

Apache KIE 10.2 :: Release procedure

This document describes the pieces that compose the Apache KIE 10.2 release and was created as a direct copy from its 10.1 basis, together with important information of how this is going to be achieved.

The table at the top contains all 5 repos that compose Apache KIE 10.2 with their respective commands to build, update versions, and notes on where final artifacts are created.

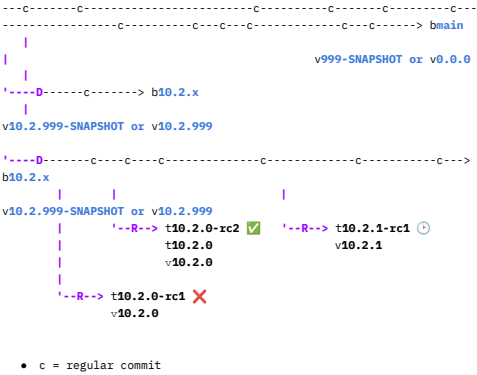
The "Branches strategy" section depicts a timeline of Git commits, with reasonable detail, for what the Git tree will look like moving forward.

The "Automations" section describes automatic jobs that are executed on Apache's Jenkins environment to achieve milestones of an Apache release.

#	REPO (incubator-kie-[repo])	GIT REF	OPERATING SYSTEM	REQUIREMENTS	BUILD COMMAND	PRODUCED ARTIFACTS	UPDATE OWN VERSION COMMAND (commits D and R)	UPDATE UPSTREAM VERSIONS (commits D and R)	Additional release command (commit R)
1	drools	TAG: 10.2.0	Ubuntu 22.04.4	- JDK 17.0.12 - Maven 3.9.6	mvn clean install -DskipTests -Dfull	JARs will be installed to local Maven repository	mvn versions:set -DnewVersion=<version> -DprocessAllModules -DgenerateBackupPoms=false	n/a (root repo)	None
2	optaplanner	TAG: 10.2.0	Ubuntu 22.04.4	- JDK 17.0.12 - Maven 3.9.6	mvn clean install -DskipTests -Dfull	JARs will be installed to local Maven repository	mvn versions:set -DnewVersion=<version> -DprocessAllModules -DgenerateBackupPoms=false	n/a → Same as \$(project.version)	None
3	kogito-runtimes	TAG: 10.2.0	Ubuntu 22.04.4	- JDK 17.0.12 - Maven 3.9.6	mvn clean install -DskipTests -Dfull	JARs will be installed to local Maven repository	mvn versions:set -DnewVersion=<version> -DprocessAllModules -DgenerateBackupPoms=false mvn -pl :kogito-quarkus-workflow-common-deployment versions:set-property -Dproperty=data-index-ephemeral.image.tagVersion -DgenerateBackupPoms=false -DnewVersion=<stream-name>	n/a → Same as \$(project.version)	None
4	kogito-apps	TAG: 10.2.0	Ubuntu 22.04.4	- JDK 17.0.12 - Maven 3.9.6 - GraalVM for JDK 17 - Docker 25+	mvn clean install -DskipTests -Dfull -Pjtexecutor-native	JARs will be installed to local Maven repository	mvn versions:set -DnewVersion=<version> -DprocessAllModules -DgenerateBackupPoms=false	n/a → Same as \$(project.version)	None

#	REPO (incubator-kie-[repo])	GIT REF	OPERATING SYSTEM	REQUIREMENTS	BUILD COMMAND	PRODUCED ARTIFACTS	UPDATE OWN VERSION COMMAND (commits D and R)	UPDATE UPSTREAM VERSIONS (commits D and R)	Additional release command (commit R)
5	kie-tools	TAG: 10.2.0	Ubuntu 22.04.4	- JDK 17.0.12 - Maven 3.9.6 - pnpm 9.3.0 - Go 1.23 - Make - Python 3.13 - Helm 3.16 - Node.js 22 - Docker 25+	pnpm bootstrap && KIE_TOOLS_BUILD__runTests=false KIE_TOOLS_BUILD__buildExamples=true KIE_TOOLS_BUILD__buildContainerImages=true pnpm -t build:prod	- JARs will be installed to local Maven repository - Container images will be build to the local image registry - Binaries and final artifacts at - packages/"dist - examples/"dist	pnpm update-version-to <version> - E.g., 10.2.0 or 10.2.999 - Needs to be SemVer-compatible. pnpm update-stream-name-to <stream-name> - E.g., 10.2.x or main	pnpm update-kogito-version-to --maven <version> E.g., --maven 10.2.999-SNAPSHOT; or --maven 10.2.0	- None TODO: Confirm --> For 10.2 we're not making it possible to consume Go modules as libraries

