Geronimo Eclipse Plugin Release Process

Steps

Before starting the process, make sure these work below has been done:

- All version numbers in GEP are correct for issuing "mvn release:prepare", including plugin version in MANIFEST.MF
- Assure all the dependency version number is synchronized with Geronimo server
- With tool Apache RAT, scan all the source code to make sure Apache license headers have been added into neccessarily. The header is neccessary for .java, .xsd,.xml,.html and .properties, which contain functional code or configuration information.

1. ✅ The plan is to sync Geronimo Eclipse Plugin (GEP) releases with those of the Geronimo server. The goal is to release GEP as soon as possible after the release of the corresponding server.

   > ⚠ **Note:** These notes are written for release 2.2. Change the version number accordingly for your release!

2. ✅ Pick up JIRA list of fixed bugs and new features from Release Notes and defer unresolved JIRAs
3. ✅ Update the release notes. They are the file PLUGIN_RELEASE-NOTES-2.2.txt in trunk. Copy the JIRA list from Step 2 into it.
4. ✅ Remove snapshot from version number of dependency. Eg. geronimo-kernel-2.2-SNAPSHOT.jar in MANIFEST.MF of runtime.v22.
5. ✅ Build the new **branches** tree that will soon be released
6. ✅ Ensure that your GPG public key is in the /www/www.apache.org/dist/geronimo/KEYS file and KEYS in svn server before publishing anything to the distribution location
7. ✅ Prepare the release with maven release plugin. Refer to http://maven.apache.org/developers/release/apache-release.html
   a. mvn release:prepare -Pallsubproject
   b. mvn release:perform -Pallsubproject

   > ⚠ **Note:** Profile allsubproject is added since GEP 2.1.7 and 2.2.1 to include testsuite module during release but exclude the testsuite execution

8. ✅ Set up a staging site for verifying update site at people.apache.org. Eg. http://people.apache.org/builds/geronimo/eclipse/updates/, this staging site locates on people.apache.org here

   ```
   www/people.apache.org/builds/geronimo/eclipse/updates
   ```

   a. Extract geronimo-eclipse-plugin-2.2-updatesite.zip and calculate check sum for each files
   b. Upload all the files into update site
   c. Upload deployable.zip and updatesite.zip in a version sub directory, such as www/people.apache.org/builds/geronimo/eclipse/2.2.1
9. ✅ Test the staging site by downloading the adapter from it
10. ✅ Create the How to install GEP v2.2 wiki page. Be sure this includes the pre-reqs. The idea is to have a single page with this release-specific information. In the vote note below, refer to this page, with overrides for the staging site URL. The idea here is to create the install page before hand, and use it for the voting, so that folks will review the doc we'll go live with when the vote passes. There is a How to install GEP page, with children of the various versions. These are for folks who may want to install old versions. When the vote passes, move all it's children to the v2.2 page, then rename the How to install GEP page to the v2.2.1 page. Make it a child of the v2.1.2 page. Then, rename the v2.1.2 page to the generic How to install GEP page.
11. ✅ Close the staging repository in maven staging site in Sonatype Nexus Maven Repository Manager so people reviewing the vote can see your artifacts.
12. ✅ Make sure to start a **[VOTE]** thread on the Geronimo dev list. In the vote notice, please include:
    - The precise names and versions being voted on (e.g. geronimo-eclipse-plugin-2.2-deployable.zip and geronimo-eclipse-plugin-2.2-updatesite.zip)

      > ⊘ **Tips**
      >
      > 1. Be certain to use the Release Candidate (RC) acronym in the name as it will not likely pass on the first vote
      > 2. Be certain to use the same version numbers as the Geronimo Server (e.g., 2.2)

    - URL of the staging site (with instructions on how to use it)
    - Eclipse requirements if any (with instructions on how to download them and from where)

      > ⊘ **Tip**
      >
      > Be certain to get enough binding votes according to the Apache Foundation requirements, which in general stipulates 3 PMC +1 votes (see http://www.apache.org/foundation/voting.html#ReleaseVotes for more details)

13. ✅ After the vote has been approved:
    - ✅ Publish the results on a **[RESULTS]** reply to the original **[VOTE]** thread on the Geronimo dev list
    - ✅ Promote artifacts in maven staging site. You may find them on http://repository.apache.org/ (The artifacts are promoted immediately.)

