ReleaseCreation

This page is a brain dump of the various items needed for a release.

DRAFT

Requirements

- a Java JDK in according with target JDK (Java 8 for JMeter 5.0)
- a recent Ant binary (1.10.5+)
- subversion (svn) client installed (version 1.9+); the command-line utility svnmucc is needed (it should be included in the svn client installation (svn-tools dir), but please check)
- a text editor
- hostname must resolve to ip address (not loopback)
- Internet access

Preparation

- request Bugzilla version update (on INFRA or if you are JMeter committer, ask for sufficient Bugzilla karma to do this yourself)
- ensure changes.xml is up to date. If necessary, commit after update.
- ensure unit tests pass
- do not update the version in build.xml yet; trunk should remain a SNAPSHOT
- check out SVN trunk into a clean workspace, for example (assuming current trunk is 5.0-SNAPSHOT in preparation for releasing 5.0):

```
svn co https://svn.apache.org/repos/asf/jmeter/trunk jmeter_v5_0_RC1
```

- Move to the new workspace:

```
cd jmeter_v5_0_RC1
```

- update the version / year in the new workspace: JMeterVersion.java (version and year) and NOTICE (year)
  - ./src/core/org/apache/jmeter/util/JMeterVersion.java:
    - section: *private static final String VERSION = "5.0";*
    - section: *private static final String COPYRIGHT = "Copyright (c) 1998-2019 The Apache Software Foundation";*
  - ./NOTICE:
    - section: *Copyright 1998-2019 The Apache Software Foundation*
- update xdocs/download_jmeter.xml to the new version
  - section: <!ENTITY release '5.0'>
- Download external jars

```
ant download_jars
```

- build site documents, having first cleaned out existing files (so obsolete files will be removed):
  - TODO should this be done on trunk, before creating the clean checkout? Are there any references to the current version in the docs?

```
ant clean-docs
ant docs-site
ant docs-api -Djmeter.version=5.0
```

- check if you need to add some new files or remove some deleted files from docs directory (using svn status command)

```
svn status
```

- commit docs changes (CHECK IF THEY HAVE NEW PAGES/FILES : make a svn add before commit)

```
svn commit -m "Prepare next release 5.0: update docs tree and overview (RC1)"
```

- make a last SVN update to update revision number for your workspace

```
svn update
```

- edit **now** the version for build.xml (remove -SNAPSHOT)
    - ./build.xml:
        - section: *<property name="jmeter.version" value="5.0"/>*
- create the new RC tag, e.g. v5_0_RC1 from the workspace

```
svn copy -m "Tag for pre-release" . https://svn.apache.org/repos/asf/jmeter/tags/v5_0_RC1
```

- this will create the tag from the contents of the workspace. Only the updated files will be shown in the commit message; most files will be shown as being copied from trunk.
    - N.B. Tags must be immutable, i.e. must not be changed once created. A tag can be deleted if it is no longer needed, but must not be recreated.
        - If there is a problem with the contents of an RC tag, create a new tag, for example v3_2_RC2
        - If the tag is part of an ongoing release vote, do not delete it whilst the vote is ongoing, even if the vote is now using a later tag. It can be useful to be able to compare the contents of tags as part of the voting process.
        - Once the release vote is over, intermediate tags are no longer needed, and can be deleted.
        - If the tag has not been used in a release vote, it can be deleted immediately.

- Switch to the new tag

```
svn switch https://svn.apache.org/repos/asf/jmeter/tags/v3_2_RC1
```

- build code to make a testing binary with this ant task:

```
ant -Djmeter.version=5.0 -Duser.name=milamber@apache.org distribution
```

- test the new binaries (move/extract binaries in a temporary directory outside the workspace)
    - GUI and non-GUI using host with graphic capability
    - non-GUI only using host with no graphic capability
    - ideally test on minimum supported Java and latest Java - especially for GUI

Generate Release

- If tests are ok, from the workspace jmeter_v3_2_RC1
    - (no need to fetch the tag again, as we already have a clean workspace)
- sign distribution files and maven files via ant tasks:

```
ant _dist_maven -Djmeter.version=5.0
ant sign_dist -Dgpg.keyname=0612B399 -Dgpg.secretKeyring=~/.gnupg/secring.gpg
```

- Upload Maven files to Maven repository:

```
ant maven_upload -Djmeter.version=5.0 -DrepoType=releases
```

- Go to Apache Repository
    - Section **Staging Repositories**
        - Check the RC box, add click on Close button (see Closing a staging repository)
    - You will receive a email with JMeter Maven Repo URLs
        - The main URL must be added to email vote.
- Generate RAT report (example on Linux with Java)

```
cd dist/
java -jar $RAT_HOME/apache-rat-0.12.jar ./apache-jmeter-5.0.tgz > rat-report-jmeter-5.0RC1.txt
unix2dos rat-report-jmeter-5.0RC1.txt
```

Upload release artifacts to staging directory

JMeter releases are now published using svnpubsub.

For release votes, the files are uploaded to the dev/jmeter tree at https://dist.apache.org/repos/dist/dev/jmeter/

If the vote succeeds, they are renamed to the release directory at https://dist.apache.org/repos/dist/release/jmeter/ which is automatically published to the ASF mirror system.

```
ant RC_upload -DRC=RCn -DrepoType=live [-Dasf.ldap.password=secret [-Dasf.ldap.username=user] ]
```

The files should now be available under https://dist.apache.org/repos/dist/dev/jmeter/

Create docs branch

The site documentation now needs to be copied to the appropriate branch directory from the tag.
[Note: if the vote fails, the branch will need to be removed and recreated]

```
svn cp -m"Create docs branch" https://svn.apache.org/repos/asf/jmeter/tags/v3_2_RC1 https://svn.apache.org/repos
/asf/jmeter/branches/docs-5.0/
```

The Javadocs are not yet part of SVN at this point. For simplicity when reviewing the release candidate, they should be added to the docs branch here:

```
svn import -m "Add Javadocs" docs/api https://svn.apache.org/repos/asf/jmeter/branches/docs-5.0/docs/api
```

It would be possible to add them directly to the live SVN directory later, but they need to be available for review, so it's easiest to create the branches/docs-m.n/docs tree as an exact copy of what will be released.