Branches strategy



- **D** = dev version commit → own version updated to major.minor.999-SNAPSHOT or major.minor.999 **AND** upstream versions update.
- **R** = release commit → version updated to major.minor.patch **AND** upstream versions update **AND** code changes for git-dependent ecosystems like Go modules.
- t = Git tag
- b = Git branch
- v = Project version
- | = automated process

-
- **Tags** are
 - major.minor.patch-rcN (E.g., 10.2.0-rc1)
 - major.minor.patch (E.g., 10.2.0)
 - **Tags** always:
 - Point to an **R** commit
 - Come from a “release branch” that is not pushed and is only used to locally do the **R** commit.
 - Originate from a **development stream**

-
- **Development streams** are:
 - - main → Keep 999-SNAPSHOT or 0.0.0 as version.
 - - major.minor.x → Keep major.minor.999-SNAPSHOT or major.minor.999 as version (E.g., 10.2.999)
-

Automations

Orange: Implemented, but requires verification and validation after changes from 10.0

Green: Good to go for 10.2!

- **AUTOMATION A:** Create new “minor” version development stream (E.g., 10.2.x)
 - Description
 - Sets up a new set of branches on all relevant repositories where the new release will come from.
 - **Exact execution steps**
 - Trigger [Drools prepare-release-branch](#) job

- set DROOLS_VERSION = 10.2.0 (**NOTE:** The versions of the Maven modules is inferred from this string to be 10.2.999-SNAPSHOT)
- click on **Build** button
- wait for the job to finish its execution
 - The prepare-release-branch job will keep retrying and waiting for the seed job to finish (which takes some time)
- **Kennedy Execution Notes**
 - **Drools nightly build failing due to**
<https://github.com/apache/incubator-kie-kogito-pipelines/commit/3e75837dffb2b90f1877b7b41c1d6e95b350f2e7> ,
reverted commit in 10.2.x and now passing
- Trigger [OptaPlanner prepare-release-branch](#) job
 - set OPTAPLANNER_VERSION = 10.2.0 (**NOTE:** The versions of the Maven modules is inferred from this string to be 10.2.999-SNAPSHOT)
 - click on **Build** button
 - wait for the job to finish its execution
 - **Kennedy Execution Notes**
 - **Optaplanner-quickstarts build failing for 2 reasons:**
 - **No 10.2.999-SNAPSHOT in the apache.repository.**
This is generated from the drools nightly job, which was failing.
 - **Optaplanner-operator used in maven command, removed with PR**
<https://github.com/apache/incubator-kie-optaplanner-quickstarts/commit/9852bb0a1901a91b162d99f192288168e16f39f7> and now passing
 - **Nightly build failing, missing GIT_AUTHOR(and probably more) in properties config**
- Trigger [Kogito prepare-release-branch](#) job
 - set KOGITO_VERSION = 10.2.0 (**NOTE:** The versions of the Maven modules is inferred from this string to be 10.2.999-SNAPSHOT)
 - set DROOLS_VERSION = 10.2.999-SNAPSHOT
 - click on **Build** button
 - wait for the job to finish its execution and timestamped SNAPSHOTs are published.
 - **NOTE:** This includes `kogito-runtimes` and `kogito-apps`
 - **Alex Execution Notes:**
 - **The execution of the job didn't trigger the seed, I had to run 0-seed-job manually**

