Releasing jclouds

Checklist Summary of Steps Involved in Releasing

1. Email the dev list to let them know the release process is beginning and ask all to clean up any JIRA issues they are familiar with.
2. Have a GPG signing key configured.
3. Clone all repos to be released, if you haven't already.
4. Check out the branch to be released in each repo.
5. Update any POM properties in the repos pointing at the SNAPSHOT version and change those to the release version.
6. Run mvn release commands for each repo, in order, as described below. If building a 1.6.x release, use a Java 6 JDK
7. Push the release commits and RC tags to the remote repos.
8. Close the Nexus staging repository.
9. Copy the source release tarballs, signatures and hashes to a new release candidate directory.
10. Email the dev list, opening the 72 hour vote window on the RC.
11. If the vote fails or there are issues that need to be addressed, cancel, rinse and repeat.
12. Once the PMC vote passes, release the Nexus staging repo.
13. Send a RESULT email in reply to the PMC vote thread with the result.
14. Copy the RC tarballs etc to the release area.
15. Push non-RC tags to the remote repos.
16. Update the version on the branch in the repos to the next SNAPSHOT and push.
17. Update the jclouds.apache.org site.
    a. Add a new release notes file in releasenotes.
    b. Update latest_version and latest_snapshot (if necessary) in _config.yml
    c. Add the release to the DOAP file
    d. Deploy the documentation.
18. Update jclouds-examples with any relevant changes.
19. Publish the API docs for the release.
20. Mark the release as, well, released in JIRA and create the next release in JIRA, if it's not already there.
21. Send out an ANNOUNCE email to the appropriate lists (see below) to announce the release!

Prerequisites for doing a release

Apache Nexus Account and Local Configuration

See http://www.apache.org/dev/publishing-maven-artifacts.html for the details. You should already have a Nexus account created for you automatically. If you aren't able to log in to https://repository.apache.org, and you're a jclouds committer, you should probably open a JIRA with INFRA.

GPG Signing Key

- Make sure you've got a GPG signing key set up. See http://www.apache.org/dev/release-signing.html for details.
- Associate your key with your ASF account by adding the fingerprint at https://id.apache.org/.
- At least for now, also make sure your key is in people.apache.org:/www/www.apache.org/dist/jclouds/KEYS.
- If/when you can, get linked into the Apache Web of Trust.
- Optionally, you can also add your GPG passphrase to your ~/.m2/settings.xml, or enable gpg-agent for the maven-gpg-plugin, by doing something like this:

```
<profile>
  <id>apache-release</id>
  <properties>
    <gpg.useagent>true</gpg.useagent>
    <gpg.passphrase>[GPG_PASSWORD]</gpg.passphrase>
  </properties>
</profile>
```

- But remember that this is not particularly secure, so you probably should just suck it up and enter your passphrase when you kick off the build, though you can leave gpg.useagent set to true to avoid having to re-enter it for each repo.

Mailing Lists

You will need to be subscribed to all mailing lists to which announcements need to be sent.

- dev@jclouds.a.o
- user@jclouds.a.o
- announce@a.o

Cutting an RC

Running the RC build

1. Check that your Git username and email are set to match your Apache account.
2. Run mvn --version and verify that you are using a Java 7 JDK for 1.9.x or newer releases.

   a. on OS X
      export JAVA_HOME="/Library/Java/JavaVirtualMachines/jdk1.7.x_yz.jdk/Contents/Home/"
3. Give your build more memory. e.g.
   a. export MAVEN_OPTS="-Xms512m -Xmx1024m -XX:PermSize=256m -XX:MaxPermSize=1024m"
4. Checkout https://dist.apache.org/repos/dist/dev/jclouds
   a. Invoke the **prepare_release.sh** script as follows:
      i. ./prepare_release.sh -r 1.9.0 -n 1.9.1-SNAPSHOT -b 1.9.x (where the "-r" parameter defines the release version, "-n" the next snapshot and "-b" the branch to be released, i.e. 2.0.x or `master`).
   b. The script will clone all repos, update all pom.xml files to the release version, create and push the tags, update the release branch to the next snapshot and create the staging repository. It will also take care of computing the RC version based on the existing tags, so you don't have to care about that.
   c. <u>Note that we only push the tags and not the branch with the next development SNAPSHOT already set</u>. This will make things easier if anything goes wrong and we have to start a new release candidate.
5. Once it finished, go to https://repository.apache.org, log in, and go to Staging Repositories. Find the open org.apache.jclouds repository, which your builds have been uploading to. Select it and close it. That'll take a little while. Then copy the URL for the repo - you'll need that for the VOTE thread.
6. Copy the tag references and hashes for each repo from the script output. So you can include them in the vote thread.

