Jenkins Documentation

Job Creation

An excellent article on the process of creating new Bitbucket Team/Project type Jenkins jobs can be found [here](https://support.cloudbees.com/hc/en-us/articles/115000053051-How-to-Trigger-Multibranch-Jobs-from-Bitbucket-Server-). Please read the relevant section for our BitBucket server version prior to continuing here.

At a high level the process works as follows:

1. Login to the Jenkins instance. Note: The instance found at [http://jenkins.devokd.ercot.com](http://jenkins.devokd.ercot.com/) currently does not require users to login. At some point in the future that will change and you will need to use your standard ERCOT credentials.
2. Click the New Item button on the left navigation pane. You will be presented with a list of options for the new item you're creating and asked to give it a name. Select "Bitbucket Team/Project" and provide a name. We recommend using your product's name or the name of your Bitbucket project. Click ok.
3. You will see a configuration screen. You will ***only*** need to configure the credentials, owner, scan organization folder triggers, and scan child triggers fields. You will need to scroll to find them all. After you've configured those elements click save. You will be taken to a screen showing Jenkins scanning your project. Jenkins will create folders for all repos that meet the configured criteria and within those folders jobs for each branch that has a Jenkinsfile at its root.
   1. Bitbucket Team/Project / Credentials - Set the field to the credential with the description "Jenkins Username/Password Bitbucket Login". These are the credentials Jenkins will use to communicate with Bitbucket.
   2. Bitbucket Team/Project / Owner - Set the field to the project key for your Bitbucket project. The owner is the project Jenkins will monitor in Bitbucket.
   3. Scan Organization Folder Triggers / Interval - Set the interval field to "1 Hour". The scan interval is the frequency at which Jenkins will re-scan your Bitbucket project to check for new repositories. You may scan less frequently if you like.
   4. Scan Child Triggers / Interval - Set the interval field to "30 Minutes". The scan interval is the frequency at which Jenkins will re-scan your repos to check for new branches. You may scan less frequently if you like.

Shared Pipeline Library

In the interest of adhering to the DRY (Don't repeat yourself) principle ERCOT is building a set of libraries for accomplishing common tasks using Jenkins. These Global Variables all leverage the Jenkins Kubernetes Plugin's ability to spawn agents dynamically to enable scalability, build isolation, improving build environment consistency, as well as offering significant reductions to the effort required to maintain build environments. If your team has a process that is not currently represented or you believe you have some process that might be useful to other teams please feel free to submit a PR with an implementation or to contact the Delivery Support Services ([jdischer@ercot.com](mailto:jdischer@ercot.com) or [bjohnson@ercot.com](mailto:bjohnson@ercot.com)) or Common Platforms ([mackermann@ercot.com](mailto:mackermann@ercot.com)) for assistance with integration. Note that although we welcome contributions you are under no obligation to do so. We only ask that if you create your own pipelines that they be stored in source control in a similar manner.

ContainerImage Global Variable

The ContainerImage pipeline takes a git repository with a Dockerfile file at its base, builds the Dockerfile, then pushes the resulting container image into the designated container image registry. The resulting image will be named using the pattern <registry>/ercot/<product><module>:<tags>.

A Typical Jenkinsfile:

ContainerImage {

product = '<name-of-ercot-product>'

module = '<name-of-product-component>'

}

Configuration:

| **Name** | **Type** | **Required** | **Description** |
| --- | --- | --- | --- |
| cloud | String | FALSE | Identifier for the "cloud" Jenkins should use to provision dynamic agents. Defaults to "kubernetes". |
| buildImage | String | FALSE | Identifier for the image that will serve as the container build agent. Defaults to "[dev-registry.ercot.com/ercot/jenkins/buildah-agent:latest](https://stash.ercot.com/projects/CICD/repos/jenkins-buildah-agent/browse)" |
| acctId | String | FALSE | Identifier for the Jenkins Credential used to authenticate to the container registry. |
| registry | String | FALSE | Identifier for the container registry host Jenkins should use during the image tagging and pushing processes. Defaults to "dev-registry.ercot.com". |
| product | String | TRUE | Name of the ERCOT product. i.e. cmm, noticebuilder, ftgui, etc |
| module | String | TRUE | Name of the component within the ERCOT product. i.e. frontend, backend, transformer, etc |

Generic Global Variable

The Generic pipeline takes a git repository with a "build" file at its base, runs dos2unix on the contents of the git repository, makes the build file executable, then executes the build file and archives any artifacts based on the comma separated list of extension patterns.

A Typical Jenkinsfile:

Generic {

buildImage = '<name-of-build-agent-image>'

archive = '<file-archiving-pattern>'

}

Configuration:

| **Name** | **Type** | **Required** | **Description** |
| --- | --- | --- | --- |
| cloud | String | FALSE | Identifier for the "cloud" Jenkins should use to provision dynamic agents. Defaults to "kubernetes". |
| buildImage | String | FALSE | Identifier for the image that will serve as the generic build agent. Defaults to "[dev-registry.ercot.com/ercot/jenkins/base-agent:latest](https://stash.ercot.com/projects/CICD/repos/jenkins-agent/browse)" |
| acctId | String | FALSE | Identifier for the Jenkins Credential used to authenticate to the artifact storage repository. |
| archive | String | FALSE | File pattern for gathering artifacts to be archived. i.e. \*\*/\*.war |

HelmChart Global Variable

The HelmChart pipeline takes a git repository contiaining a properly formed helm chart, builds the chart using the Helm command line tool, and pushes the chart to the designated help chart repository.