- *0-seed-job parameters are the same of the previous execution, pointing only to main (the prepare-release-branch added a info into main that included 10.2.x, so it process everything again, and finally creates the 10.2.x jobs)*
- Trigger [KIE Tools setup-branch](#) job
 - set CHECKOUT_BRANCH_NAME = main
 - set NEW_BRANCH_NAME = 10.2.x
 - set KIE_TOOLS_VERSION = 10.2.999
 - set KOGITO_MAVEN_VERSION = 10.2.999-SNAPSHOT
 - click on **Build** button
 - wait for the job to finish its execution
 - **Alex Execution Notes**
 - *KOGITO_IMAGES_TAG missing*
 - *Error with expected folder that didn't exist anymore*
 - *Renamed the ci(disabled) inside main to kie-tools-ci-jobs and moved it to root of kie-tools*
 - *Trigger again, which worked!*
 - **Kennedy Execution Notes**
 - *Moved the ci folder from main*
<https://ci-builds.apache.org/job/KIE/job/kie-tools/job/main/>,
to the root <https://ci-builds.apache.org/job/KIE/job/kie-tools/>
and renamed it from ci to kie-tools-ci-jobs
 - *Ran into heap space errors so added a new parameter, Created PR to add it to the job by default*
<https://github.com/apache/incubator-kie-kogito-pipelines/pull/1286> .
- Manually duplicate `main` folder for KIE Tools automations
 - Navigate to <https://ci-builds.apache.org/job/KIE/job/kie-tools/>
 - Go to <https://ci-builds.apache.org/job/KIE/job/kie-tools/newJob> and name it 10.2.x
 - 👉 At the bottom, use the “Copy from” field and enter “main”.
- Manually duplicate `main` folder for Misc. automations
 - Navigate to https://ci-builds.apache.org/job/KIE/job/_misc/
 - Go to https://ci-builds.apache.org/job/KIE/job/_misc/newJob and name it 10.2.x
 - 👉 At the bottom, use the “Copy from” field and enter “main”.

- **AUTOMATION B:** Create new CI jobs for “minor” version development stream (E.g., 10.2.x)
 - Description
 - Create new CI jobs for the newly created stream.
 - **Exact execution steps**
 - None. **AUTOMATION B** is triggered as sub-jobs of **AUTOMATION A** automatically
 - (+) In case of kogito-pipelines framework the prepare-release-branch jobs write into pipeline config which then triggers the DSL generation of the pipelines for the newly added branch. (adding 10.2.x into .ci/jenkins/config/main.yaml in main branch of drools/kogito-pipelines/optaplanner)

- **AUTOMATION C:** Create new CD jobs for daily SNAPSHOTS of the new “minor” version development stream (E.g., 10.2.999-SNAPSHOT)
 - Description
 - Publishes Container images (I.e., 10.2.x tags) and public Maven libraries (I.e., 10.2.999-SNAPSHOT)
 - **Exact execution steps**
 - None. **AUTOMATION C** is automatically triggered as a sub-job of **AUTOMATION A**. For KIE Tools, when you duplicated the `main` folder on Jenkins, you already configured these daily jobs, so there is no further action here.

- **AUTOMATION D:** Release candidate job on Jenkins
 - Inputs
 - Branch name: E.g., 10.2.x
 - Tag name: E.g., 10.2.0-rc1
 - Packages/artifacts version: 10.2.0
 - Results
 - Release candidate tags pushed to all 5 GitHub repos
 - Sources + artifacts on <https://dist.apache.org/repos/dist/dev/incubator/kie>
 - Description
 - **NOTE:** These steps are for reading only. Do not execute these steps manually. Proceed to “Exact execution steps” for performing AUTOMATION A.
 - **NOTE:** Dependencies are fetched from:
 - 1. Maven staging repos (*drools*, *optaplanner*, *kogito-runtimes*, and *kogito-apps*)