Starting the Vote

1. Checkout https://dist.apache.org/repos/dist/dev/jclouds. Create a new directory for the RC, i.e., 1.9.0-rc1.
2. Go to the candidate directory and run the following, substituting the correct version and the now-closed staging repository accordingly. This will copy down all the source tarballs, their signatures and their checksums.
   a. ./fetch_jclouds_rc.sh 1.9.0 https://repository.apache.org/content/repositories/orgapachejclouds-016 1.9.0-rc1/
3. It's not a bad idea to try taking each of the tarballs, blowing them up and then building, with a standard mvn clean install. Remember that RAT checks are done as part of the Maven build, so you don't need to run those separately.
4. svn add the new directory and svn rm old RCs (i.e., any older 1.9.x RC). svn ci, and use that link under https://dist.apache.org/repos/dist/dev/jclouds/ for the release vote email.
5. Get the JIRA release notes for the release by going to https://issues.apache.org/jira/secure/ConfigureReleaseNote.jspa?projectId=12314430 and choosing HTML and the version you're releasing. Save that link for inclusion in the vote/discuss emails.
6. Send two emails to dev@jclouds.apache.org, a [VOTE] thread and a [DISCUSS] thread. Set an end time for the vote, generally 72 hours from when you're sending it (but it's generally considerate to round up, and not a bad idea to give an extra day if the release vote window goes over the weekend). Example templates for the emails are below. Replace links, versions, RC numbers, release note link, etc.

   a. [VOTE] email:

```
Subject: [VOTE] Release Apache jclouds 1.6.3 RC3
Body:
Hello,

This is the third release candidate for Apache jclouds 1.6.3.

Please use the separate [DISCUSS] thread for anything but votes.

It fixes the following issues:
https://issues.apache.org/jira/secure/ReleaseNote.jspa?version=12324412&styleName=Html&projectId=12314430

*** Please download, test and vote by Saturday, June 15th, 09:00 PDT / 12:00 EDT / 18:00 CET.

Note that we are voting upon the source (tag), binaries are provided for convenience.

Source and binary files:
https://dist.apache.org/repos/dist/dev/jclouds/jclouds-1.6.3-candidate-3

Maven staging repo:
https://repository.apache.org/content/repositories/orgapachejclouds-016

The tags to be voted upon:
- jclouds - https://git-wip-us.apache.org/repos/asf?p=jclouds.git;a=tag;
h=8ed5571b0ecf7d79c64bc16642264684882f0311 (SHA-1: 079f9f0bcc9a4374cc73692fc16c5423b70fecbe)
- jclouds-labs - https://git-wip-us.apache.org/repos/asf?p=jclouds-labs.git;a=tag;
h=54b084e474e0c22b539ed7cce0b0ebbc82b310cb (SHA-1: da9de656e7f9ae9706a287a45ecfe60e79833ae9)
- jclouds-labs-aws - https://git-wip-us.apache.org/repos/asf?p=jclouds-labs-aws.git;a=tag;
h=54b084e474e0c22b539ed7cce0b0ebbc82b310cb (SHA-1: da9de656e7f9ae9706a287a45ecfe60e79833ae9)
- jclouds-labs-google - https://git-wip-us.apache.org/repos/asf?p=jclouds-labs-google.git;a=tag;
h=54b084e474e0c22b539ed7cce0b0ebbc82b310cb (SHA-1: da9de656e7f9ae9706a287a45ecfe60e79833ae9)
- jclouds-labs-openstack - https://git-wip-us.apache.org/repos/asf?p=jclouds-labs-openstack.git;a=tag;
h=54b084e474e0c22b539ed7cce0b0ebbc82b310cb (SHA-1: da9de656e7f9ae9706a287a45ecfe60e79833ae9)
- jclouds-karaf - https://git-wip-us.apache.org/repos/asf?p=jclouds-karaf.git;a=tag;
h=9e1f0d14285c8edeb35499f313aa7dbfab4a86f6 (SHA-1: 4b515ee6bf7b2bcc25747b75f7ccd87e4768f66a)
- jclouds-cli - https://git-wip-us.apache.org/repos/asf?p=jclouds-cli.git;a=tag;
h=2e3575f56de2bf67469782d94d72ce8e621ddda5 (SHA-1: adfe2e213bb0121d1faa6b7c135c092b70a93504)

jclouds KEYS file containing PGP keys we use to sign the release:
http://www.apache.org/dist/jclouds/KEYS

[ ] +1
[ ] 0  (explain why)
[ ] -1 (explain why)
```