- ✅ Publish the staging site, including the deployable.zip and updatesite.zip files, and the release notes to the **Geronimo Eclipse distribution** location on people.apache.org here:

```
/www/www.apache.org/dist/geronimo/eclipse
```

- ✅ Ensure that the version subdirectory is used (e.g., 2.1.2)
- ✅ Ensure that the file(s) you copy to the distribution directory have 0775 dir permission and a 0644 file permission set on them
- ✅ Ensure that the file(s) you copy to the distribution directory have the RCx acronym removed from the filename

14. ✅ Ensure that the corresponding **Geronimo Eclipse distribution** website is updated automatically in the next 24 hours and properly with the deployable.zip file and the updatesite.zip file

15. ✅ Ensure that the file(s) you copy have the proper checksum files as well (i.e., .asc, .md5, .sha) using these gpg commands:

```
gpg --print-md MD5 [fileName] > [fileName].md5
gpg --print-md SHA1 [fileName] > [fileName].sha
gpg --armor --output [fileName].asc --detach-sig [fileName]
```

NOTE: To generate keys, you need to notice the rules here http://www.apache.org/dev/openpgp.html and http://www.apache.org/dev/key-transition.html
NOTE: Maven release plugin will generate the signature and checksum for all jar files, but for other files, you need to generate them manually.
16. ✅ Update the **Geronimo Eclipse Update Site** with the contents of the updatesite.zip:

a.
```
1. /www/www.apache.org/dist/geronimo/eclipse/updates/features
2. /www/www.apache.org/dist/geronimo/eclipse/updates/plugins
3. /www/www.apache.org/dist/geronimo/eclipse/updates/site.xml
```

> ⊙ **Warning**
>
> Never delete old releases from the update site - always append to it !! Likewise, do not create a new update site as this will break everyone who has this current site configured, and WTP which uses the update site for the downloadable server adapter support.

> ⊙ **Warning**
>
> If you use FTP to transfer the files to the *Geronimo Eclipse Update Site *make sure to transfer everything in binary mode. Otherwise, the checksum files may not transfer correctly (especially if they are transferred in text mode).

b. ✅ Ensure that the file(s) you copy to the update site directories have 0755 dir permission and a 0644 file permisison set on them
17. ✅ Ensure that corresponding **Geronimo Eclipse Update Site** website (http://apache.org/dist/geronimo/eclipse/updates/) is updated automatically in the next 24 hours with the contents of the updatesite.zip file.
18. ✅ Update the Apache Geronimo Development Tools Subproject web site. Please include:
- Updated Release Notes
- All Eclipse and WTP prerequisites and download instructions
- Links to recent, pertinent, and relevant presentations (e.g., Shiva's, EclipseWorld, look for others, etc)

19. ✅ Administer the GERONIMODEVTOOLS JIRA project to update the released and unreleased versions. Remember to mark the version as released.

20. ✅ Update the Geronimo Home Page with a News item that we have an updated Geronimo Eclipse Plugin available. Announce the availability in user@geronimo.apache.org, dev@geronimo.apache.org and announce@apache.org.

21. ✅ Final step is to update the Geronimo Release Roadmaps with the release date.
22. ✅ Update the version number of dependency and devtools in trunk for next release version. Begin the development of next release.
a. ✅ Rename PLUGIN_RELEASE-NOTES-2.1.2.txt to PLUGIN_RELEASE-NOTES-2.1.3.txt.
b. ✅ Update version in plugin.xml from 2.1.2 to 2.1.3
c. ✅ Update version in MANIFEST.MF from 2.1.2 to 2.1.3