- Trigger jobs that will build the release artifacts
 - Release candidate tags on GitHub repos
 - git clone all 5 repos @ [input Branch name]
 - git checkout origin/10.2.x
 - Do an **R** commit by running the configured scripts (updating own and upstream dependencies versions)
 - git tag [input Tag name]
 - Push new RC tag
 - Artifacts (Built from RC tag. E.g., 10.2.0-rc1)
 - (-) **ALL ARTIFACTS NEED A CHECKSUM AND A DIGITAL SIGNATURE FILE ASSOCIATED WITH IT @**
<https://dist.apache.org/repos/dist/dev/incubator/kie/10.2.0-rc1>
 - [artifact].sha512 ← Checksum
 - [artifact].asc ← Digital signature
 - **Sources**
 - All 5 GitHub repos
 - incubator-kie-10.2.0-rc1-sources.zip
 - **Maven (drools, optaplanner, kogito-runtimes, kogito-apps, kie-tools)**
 - <https://infra.apache.org/publishing-maven-artifacts.html> (Not uploaded to <https://dist.apache.org>)
 - Repos
 - drools
 - optaplanner
 - kogito-runtimes
 - kogito-apps
 - kie-tools
 - **Helm charts (kie-tools)**
 - packages/kie-sandbox-helm-chart
 - incubator-kie-10.2.0-rc1-sandbox-helm-chart.tar.gz
 - incubator-kie-10.2.0-rc1-runtime-tools-helm-chart.tar.gz **(NEW IN 10.2)**
 - **Container images (kie-tools)**
 - - incubator-kie-10.2.0-rc1-kogito-base-builder-image.tar.gz
 - incubator-kie-10.2.0-rc1-kogito-data-index-ephemeral-image.tar.gz

- incubator-kie-10.2.0-rc1-kogito-data-index-postgresql-image.tar.gz
- incubator-kie-10.2.0-rc1-kogito-jit-runner-image.tar.gz
- incubator-kie-10.2.0-rc1-kogito-jobs-service-allinone-image.tar.gz
- incubator-kie-10.2.0-rc1-kogito-jobs-service-ephemeral-image.tar.gz
- incubator-kie-10.2.0-rc1-kogito-jobs-service-postgresql-image.tar.gz
- incubator-kie-10.2.0-rc1-kogito-management-console-image.tar.gz
- incubator-kie-10.2.0-rc1-kogito-task-console-image.tar.gz
- **(DELETED IN 10.2)**
- incubator-kie-10.2.0-rc1-kogito-db-migrator-image.tar.gz **(NEW IN 10.2)**
- incubator-kie-10.2.0-rc1-cors-proxy-image.tar.gz
- incubator-kie-10.2.0-rc1-dashbuilder-viewer-image.tar.gz
- incubator-kie-10.2.0-rc1-sandbox-webapp-image.tar.gz
- incubator-kie-10.2.0-rc1-sandbox-extended-services-image.tar.gz
- incubator-kie-10.2.0-rc1-sandbox-dev-deployment-base-image.tar.gz
- incubator-kie-10.2.0-rc1-sandbox-dev-deployment-dmn-form-webapp-image.tar.gz
- incubator-kie-10.2.0-rc1-sandbox-dev-deployment-quarkus-blank-app-image.tar.gz

- incubator-kie-10.2.0-rc1-serverless-logic-web-tools-base-builder-image.tar.gz
- incubator-kie-10.2.0-rc1-serverless-logic-web-tools-swf-builder-image.tar.gz
- incubator-kie-10.2.0-rc1-serverless-logic-web-tools-swf-dev-mode-image.tar.gz
- incubator-kie-10.2.0-rc1-sonataflow-builder-image.tar.gz
- incubator-kie-10.2.0-rc1-sonataflow-devmode-image.tar.gz
- incubator-kie-10.2.0-rc1-sonataflow-operator-image.tar.gz
- incubator-kie-10.2.0-rc1-sonataflow-management-console-image.tar.gz **(NEW IN 10.2)**
- **NPM packages (kie-tools)**
 - All public packages on kie-tools
 - incubator-kie-10.2.0-rc1-tools-npm-packages.zip
- **Extended Service binaries for macOS x86, macOS arm64, Windows x86 and Linux x86 (kie-tools)**
 - packages/extended-services
 - ~~incubator-kie-10.2.0-rc1-sandbox-extended-services-macOS-arm64.zip~~
 - ~~incubator-kie-10.2.0-rc1-sandbox-extended-services-macOS-x86.zip~~
 - ~~incubator-kie-10.2.0-rc1-sandbox-extended-services-windows-x86.zip~~
 - incubator-kie-10.2.0-rc1-sandbox-extended-services-linux-x86.tar.gz
- **KIE Sandbox Dev deployment upload service (kie-tools)**
 - packages/dev-deployment-upload-service

