

**git\_comments:**

1. `#!/usr/bin/env bash ##### Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. #####`
2. fail on errors

**git\_commits:**

1. **summary:** [FLINK-18643][Azure] Build a Flink snapshot release with the nightly cron-job.  
**message:** [FLINK-18643][Azure] Build a Flink snapshot release with the nightly cron-job. This closes #13125

**github\_issues:****github\_issues\_comments:****github\_pulls:**

1. **title:** [FLINK-18643] Deploy Flink snapshot build with nightly CI job  
**body:** ## What is the purpose of the change Every night, we will deploy a SNAPSHOT version to Apache's snapshot maven repository. The Flink PMC got special credentials for that purpose. We are also creating a Flink binary release every night. The binaries are provided as Azure artifacts. We are re-using existing scripts for creating regular releases. ## Verifying this change This is a test run of the two jobs: [https://dev.azure.com/rmetzger/Flink/\\_build/results?buildId=8262&view=results](https://dev.azure.com/rmetzger/Flink/_build/results?buildId=8262&view=results)

**github\_pulls\_comments:**

1. Thanks a lot for your contribution to the Apache Flink project. I'm the @flinkbot. I help the community to review your pull request. We will use this comment to track the progress of the review. ## Automated Checks Last check on commit b6c7c2440191c0fba1a8ca796bbc15e70eeb049e (Wed Aug 12 05:47:24 UTC 2020) \*\*Warnings:\*\* \* No documentation files were touched! Remember to keep the Flink docs up to date! <sub>Mention the bot in a comment to re-run the automated checks.</sub> ## Review Progress \* ? 1. The [description] looks good. \* ? 2. There is [consensus] that the contribution should go into to Flink. \* ? 3. Needs [attention] from. \* ? 4. The change fits into the overall [architecture]. \* ? 5. Overall code [quality] is good. Please see the [Pull Request Review Guide](<https://flink.apache.org/contributing/reviewing-prs.html>) for a full explanation of the review process.<details> The Bot is tracking the review progress through labels. Labels are applied according to the order of the review items. For consensus, approval by a Flink committer of PMC member is required <summary>Bot commands</summary> The @flinkbot bot supports the following commands: - '@flinkbot approve description' to approve one or more aspects (aspects: 'description', 'consensus', 'architecture' and 'quality') - '@flinkbot approve all' to approve all aspects - '@flinkbot approve-until architecture' to approve everything until 'architecture' - '@flinkbot attention @username1 [@username2 ..]' to require somebody's attention - '@flinkbot disapprove architecture' to remove an approval you gave earlier </details>
2. <!-- Meta data { "version" : 1, "metaDataEntries" : [ { "hash" : "b6c7c2440191c0fba1a8ca796bbc15e70eeb049e", "status" : "SUCCESS", "url" : "https://dev.azure.com/apache-flink/98463496-1af2-4620-8eab-a2ecc1a2e6fe/\_build/results?buildId=5433", "triggerID" : "b6c7c2440191c0fba1a8ca796bbc15e70eeb049e", "triggerType" : "PUSH" }, { "hash" : "c24b96bd51f97fb3922d61dd7338a5dd5ff9036b", "status" : "UNKNOWN", "url" : "TBD", "triggerID" : "c24b96bd51f97fb3922d61dd7338a5dd5ff9036b", "triggerType" : "PUSH" } ] }--> ## CI report: \* b6c7c2440191c0fba1a8ca796bbc15e70eeb049e Azure: [SUCCESS]([https://dev.azure.com/apache-flink/98463496-1af2-4620-8eab-a2ecc1a2e6fe/\\_build/results?buildId=5433](https://dev.azure.com/apache-flink/98463496-1af2-4620-8eab-a2ecc1a2e6fe/_build/results?buildId=5433)) \* c24b96bd51f97fb3922d61dd7338a5dd5ff9036b UNKNOWN <details> <summary>Bot commands</summary> The @flinkbot bot supports the following commands: - '@flinkbot run travis' re-run the last Travis build - '@flinkbot run azure' re-run the last Azure build </details>
3. I generally want to reduce the number of scripts that we have around + using the release scripts for nightly and full releases seems elegant: We'll ensure that the release scripts are working during the release cycle. But I can combine my desire with your suggestion and move the s3 upload part into the pipeline definition, because I generally agree that uploading to S3 is a good idea.
4. S3 Upload ✓ `task: CmdLine@2` ✓ renaming displayName: ❌ (will do while merging)
5. Thanks a lot for your review!

**github\_pulls\_reviews:**

1. what is this mirror for? If it does what I think it is (mirroring the artifacts to maven central where they are publicly accessible to all users), then it should be removed.
2. ```suggestion export RELEASE\_VERSION=\$(MVN\_RUN\_VERBOSE=false run\_mvn help:evaluate -Dexpression=project.version -q -DforceStdout)```
3. ```suggestion stage\_name: cron\_snapshot\_deployment```
4. ```suggestion - task: PublishPipelineArtifact@1 displayName: Upload snapshot binary release```
5. is it a hard requirement that the script is the first item? If not, then having the displayName as the first item would eliminate the requirement of needing a comment to explain what it does.
6. This mirror is for downloading artifacts more reliably from Google's CDN, instead of the Apache Maven Central servers.
7. I don't think there's anything we can do about it.
8. ```suggestion displayName: Build snapshot binary release```
9. well, that makes sense, and in hindsight was rather obvious.
10. what we could maybe do is specify `task: CmdLine@2`, have the display name below, then the script block? I think we can make these files much more readable if the display name is closer to the start of the block.
11. Yep, that works. Once I have validated that the S3 upload works, I'll update the PR.

