {
8   scm {
9     git {
10      remote {
11        url("https://github.com/olekkravchenko/mbt_robot_listener")
12        branch("*/master")
13      }
14    }
15  }
16  steps {
17    virtualenv {
18      pythonName('Python3.7')
19      clear(false)
20      systemSitePackages(true)
21      command(script_steps)
22    }
23  }
24 }
25
```

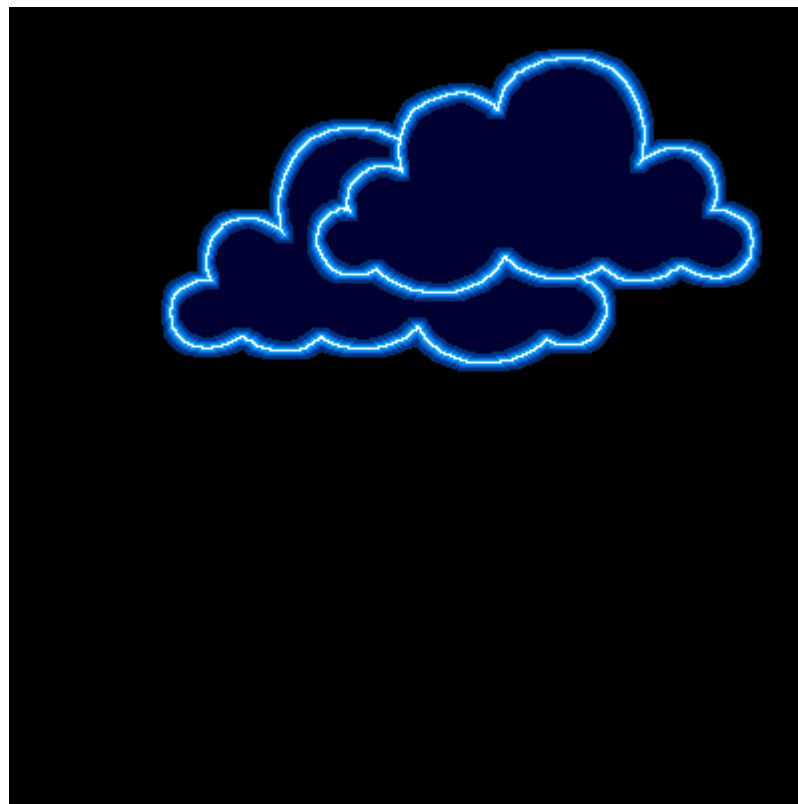
**Note:** Job's execution history is preserved after job is regenerated

# What Did We Achieve?

---



If the lightning strikes our Jenkins host we can re-create all our jobs easily





# Summary

# Advantages

---



1. First and foremost: Configuration as Code
  - a) Can leverage a Version Control system
  - b) It is easier and faster to update the jobs
  - c) It is easier to track and fix job configuration errors
  
2. Probably the easiest way to control multiple related jobs
  - a) Can keep jobs independent and self-sufficient
  - b) Apply complex modifications in one go
  
3. Much more pleasant and fun than just GUI clicking

# Drawbacks

---



Honestly, I don't see any 😊

Except of some enigmatic Groovy error messages

```
[EnvInject] - Loading node environment variables.  
Building in workspace /var/jenkins_home/workspace/Jenkins_Job_DSL_Plugin/Jenkins_Parametrized_MultiJob_Seed  
Processing provided DSL script  
ERROR: (script, line 12) No signature of method: script.multijob() is applicable for argument types:  
(org.codehaus.groovy.runtime.GStringImpl, script$_run_closure1) values:  
[Jenkins_Job_DSL_Plugin/SmokeTestJob_Container, script$_run_closure1@ddbc1cf]  
Possible solutions: multiJob(java.lang.String), multiJob(java.lang.String, groovy.lang.Closure)  
Finished: FAILURE
```

# It May Help You a Little (or a Lot)

## Jenkins Job DSL Documentation (Installed with plugin)

**SAMSUNG** 2

Jenkins > Jenkins\_Job\_DSL\_Plugin > SmokeTestJob\_P1\_AA\_P2\_11\_P3\_B2

← Up

- Status
- Changes
- Workspace
- Build with Parameters
- Delete Project
- Configure
- Rebuild Last
- Job Config History
- Rename

### Project SmokeTestJob\_P1\_AA\_P2\_11\_P3\_B2

Full project name: Jenkins\_Job\_DSL\_Plugin/SmokeTestJob\_P1\_AA\_P2\_11\_P3\_B2

- Workspace
- Recent Changes
- Seed job: [Jenkins\\_Job\\_DSL\\_Plugin » Jenkins\\_Parametrized\\_Job\\_Seed](#)
- Template job: [Jenkins\\_Job\\_DSL\\_Plugin » SmokeTestJob\\_Template](#)

#### Build History

trend ^

✓ #2	Sep 26, 2019 10:09 AM	
✓ #1	Sep 26, 2019 9:51 AM	

RSS for all RSS for failures

#### Upstream Projects

- ⊖ [Jenkins\\_Job\\_DSL\\_Plugin » SmokeTestJob\\_Container](#)

#### Permalinks

- [Last build \(#2\), 6 hr 25 min ago](#)
- [Last stable build \(#2\), 6 hr 25 min ago](#)
- [Last successful build \(#2\), 6 hr 25 min ago](#)
- [Last completed build \(#2\), 6 hr 25 min ago](#)



# It May Help You a Little (or a Lot)

Jenkins Job DSL Playground ([job-dsl.herokuapp.com](http://job-dsl.herokuapp.com))

Jenkins Job DSL Playground

Docs

▶ Run

Groovy

XML

```
1 def parametrized_jobs_folder = 'Jenkins_Job_DSL_Plugin'
2 def parametrized_jobs_list = [
3   'SmokeTestJob_P1_AA_P2_11_P3_A1',
4   'SmokeTestJob_P1_AA_P2_11_P3_B2',
5   'SmokeTestJob_P1_AA_P2_11_P3_C3',
6   'SmokeTestJob_P1_AA_P2_11_P3_D4',
7   'SmokeTestJob_P1_BB_P2_11_P3_A1',
8   'SmokeTestJob_P1_BB_P2_11_P3_B2',
9   'SmokeTestJob_P1_BB_P2_11_P3_C3',
10  'SmokeTestJob_P1_BB_P2_11_P3_D4',
11 ]
12 multiJob("${parametrized_jobs_folder}/SmokeTestJob_Container"){
13   parameters{
14     stringParam("MULTIJOB_PARAM1", "MJP1", "Demo parameter for MultiJob")
15   }
16   steps {
17     phase("Run all smoke test configurations"){
18       parametrized_jobs_list.each{
19         phaseJob("${it}") {
20           killPhaseCondition('NEVER')
21           abortAllJobs(false)
22           parameters{
23             stringParam('PHASE_JOB_PARAM1', 'ABC')
24           }
25         }
26       }
27       executionType('SEQUENTIALLY')
28       continuationCondition('ALWAYS')
29     }
30   }
31 }
```

1



---

# Q&A

Thank You

SAMSUNG