

# Creating a JaxMe release

This page provides information about the steps required to create a JaxMe release.

## 1. Creating a release

1. Update the "version" property in "build.xml".
2. Make sure that you have the file jimi-1.0.jar in src/documentation/lib. This file is required for generating PDF's including PNG images. Unfortunately it cannot be added to the repository, due to licensing issues. See the Forrest FAQ on how to obtain the file.
3. Update the file src/documentation/content/xdocs/news.xml in the ws-jaxme repository.
4. Make sure that you have gpg or a similar tool installed. The easiest way is via the CygWin installer (Windows) or using a RPM package provided by your vendor (Linux). If you haven't yet created a secret key, do so using "gpg --gen-key". Also, create a KEYS file using "gpg --list-sigs jochen@apache.org" >>KEYS" "gpg --armor --export jochen@apache.org" >> KEYS. Add the contents of the KEYS file to "etc/KEYS" in the CVS directory.
5. In src/documentation/content/xdocs/site.xml: Make sure, that the section build.online.site=true is commented out and the section build.online.site=false is active.
6. Create the distribution files using "ant dist". This will create, for example, four files called "ws-jaxme-0.3-bin.tar.gz", "ws-jaxme-0.3-bin.tar.gz.MD5", "ws-jaxme-0.3-src.tar.gz", and "ws-jaxme-0.3-src.tar.gz.MD5". The extension .MD5 indicates that these are no true distribution files, but simply contain an MD5 checksum over the others.  
It is quite possible, that you need to do this using Java 1.4 and not 5.0: As of Java 5.0.2, there is a bug in the Xalan version, causing an ArrayIndexOutOfBoundsException in a stylesheet transformation.
7. Sign all distribution files, creating additional files with the same names and an ending ".asc". For example: "gpg --output ws-jaxme-0.3-bin.tar.gz.asc --detach-sig --armor ws-jaxme-0.3-bin.tar.gz"
8. Upload all the files to minotaur.apache.org using scp. For example: "scp etc/KEYS dist/ws-jaxme-0.3\* jochen@minotaur.apache.org:"
9. Move the files to the directory /www/www.apache.org/dist/ws/jaxme and ensure proper permissions: "chgrp ws KEYS ws-jaxme-0.3\*" "chmod 664 KEYS ws-jaxme-0.3\*" "mv KEYS ws-jaxme-0.3\* /www/www.apache.org/dist/ws/jaxme"
10. Update the file "/www/www.apache.org/dist/ws/jaxme/README.html".
11. Upload numbered versions of the jars released to the Apache Java Repository together

with md5 sums and signatures. At the moment, this should be done by uploading to your home directory on [www.apache.org](http://www.apache.org) and then moving them into the [/www.apache.org/dist/java-repository/jaxme/jars/](http://www.apache.org/dist/java-repository/jaxme/jars/). Finally, update the SNAPSHOT symlinks in that directory. Note that the processes for the repository are still under development so it would probably be a good idea to contact the repository team (subscribe to repository at [apache.org](http://apache.org)).

12. In `src/documentation/content/xdocs/downloads.ehtml`: Add the new version and the checksums.
13. Build a new version of the online site. See the separate section below.
14. Wait one day, until the Apache mirrors are in sync.
15. Update the files `mirrors.ehtml` and `news.xml` in the directory `src/documentation/content/xdocs` of the ws-site repository.
16. Announce the new release on [FreshMeat](#). There are three projects that might need notifications: JaxMe, JaxMeJS, and JaxMeXS. Creating an announcement on freshmeat requires that you have an account and be administrator for the respective projects. If you aren't yet, contact [Jochen Wiedmann](#).
17. Announce the new release on all Apache appropriate mailing lists. This should include the JaxMe list and the announcement lists for the WebServices and Xml projects. Note that it is important to include a description of JaxMe when posting announcements to project level lists. Note that you should post from your apache email address and you will need to subscribe.

## **2. Creating the online site**

The online site is created like this:

1. In `src/documentation/content/xdocs/site.xml`: Make sure, that the section `build.online.site=false` is commented out and the section `build.online.site=true` is active.
2. Run the build script: `"ant -Dbuild.online.site=true clean Docs.all"`.
3. To export the contents of `build/docs/build/site` to the ws-site directory, extract the CVS module `ws-site/targets/jaxme` (for example, to `/tmp/jaxme` and run `"ant -Dws.site.dir=/tmp/jaxme export.site"`. After checking in this directory, everything should work well.