- incubator-kie-10.2.0-rc1-sandbox-dev-deployment-upload-service-macos-arm64.tar.gz
 - incubator-kie-10.2.0-rc1-sandbox-dev-deployment-upload-service-macos-x86.tar.gz
 - incubator-kie-10.2.0-rc1-sandbox-dev-deployment-upload-service-windows-x86.tar.gz
 - incubator-kie-10.2.0-rc1-sandbox-dev-deployment-upload-service-linux-x86.tar.gz
- **KIE Sandbox webapp (kie-tools)**
 - packages/online-editor
 - incubator-kie-10.2.0-rc1-sandbox-webapp.zip
- **Standalone Editors CDN (kie-tools)**
 - packages/kie-editors-standalone
 - incubator-kie-10.2.0-rc1-business-automation-standalone-editors.zip
 - packages/serverless-workflow-standalone-editor
 - incubator-kie-10.2.0-rc1-sonataflow-standalone-editors.zip
- **VS Code Extensions (kie-tools)**
 - packages/bpmn-vscode-extension
 - incubator-kie-10.2.0-rc1-bpmn-vscode-extension.vsix
 - packages/dmn-vscode-extension
 - incubator-kie-10.2.0-rc1-dmn-vscode-extension.vsix
 - packages/pmml-vscode-extension
 - incubator-kie-10.2.0-rc1-pmml-vscode-extension.vsix
 - packages/yard-vscode-extension
 - incubator-kie-10.2.0-rc1-yard-vscode-extension.vsix
 - packages/vscode-extension-dashbuilder-editor
 - incubator-kie-10.2.0-rc1-dashbuilder-vscode-extension.vsix

- packages/serverless-workflow-vscode-extension
 - incubator-kie-10.2.0-rc1-sonataflow-vscode-extension.vsix
 - packages/vscode-extension-kogito-bundle
 - incubator-kie-10.2.0-rc1-kogito-bundle-vscode-extension.vsix
 - packages/vscode-extension-kie-ba-bundle
 - incubator-kie-10.2.0-rc1-business-automation-bundle-vscode-extension.vsix
 - packages/extended-services-vscode-extension
 - incubator-kie-10.2.0-rc1-extended-services-vscode-extension.vsix
- **Chrome Extensions (kie-tools)**
 - packages/chrome-extension-pack-kogito-kie-editors
 - incubator-kie-10.2.0-rc1-business-automation-chrome-extension.zip
 - incubator-kie-10.2.0-rc1-business-automation-chrome-extension-editors.zip
 - packages/chrome-extension-serverless-workflow-editor
 - incubator-kie-10.2.0-rc1-sonataflow-chrome-extension.zip
 - incubator-kie-10.2.0-rc1-sonataflow-chrome-extension-editors.zip
- **kn-plugin-workflow (kie-tools)**
 - packages/kn-plugin-workflow
 - incubator-kie-10.2.0-rc1-sonataflow-knative-plugin-macOS-arm64.zip
 - incubator-kie-10.2.0-rc1-sonataflow-knative-plugin-macOS-x86.zip
 - incubator-kie-10.2.0-rc1-sonataflow-knative-plugin-windows-x86.zip
 - incubator-kie-10.2.0-rc1-sonataflow-knative-plugin-linux-x86.zip
- **sonataflow-operator.yaml (kie-tools)**

- packages/kogito-serverless-operator
 - incubator-kie-10.2.0-rc1-sonataflow-operator.zip
- **kie-sandbox-accelerator-quarkus**
 - packages/kie-sandbox-accelerator-quarkus
 - incubator-kie-10.2.0-rc1-sandbox-accelerator-quarkus.zip **(NEW IN 10.2)**

○ Exact execution steps

- Drools:
 - Remove the drools-docs module directly on github
 - drools-doc contains files that are not allowed for the source release (including .zip and other .js/font files)
 - Kennedy Execution Notes:
 - Most of the following steps aren't needed. Drools-docs has since been moved to a different repository. There's a few references to it in the codebase, but potentially only need to remove it from the bom, if that.
 - Steps executed:
 - `git clone`
`git@github.com:apache/incubator-kie-drools.git`
 - `cd incubator-kie-drools`
 - `git co 10.2.x`
 - `git branch 10.2-docs-removal`
 - `git co 10.2-docs-removal`
 - `rm -rf drools-docs`
 - `//remove code reference of drools-docs from the root pom.xml file`
 - `mvn clean install -DskipTests -PfullProfile`