- [DISCUSS] email

```
Subject: [DISCUSS] Release Apache jclouds 1.6.3 RC3
Body:
This thread is for discussion of the third release candidate for Apache jclouds 1.6.3.

Please use this thread for discussion of issues uncovered in the RC, questions you may have about the RC, etc.

If you want to help to validate the release, you'll find a set of scripts and the corresponding instructions here:
https://cwiki.apache.org/confluence/display/JCLOUDS/Validate+a+Release

You can also go run live tests for your preferred providers and post the results here, or use the projects in the
jclouds-examples repo, such as the "compute-basics" to rapidly smoke test the providers you are interested in:
https://github.com/apache/jclouds-examples


Thank you.
```

Canceling the PMC Vote

1. If, for some reason (e.g. a -1 that is not resolved), the vote needs to be canceled:
2. Reply to the [VOTE] thread on dev@jclouds.apache.org announcing the cancellation of the vote. Modify the template as appropriate.

```
Subject: [CANCEL][VOTE] Release Apache jclouds 1.6.3 RC3
Body:
This vote is being cancelled due to issues uncovered during validation. A new release candidate will be created
once these issues have been addressed. Thank you.
```

1. Prepare the release branch for the next RC:

   a. If the changes to the release branch were not pushed (as suggested in the release steps), then just reset the branch: `git checkout <release branch> && git fetch origin && git reset --hard origin/<release branch>`
   b. If the changes to the release branch were pushed, then Revert (using git revert) the release commits and push them to the remote repos.
2. If applicable, commit any additional changes required to resolved the issues identified, and push these too.
3. Drop the staging repository (take care! drop it; don't release it!)
4. Respin

Finishing the PMC Vote

1. If, after the voting window is closed, there are at least three binding +1s and no binding -1s, the PMC has signed off on the release and it can, well, be released.
2. Reply to the [VOTE] thread on dev@jclouds.apache.org announcing the closure of the vote, regardless of its outcome, and give the vote counts. Modify the template as appropriate.

```
Subject: [RESULT][VOTE] Release Apache jclouds 1.6.3 RC3
Body:
The vote is now closed, and with X binding +1s, we're ready to release/this vote has not passed.
```

1. Assuming the vote has passed, it's time to release!

Actually Releasing

1. First, in each of the repos, create and push a new tag, like jclouds-1.6.3, pointing at the same commit as the RC tag we included in the vote emails. Push those tags to the remote.
   a. git tag rel/jclouds-1.9.0 rel/jclouds-1.9.0-rc1
   b. git push origin rel/jclouds-1.9.0
2. Push the release branch. It contains the commit that updates all versions to the next SNAPSHOT and all *jclouds.version* properties

   a. git push origin 1.9.x
3. Checkout https://dist.apache.org/repos/dist/release/jclouds. Create a new directory for the release, i.e., 1.9.0. Copy the release artifacts from the RC into that directory. Point the stable symlink to that new directory. svn add the new directory and the stable symlink, and svn rm any old releases (i.e., any older 1.9.x releases) - these are archived on archive.apache.org, but shouldn't be in the primary dist area. svn ci - your added files should show up on http://www.apache.org/dist/jclouds shortly, but will take a while to propagate to the various mirrors.
4. While waiting, go to https://repository.apache.org, log in, and go to Staging Repositories. Select the staging repo for the RC, and click Release - this will move the artifacts into the releases repo, and from there they'll be synced to Maven Central as well.
5. If you've got admin access to JIRA, mark the version you just released as, well, released, and if the next version in the train doesn't already exist, create it. If you're not in the JIRA admins for jclouds, drop an email to the dev list asking for someone to do this.
6. Create the release notes in the jclouds main site and update the DOAP file to include the new release there.
7. Once the release bits have shown up on the mirrors, send the following email, updating the content accordingly, to announce@apache.org, dev@jclouds.apache.org and user@jclouds.apache.org:

```
Subject: [ANNOUNCE] Apache jclouds 2.1.1 released
Body:
The Apache jclouds team is pleased to announce the release of jclouds 2.1.1.

Apache jclouds is an open source multi-cloud toolkit for the Java platform that gives you the freedom to create
applications that are portable across clouds while giving you full control to use cloud-specific features.

The source archives for the release are available here:
https://jclouds.apache.org/start/install/

The Maven artifacts for the release are available in Maven Central, under the org.apache.jclouds group ID.

The release notes are available here:
https://jclouds.apache.org/releasenotes/2.1.1/

We welcome your help and feedback. For more information on how to report problems, and to get involved, visit the
project website at:
https://jclouds.apache.org/

The Apache jclouds Team
```