

Release Notes

Table of contents

1 Release 1.3.1.....	2
2 Release 1.3.0.....	2
3 Release 1.2.1.....	2
4 Release 1.2.0.....	2
5 Release 1.1.0.....	2
6 Release 1.0.0.....	3
7 Release 0.99.0.....	3
8 Release 0.98.0.....	3
9 Release 0.97.0.....	3
10 Release 0.96.0.....	3
11 Release 0.95.0.....	4

Note:

For a detailed list of changes, have a look at the [ChangeLog](#).

1. Release 1.3.1

- Bugfix for a `NullPointerException` in the `Info`-class. The parser could not deal with empty lines within image-properties.

2. Release 1.3.0

- Added support for ImageMagick's compare-command
- Update of ImageMagick's commandline options
- Rework of the `Info` class: the implementation prior to 1.3.0 had a number of flaws and shortcomings. If the image contained more than one scene, the constructor requesting basic attributes would fill the attributes from the *first* scene, whereas the constructor requesting full attributes would fill the attributes from the *last* scene. In addition, `Info.getImageWidth()` and `Info.getImageHeight()` didn't return the image width and height, but the page width and height (although for most images this is identical). With the new implementation, these bugs were fixed and all attributes of all scenes are available. Please read the API-documentation for details.
- Support more image-formats for `BufferedImages`. If you installed the JAI-extensions you gain support for CMYK-images through the JAI-tiff-plugin.

3. Release 1.2.1

- Documentation-update: im4java is available at maven-central

4. Release 1.2.0

- Code generation is much faster
- Improved `Info` class: simple interface to retrieve basic image-properties
- fixed a number of bugs

5. Release 1.1.0

- Support ImageMagick's read-modifier for image-placeholders
- Support [parallel processing](#) of images. This feature will efficiently use your multi-processor computer (note that this is mainly useful for desktop-applications). Note that the im4java-API for parallel processing is not yet considered as stable. This depends on feedback and bug-reports. I am not overly optimistic that I found and fixed all bugs and race-conditions.
- The interface `org.im4java.process.ProcessListener` is deprecated and is replaced with `org.im4java.process.ProcessEventListener`. Transition should be easy, since the interfaces are very similar. This change was

necessary to support parallel processing. The old interface will be removed with the first release after beginning of 2011.

- Fixed a serious bug in the implementation which could lead to deadlocks under Windows (thanks to Jaroslav Snajdr for pointing this out and providing a patch).
- The binary-distribution now contains two versions of the im4java-jar: one for JRE 1.6, one for 1.5.
- The source-distribution will again contain all generated java-source-files.

6. Release 1.0.0

- Relaunch of the project-web, adding a lot of documentation
- Refactored Test.java
- All tests now work under Windows
- Added script-generator
- Support the environment-variable IM4JAVA_TOOLPATH to set the searchpath for tools externally
- Removed generated files (including doc) from the source distribution
- Removed source from the binary distribution

7. Release 0.99.0

- Added support for defining search paths
- Minor fixes to interface-definition-files

8. Release 0.98.0

- Added support for exiftool
- Added support for dcraw
- minor fixes to interface-definition-files

9. Release 0.97.0

- Added support for ufraw
- Added support for asynchronous execution of commands
- Source-generation now supports commands with long-style (--foo=bar) options

10. Release 0.96.0

- Source-generation now supports more than one interface-definition-file
- Added support for GraphicsMagick. You can hard-code GM-usage, but if you stick to the common subset of IM and GM commands, you can switch to GM at runtime by setting the system-property im4java.useGM=true.
- Added support for jpegtran
- Added some utility-classes in org.im4java.process
- ImageCommand now sends stdout to System.out as default

- Incompatible change to IdentifyCmd: output is not captured by default anymore. To capture output, use the new class `org.im4java.process.ArrayListOutputConsumer`. See `org.im4java.test.Test.testIdentify()` for an example.

11. Release 0.95.0

- Initial release