- `rm -rf`
drools-test-coverage/test-compiler-integration/src/test/resources/only-jar-pojo-not-kjar-no-kmodule-10.2.999-SNAPSHOT.jar
- `git status`
- `git add .`
- `git commit -m "removal`
drools-docs from the repository
as it contains invalid
artifacts for the source
release (apache policy issues)"
- `git push origin`
- `//open PR`

● ~~Update [Drools 10.2.x release deploy job](#):~~

○ ~~Set the property:~~

~~MAVEN_REPO_CREDS_ID=jenkins-deploy-to-nexus-staging~~

- Alex Execution Notes:
 - This step was not needed; already in place.

● Trigger [Drools 10.2.x release job](#):

- Drools → 10.2.x → release → 0-drools-release
- set RELEASE_VERSION = 10.2.0
- Set GIT_TAG_NAME = 10.2.0-rc1
- click on **Build** button
- wait for the job to finish its execution
- Close Nexus Staging [Drools] repository
- Alex Execution Notes:
 - Identified that all Jenkin jobs are not working; changes were made directly to 10.0.x and never brought back to main. The main also changed, so all pipelines need to be reconciled.
 - After changing the drools pipeline, started with the parameters above
 - Note 2: there's another parameter about resume build, that

- Once pipeline fixed, no additional issues found and drools artifacts were successfully deployed to staging nexus
 - Missing close nexus instruction [now added]
 - Error on closing, signature failure. Had to upload to KYES again to different server
- OptaPlanner:
 - Trigger [OptaPlanner 10.2.x release job](#):
 - set RELEASE_VERSION = 10.2.0
 - Set GIT_TAG_NAME = 10.2.0-rc1
 - SKIP_TESTS = Enabled
 - click on **Build** button
 - wait for the job to finish its execution
 - Close nexus staging repository
 - Alex Execution Notes:
 - Updated the jenkins pipelines from the 10.0.x branch, similar to drools. Optaplanner less impacted, changes were very minimal.
 - Build failed, due operator. Action was to remove the operator, as in the 10.0.0 release.
 - After pushing the operator release, retrigger the build with the same parameters.
 - Build failed, but it's expected... as the pipeline expects to push the operator image, but not exists.
 - Even with failure, the artifacts are preserved and pushed to nexus
 - Kennedy Execution Notes:
 - Not working due to missing properties in the "Properties Content" of the Configuration of the Jenkins job. Copied from 10.1.x.
- Kogito Runtimes:
 - Trigger [Kogito Runtimes 10.2.x release deploy job](#):
 - Set PROJECT_VERSION = 10.2.0
 - Set DROOLS_VERSION = 10.2.0
 - Set GIT_TAG_NAME = 10.2.0-rc1
 - Set SKIP_TESTS = Enabled
 - click on **Build** button
 - wait for the job to finish its execution
 - Close nexus staging repo
- Kogito Apps:
 - Update [Kogito Apps 10.2.x release deploy job](#):
 - Set the property:
MAVEN_REPO_CREDS_ID=jenkins-deploy-to-nexus-staging