12. ``suggestion - task: Cache@2 displayName: Cache Maven local repo inputs: key: \$(CACHE\_KEY) restoreKeys: \$(CACHE\_FALLBACK\_KEY) path: \$(MAVEN\_CACHE\_FOLDER) continueOnError: true ``

#### jira\_issues:

1. **summary:** Migrate Jenkins jobs to ci-builds.apache.org

**description:** Infra is [reworking the Jenkins

setup](https://lists.apache.org/thread.html/re974eed417a1bc294694701d5c91b4bf92689fcf32a4c91f169be87d%40%3Cbuilds.apache.org%3E), so we have to migrate our jobs that do the snapshot deployments. Alternatively, find other ways to do this (Azure?) to reduce number of used infrastructure services. /cc [~rmetzger]

#### jira\_issues\_comments:

1. Thanks for spotting this announcement by Infra. I'm not sure how many people are really using the snapshot versions we are deploying. I would actually propose to stop publishing nightly builds. If we want to continue, Azure DevOps has a module called "Artifacts", that also seems to support Maven: <https://docs.microsoft.com/en-us/azure/devops/pipelines/artifacts/maven?view=azure-devops> It should be fairly simple to integrate that with the nightly build. For a nightly "binary release", we could just upload them as a build artifact (finding that file will be insanely difficult for our users though). Alternatively, we keep uploading them to s3 (from Azure). Shall we do a quick survey on the dev@ and user@ list to understand the need for nightly builds?
2. ??For a nightly "binary release", we could just upload them as a build artifact (finding that file will be insanely difficult for our users though). Alternatively, we keep uploading them to s3 (from Azure).?? That is fine since we are not allowed to advertise snapshot binaries to \_users\_ anyway. The only people that should be able to find it are developers, and there we can be as obtuse as we want. ??If we want to continue, Azure DevOps has a module called "Artifacts", that also seems to support Maven[...]? We'd have to check whether we are allowed to put ASF credentials into a non-ASF managed Azure account. That is, if we want to publish to the ASF snapshot repository (which I suppose we do want). ??I'm not sure how many people are really using the snapshot versions we are deploying.?? I use them personally, and would suspect that other developers do too. The snapshot binaries are also required for ensuring that the docker images are compatible with the upcoming Flink version.
3. Just the other day I had a user asking if nightly builds are available because they wanted to see if a fix that was not yet released is working for them. See here: <https://issues.apache.org/jira/browse/FLINK-18478?focusedCommentId=17158523&page=com.atlassian.jira.plugin.system.issuetabpanels%3Acomment-tabpanel#comment-17158523>
4. I agree. It would be nice to keep publishing nightlies.
5. Looks like the feedback is overwhelmingly for keeping the nightlies :) > We'd have to check whether we are allowed to put ASF credentials into a non-ASF managed Azure account. That is, if we want to publish to the ASF snapshot repository (which I suppose we do want). I don't think ASF Infra is giving out access keys for non-humans to outside infrastructure. But what's the problem with pushing it to Azure's maven snapshot repository? I would vote to just do that for simplicity reasons.
6. > But what's the problem with pushing it to Azure's maven snapshot repository? I would vote to just do that for simplicity reasons. All users of snapshot artifacts need to update their snapshot repository settings, and we have to update our documentation.
7. Fair point. I guess I'm using the snapshot repository not very frequently, that's why I don't have a strong opinion here. Looks like we need to ask INFRA if this is possible.
8. I have asked in INFRA-20629.
9. Azure Artifacts does not support snapshot versions very well: <https://developercommunity.visualstudio.com/idea/852929/add-support-for-maven-snapshot-versions-in-azure-a.html> They don't allow overwriting existing artifacts. We would have to create a new repository every night (even deleting artifacts and re-uploading them is not possible). I'm not very confident that INFRA will create us a service account for uploading snapshots. Maybe we should look into GitHub packages: <https://docs.github.com/en/packages/using-github-packages-with-your-projects-ecosystem/configuring-apache-maven-for-use-with-github-packages> (through flink-ci) (it seems to support SNAPSHOT deployments). I have a branch ready that adds support for creating a binary release with the nightly run.
10. Resolved in <https://github.com/apache/flink/commit/8ec8b63b4c1473d2e28fe8606a0b7bc6aa918e3e>
11. [~rmetzger] What about the 1.10/1.11 snapshots?
12. I haven't considered this, but yes, I'm able to backport the changes to 1.11 easily. I would push the change once the nightly creation has worked on master (it didn't last night, which was the first night (fixed)) For 1.10, my strategy would be "hope". If somebody really needs a snapshot build, they need to build it themselves, or we manually deploy one. But I'm not sure if it is worth the effort, without any known users. Are you okay with removing the build configuration from Jenkins now?
13. The upload worked on master this night. Pushed change to release-1.11: <https://github.com/apache/flink/commit/0782d01315abebb360185a5f2c38eddcc7b14f4e>
14. hmm...I guess it's okay to only cover 1.11+ with this. Go ahead and remove the jenkins setup.
15. I've disabled the Jenkins profiles for Flink (except the statefun snapshot deployment ones)