Checks to do before sending vote email

- review RAT report
- See ReleaseChecking for information on what to check

Vote on release

- send round [VOTE] e-mail to dev

- Wait for any feedback (at least 72 hours); address any issues raised, if necessary by creating another release tag
- If at least 3 PMC votes, then continue

After VOTE success

- send round [VOTE] [RESULT] e-mail to same mailing lists

- Copy the RC tag to the release tag

```
svn copy https://svn.apache.org/repos/asf/jmeter/tags/v5_0_RC1 https://svn.apache.org/repos/asf/jmeter/tags
/v5_0 -m "Vote succeeded, create the 5.0 tag"
```

Move RC dist files to release

This is done using svnmnucc. It moves the files from the staging director https://dist.apache.org/repos/dist/dev/jmeter/ to the live are at https://dist.apache.org/repos/dist/release/jmeter/

```
ant RC_publish -DRC=RCn -DrepoType=live [-Dasf.ldap.password=secret [-Dasf.ldap.username=user] ]
```

Update main distribution files

- Checkout the top-level of the live dist project on your computer:

```
svn co --depth files  https://dist.apache.org/repos/dist/release/jmeter/ dist-jmeter
```

- Edit HEADER.html to modify with new version number
- If necessary, add your GPG public key to KEYS
- If necessary, update the META file with your GPG key id (if you act as the release manager for the first time. Please visit https://checker.apache.org/doc/README.html)
    - The META file needs to be signed by the PMC Chair of project with this command:

```
gpg -u emailOfPmcChairJMeter@apache.org --armor --output META.asc --detach-sig META
```

- To verify the good signature, use this command:

```
$ gpg --verify META.asc META

gpg: Signature made mar. 12 sept. 2017 18:05:19 WEST
gpg:                using RSA key C4923F9ABFB2F1A06F08E88BAC214CAA0612B399
gpg:                issuer "milamber@apache.org"
gpg: Good signature from "Milamber (ASF) <milamber@apache.org>" [ultimate]
gpg:                 aka "Milamber (Milamberspace) <milamberspace@gmail.com>" [ultimate]
```

- Commit with svn

```
svn commit -m "JMeter 5.0 release - Prepare update site" dist-jmeter
```

Maven Release

- Go to Apache Repository
- Section **Staging Repositories**
    - Check the RC box, add click on Release button (see: Release a staging repository)
    - Comment: "JMeter 5.0 release"
- Maven Central will catch up the new release in next sync (check: ga)

Wait for the update mirrors and central maven

- **wait a day** or so for mirrors to catch up distribution files and Maven Central update. Note: the mirrors list can be checked Here

Update Doap file

- Get latest Doap file from SVN.

```
svn co https://svn.apache.org/repos/asf/jmeter/doap
```

- update doap_JMeter.rdf to add the new release
- and commit

```
svn commit -m "JMeter 5.0 release"
```

Publish JMeter site

This is now done using svnmucc. Create a script - for example "updatesite.txt" - containing the following:

```
rm
https://svn.apache.org/repos/asf/jmeter/site

cp
HEAD
https://svn.apache.org/repos/asf/jmeter/branches/docs-5.0/docs
https://svn.apache.org/repos/asf/jmeter/site

cp HEAD
https://svn.apache.org/repos/asf/jmeter/doap/doap_JMeter.rdf
https://svn.apache.org/repos/asf/jmeter/site/doap_JMeter.rdf
```

Change the /docs-5.0/ path segment above as appropriate!

Now use svnmucc to update the live site:

```
svnmucc -m "Update website for the new release JMeter 5.0" -X updatesite.txt
```

The SvnPubSub system will update the live site within a few seconds.

Note: it's important that the existing contents are removed and replaced, rather than just copying the current site over. This ensures that obsolete files are removed.

Updating the current JMeter site

The branches/docs-m.n SVN tree is created from trunk when the release is created. Subsequent updates to trunk documents may not be appropriate to the current release - e.g. changes to test elements

- so any necessary changes to the site should be made to the branch (and trunk if appropriate).

Then rebuild the site docs (not the Javadocs!) - `ant docs-site -Ddocs.force=true` - and update the SVN branch with the changed files.

The site can then be published as above.

If only a few changes are necessary, these can be updated using normal SVN commit.

Announce

Make sure: JMeter website is update AND download mirrors too.

- update Wiki version details
- send [ANNOUNCE] e-mail to user@jmeter.a.o, dev@jmeter.a.o
  - See ReleaseAnnounceEmailTemplate
  - Double check all URL in email announce!
- also copy to announce AT ao (has to be sent from an ao e-mail address)
- N.B. to track bounces and unsubscribes, it may be a good idea to use a separate mail to Announce

Update reporter.apache.org

Add release data (version and date) to reporter.apache.org - https://reporter.apache.org/addrelease.html?jmeter

Tidy up

- a few days after the release has been announced, remove any superseded releases from the binaries and source directories under https://dist.apache.org/repos/dist/release/jmeter/ using the following command:

```
ant release_delete -Djmeter.old.version=4.0 -DrepoType=live [-Dasf.ldap.password=secret [-Dasf.ldap.
usermame=user] ]
```