[Skip to end of metadata](https://wiki.ercot.com/display/CIA/Jenkins+Job+Configuration" \l "page-metadata-end)

* Created by [Johnson, Brandon](https://wiki.ercot.com/display/~bjohnson), last modified on [Sep 22, 2020](https://wiki.ercot.com/pages/diffpagesbyversion.action?pageId=133108100&selectedPageVersions=1&selectedPageVersions=2)

[Go to start of metadata](https://wiki.ercot.com/display/CIA/Jenkins+Job+Configuration#page-metadata-start)

Overview

The Jenkins instances maintained by the IT Tools Development group are configured using the JCasC plugin. The "configuration as code" approach ensures that in the in the event of an incident the group can recover quickly and with minimal disruption to customer teams. That means all core configuration for the Jenkins instances is stored in source control. The IT Tools team strongly recommends that all other teams follow the same practice. Do not configure jobs manually though Jenkins' UI. Such configuration is not easily repeatable and may be lost in the case of a destructive incident.

Job Configuration

Jenkins has a number of features that enable the programmatic configuration and maintenance of jobs but the most important tools are pipelines, the Job-DSL, and the JCasC. These three features allow developers to configure the overwhelming majority of their needs from Jenkins programmatically. If a product changes teams migrating the Jenkins job configuration is a relatively small task. If a patch breaks a Jenkins instance recovering into a clean instance is a relatively small task. If someone wants to set up an identical version of a job with minor changes it's a relatively small task. Essentially committing to configuring jobs programmatically makes distributing, modifying, copying, and restoring job configuration a relatively low effort activity compared to the high effort option of managing configuration through Jenkins' UI.

Job DSL Example

Every Jenkins instance has a corresponding configuration repository that is merged with the core configuration repository at instance deployment time. The instance specific repository grants development teams the ability to override inherited settings (i.e. adding new plugins) or to add new settings or functionality (i.e. describing jobs using the Job DSL).

jenkins:

clusterZone: "apps.devk8s.ercot.com"

master:

ingress:

hostName: "jenkins.<TEAM>.apps.devk8s.ercot.com"

tls:

- hosts:

- "jenkins.<TEAM>.apps.devk8s.ercot.com"

JCasC:

configScripts:

jobs:

- script: >

organizationFolder('<BITBUCKET\_PROJECT\_KEY>') {

description("<PROJECT\_DESCRIPTION>)

displayName('<PROJECT\_DISPLAY\_NAME>')

organizations {

bitbucket {

serverUrl("https://stash.ercot.com")

repoOwner("<BITBUCKET\_PROJECT\_KEY>")

credentialsId("JENKINS\_SERVICE\_ACCOUNT")

traits {

webhookRegistrationTrait {

mode('ITEM')

}

bitbucketBranchDiscovery {

strategyId(1)

}

bitbucketPullRequestDiscovery {

strategyId(1)

}

bitbucketSshCheckout {

credentialsId("JENKINS\_SERVICE\_ACCOUNT\_SSH")

}

}

}

}

projectFactories {

workflowMultiBranchProjectFactory {

scriptPath 'Jenkinsfile'

}

}

orphanedItemStrategy {

discardOldItems {

daysToKeep(-1)

numToKeep(-1)

}

}

configure { node ->

node / triggers / 'com.cloudbees.hudson.plugins.folder.computed.PeriodicFolderTrigger' {

spec('H H \* \* \*')

interval(86400000)

}

}

}

- script: queue('<BITBUCKET\_PROJECT\_KEY>')

When merged with the core Jenkins configuration supplied in jcasc-configuration and deployed to the Jenkins instance for <TEAM> the example above will cause Jenkins to create folder for the Bitbucket Project where Jenkins will monitor for the creation of any new repo or branch with a Jenkinsfile, automatically create a Jenkins job for that repo/branch, and set up the webhooks that notify Jenkins whenever something in that repo/branch has changed. The two script elements in the jobs array in the above YAML may be replicated as many times as is necessary to configure Jobs and monitoring for as many projects as a team requires.

Deployment

In the long term the deployment process will likely be automated as any other CI/CD process. In the shorter term if/when you've made changes to your respective jcasc-configuration-<TEAM> repo and would like to have those changes deployed to the managed instance of Jenkins please send an email to [deliverysupportsystems@ercot.com](mailto:deliverysupportsystems@ercot.com) and someone from the DSS team will work with you to get your changes deployed.