- Trigger [Kogito Apps 10.2.x release deploy job](#):
 - Set PROJECT_VERSION = 10.2.0
 - Set GIT_TAG_NAME = 10.2.0-rc1
 - click on **Build** button
 - wait for the job to finish its execution
- Trigger [Jitexecutor-native workflow](#):
 - Run: `gh workflow run publish-jitexecutor-native-rc.yml --ref 10.2.x -f release_candidate_version=10.2.0 -f drools_repository="https://repository.apache.org/content/repositories/orgapachekie-XXXX" -f kogito_runtimes_repository="https://repository.apache.org/content/repositories/orgapachekie-XXXX" --repo=https://github.com/apache/incubator-kie-kogito-apps`
 - Replace XXXX with respective repository suffix from the nexus staging repo
 - Close nexus staging repo after this step?
 - Kennedy Execution Notes:
 - Failing after 1s, any manual intervention required?
- KIE Tools:
 - Trigger [KIE Tools release-candidate](#) job
 - set BRANCH_NAME = 10.2.x
 - set RELEASE_VERSION = 10.2.0
 - set TAG_NAME = 10.2.0-rc1
 - click on **Build** button
 - wait for the job to finish its execution
 - Trigger [Extended-services release build workflow](#):
 - Kennedy Execution Notes:
 - File doesn't exist in 10.2.x, is there an alternative step we need to do?
 - Run: `gh workflow run release_build_extended_services.yml \`
 - `--repo=https://github.com/apache/incubator-kie-tools \`
 - `--ref 10.2.x \`
 - `-f dry_run=false \`
 - `-f base_ref=10.2.x \`
 - `-f`
`jitexecutor_native_macos_repository=https://repository.apache.org/content/repositories/staging/ \`
 - `-f`
`jitexecutor_native_windows_repository=https://repository.apache.org/content/repositories/staging/ \`

- -f
kogito_runtimes_repository=https://repository.apache.org/content/repositories/staging/ \
- -f
kogito_apps_repository=https://repository.apache.org/content/repositories/staging/ \
- -f
drools_repository=https://repository.apache.org/content/repositories/staging/ \
- -f release_candidate_version=10.2.0-rc1 \
- -f release_version=10.2.0
- Trigger
 - <https://ci-builds.apache.org/job/KIE/job/misc/job/10.2.x/job/zip-and-upload-sources/>
 - set TARGET_VERSION = 10.2.0-rc1
 - set SOURCES_DEFAULT_BRANCH = 10.2.0-rc1
 - click on **Build** button
 - wait for the job to finish its execution
- **NOTES:**
 - For the steps where we need to set the MAVEN_REPO_CREDS_ID property, this must be automated in the following releases. Currently we only support a single credentials id in our configuration which is currently set with the snapshots credentials. We need to implement a new configuration to allow us to set the staging credentials.
 - The same applies to the step where we update the settings.xml file to add a staging repository. Currently we have a configuration that allows us to set only a single repository. That must be changed to support multiple staging repositories.

● **AUTOMATION E:** Staging release jobs

- Not in scope given the lack of internal infrastructure for publishing artifacts with final versions.
- Staging and sanity checks will be done from source.

● **AUTOMATION F:** Git tag release jobs

- Inputs
 - Approved release candidate tag name: E.g., 10.2.0-rc3
 - Release tag name: E.g., 10.2.0
- Results
 - Release tags pushed to all 5 GitHub repos
- Description
 - Create tags in all jobs pointing to the approved release candidate tag
 - E.g., 10.2.0-rc3 is approved. Release tag 10.2.0 ==> 10.2.0-rc3

- **Exact execution steps**
 - git tag 10.2.0 10.2.0-rc1 in all repos
- **AUTOMATION G:** Promote release candidate (to <https://dist.apache.org/repos/dist/release/incubator/kie/10.2.0> and other public repositories, like Docker Hub, NPM etc)
 - Manual
 - X Maven (drools, optaplanner, kogito-runtimes, kogito-apps, kie-tools)
 - Promote from staging Maven repo
 - Automated

(<https://ci-builds.apache.org/job/KIE/job/kie-tools/job/10.2.x/job/release/job/kie-tools-release-promote>)

 - X Helm charts (kie-tools)
 - helm push
 - X Container images (kie-tools)
 - docker load + docker push
 - Tags should be loaded from the tarball.
 - X NPM packages (kie-tools)
 - npm publish for the .zip files individually
 - See [https://docs.npmjs.com/cli/v10/commands/npm-publish#:~:text=b%20a%20gzipped%20tarball%20containing%20\(a\)](https://docs.npmjs.com/cli/v10/commands/npm-publish#:~:text=b%20a%20gzipped%20tarball%20containing%20(a))
 - X Extended Service binaries for macOS x86, macOS arm64, Windows x86 and Linux x86 (kie-tools)
 - Upload binaries individually the GitHub release on `kie-tools`
 - X KIE Sandbox webapp (GitHub Pages) (kie-tools)
 - Unzip at correct location and push to `incubator-kie-kogito-online` @ gh-pages branch
 - X Standalone Editors "CDN" (GitHub Pages) (kie-tools)
 - Unzip at correct location and push to `incubator-kie-kogito-online` @ gh-pages branch
 - X Chrome Extension Editors (GitHub Pages) (kie-tools)
 - Unzip at correct location and push to `incubator-kie-kogito-online` @ gh-pages branch
 - X VS Code Extensions (kie-tools)
 - Upload VSIX files to marketplace
 - Manual upload possible by drag-and-drop.
 - X Chrome Extensions (kie-tools)
 - Upload .zip chrome extension to the marketplace
 - Use kietooling@gmail.com to log-in to Chrome.
 - Manual upload possible by drag-and-drop.
 - X kn-plugin-workflow (kie-tools)
 - Upload binaries individually the GitHub release on `kie-tools`.
 - X sonataflow-operator.yaml (kie-tools)
 - Upload file to GitHub release on `kie-tools`.