A Typical Jenkinsfile:

HelmChart {

chartName = '<name-of-ercot-product>'

appVersion = '<ercot-product-version-number>'

chartVersion = '<product-helm-chart-version-number>'

}

Configuration:

| **Name** | **Type** | **Required** | **Description** |
| --- | --- | --- | --- |
| cloud | String | FASLE | Identifier for the "cloud" Jenkins should use to provision dynamic agents. Defaults to "kubernetes". |
| buildImage | String | FALSE | Identifier for the image that will serve as the chart build agent. Defaults to "[dev-registry.ercot.com/ercot/jenkins/helm-agent:latest](https://stash.ercot.com/projects/CICD/repos/jenkins-helm-agent/browse)" |
| chartName | String | TRUE | Identifier for the chart. MUST match git repository name |
| chartVersion | String | TRUE | Version number for the Helm chart. |
| appVersion | String | TRUE | Version number for the product represented by the Helm chart. |
| keyNameId | String | FALSE | Identifier for the Jenkins Credential containing the desired signing key's name. |
| publicKeyId | String | FALSE | Identifier for the Jenkins Credential containing the desired signing key's public key. |
| privateKeyId | String | FALSE | Identifier for the Jenkins Credential containing the desired signing key's private key. |
| depRepos | Map | FALSE | Collection containing name/URL pairs for Helm Chart Repos. Note: Jenkins does not have internet access. |
| acctId | String | FALSE | Identifier for the Jenkins Credential used to authenticate to the chart repository. |
| repoHost | String | FALSE | Host name of the Helm chart repository. Defaults to "dev-registry.ercot.com". |
| repoName | String | FALSE | Name of the Helm chart repository within the repoHost. Defaults to "helm-dev". |

MavenArtifact Global Variable

The MavenArtifact pipeline takes a git repository containing a Maven project, analyzes the project's dependencies to determine whether they're up to date, runs the Maven package phase, collects outputs from JUnit, SpotBugs, CPD, PMD, and Checkstyle and generates reporting on the results. The MavenArtifact pipeline may be configured to handle managing POM versions as well as deploying artifacts to the artifact repository.

A Typical Jenkinsfile:

MavenArtifact {

deploy = true | false

}

Configuration:

| **Name** | **Type** | **Required** | **Description** |
| --- | --- | --- | --- |
| cloud | String | FALSE | Identifier for the "cloud" Jenkins should use to provision dynamic agents. Defaults to "kubernetes". |
| javaImage | String | FALSE | Identifier for the image that will serve as the java build agent. Defaults to "[dev-registry.ercot.com/ercot/jenkins/java-agent:latest](https://stash.ercot.com/projects/CICD/repos/jenkins-java-agent/browse)" |
| buildahImage | String | FALSE | Identifier for the image that will serve as the container build agent. Defauts to "[dev-registry.ercot.com/ercot/jenkins/buildah-agent:latest](https://stash.ercot.com/projects/CICD/repos/jenkins-buildah-agent/browse)" |
| helmImage | String | FALSE | Identifier for the image that will serve as the helm chart build agent. Defauts to "[dev-registry.ercot.com/ercot/jenkins/helm-agent:latest](https://stash.ercot.com/projects/CICD/repos/jenkins-helm-agent/browse)" |
| seleniumImage | String | FALSE | Identifier for the image that will serve as the selenium agent. Defauts to "[dev-registry.ercot.com/ercot/jenkins/selenium-agent:latest](https://stash.ercot.com/projects/CICD/repos/jenkins-selenium-agent/browse)" |
| mvnFlags | String | FALSE | Command line flags that should be passed to the mvn executable in addition to any implicitly configured flags. |
| version | String | FALSE | Version string for the Maven project. Note: May include Jenkins variables. i.e. ${BUILD\_NUMBER}. Including this variable will cause Jenkins to attempt to set the version number in the project POM file accordingly. |
| product | String | FALSE | ERCOT product identifier. Only used in container/chart based builds. |
| module | String | FALSE | Flag indicates whether Jenkins should treat the build process as a container/chart build or not. If this variable is specified the product variable should also be specified. |
| modules | List<String> | FALSE | Flag indicates whether Jenkins should treat the build process as a container/chart build or not. If this variable is specified the product variable should also be specified. |
| acctId | String | FALSE | Identifier for the Jenkins Credential used to authenticate to the chart repository/container registry. |
| repoHost | String | FALSE | Host name of the Helm chart repository. Defaults to "dev-registry.ercot.com". |
| repoName | String | FALSE | Name of the Helm chart repository within the repoHost. Defaults to "helm-dev". |
| deploy | boolean | FALSE | Flag dictating whether Maven artifacts should be deployed to the maven artifact repository. Defaults to false. Build jobs mapping to branches generally should not be deployed. Build jobs for container based products should rarely need to deploy these artifacts. |