- **(NEW IN 10.2)** KIE Sandbox Quarkus Accelerator
 - Git push to <https://github.com/apache/incubator-kie-sandbox-quarkus-accelerator> @ 10.2.0
- **(Added IN 10.2)**
 - Use svn mv to move artifacts from dev folder to release folder
 - Remove any old RCs still present in the dev folder
 - E.g. svn delete <https://dist.apache.org/repos/dist/dev/incubator/kie/10.2.0-rc1/> -m "Remove 10.2.0-rc1 release candidate"
- **Exact execution steps**
 - All manual

● **MANUAL: DOCUMENTATION RELEASE Procedure**

- Process for releasing new version of documentation and its publishing on the website
- Website - <https://github.com/apache/incubator-kie-website/>
- Documentation is currently hosted in these repositories
 - Drools - <https://github.com/apache/incubator-kie-docs>
 - Kogito - <https://github.com/apache/incubator-kie-docs>
 - Optaplanner - <https://github.com/apache/incubator-kie-docs>
 - Sonataflow - <https://github.com/apache/incubator-kie-kogito-docs>
- Documentation is being included to the website as static content that is built every time new updates are deployed.
 - See the [deploy](#) workflow for further details
- **Exact execution steps**
 - Drools
 - Branch is created using automation.
 - Prepare a pull request that provides updates of versions to 10.2.0 and ask for review
 - Kogito & Drools
 - Create a branch from main branch and name it 10.2.x
 - Prepare a pull request that provides updates of versions to 10.2.0 and ask for review
 - Sonataflow
 - Branch is created using automation.
 - Prepare a pull request that provides the versions to 10.2.0 and ask for review
 - Update the [antora-playbook](#) to include the latest version, under *branches* attribute
 - Website
 - Prepare a pull request that adds new version to the version matrix
 - Update Documentation page with the new 10.2 version

Release procedures

- Major

- Undefined yet. We're not planning on releasing a new major anytime soon.

- Minor (E.g, main → v10.2.0)

- Run **AUTOMATION A**.
- Run **AUTOMATION B**.
- Run **AUTOMATION C**.
- Run daily snapshots jobs created by **AUTOMATION C**.
- Run **AUTOMATION D** for sanity check as release candidate zero (rc0) (E.g., New release candidate tag 10.2.0-rc0).
- Perform sanity checks using the sources .zip from **AUTOMATION D**.
- ⚠ Blockers found? Push new commits to the development stream branch (E.g., 10.2.x) and go back to step 5.
- Once confident, start a vote on the Mailing List (dev@kie.apache.org) (Apache KIE first 72h, then IPMC 72h).
- Wait for release approval on the Mailing List.
- Run **AUTOMATION F**, then **AUTOMATION G**.

- Patch (E.g., 10.2.x → v10.2.4)

- Run **AUTOMATION D** from a minor development stream branch (E.g., 10.2.x branch).
- Perform sanity checks using the sources .zip from **AUTOMATION D**.
- ⚠ Blockers found? Push new commits to the development stream branch (E.g., 10.2.x) and go back to step 1.
- Once confident, start a vote on the Mailing List (dev@kie.apache.org) (Apache KIE first 72h, then IPMC 72h).
- Wait for release approval on the Mailing List.
- Run **AUTOMATION F**, then **AUTOMATION G**.