im4java Reference

Table of contents

1. Configuration

You can configure the behaviour of the im4java-library with the following environment-variables and system-properties:

IM4JAVA_TOOLPATH (environment)
Default searchpath for commandline tools. Must use the platform-specific path-delimiter.

im4java.useGM (system-property)
If true, use GraphicsMagick instead of ImageMagick.

im4java.maxProcs (system-property)
The maximum number of asynchronous processes the `org.im4java.process.ProcessExecutor` will run concurrently. If unset or if the property has the value *auto*, the number returned by `Runtime.availableProcessors()` is used. Note that the ProcessExecutor will at least use one process.

2. Test Cases

The package `org.im4java.test` contains a growing number of test-cases, demonstrating the various features of the im4java-library. Here is a quick overview.

TestCase1
Simple use of convert

TestCase2
Operation and sub-operations

TestCase3
Using montage

TestCase4
The ChannelMixer-class

TestCase5
Using mogrify

TestCase6
Using identify

TestCase7
Using composite

TestCase8
Using the Info-class

TestCase9
The NoiseFilter-class

TestCase10
Piping

TestCase11
Dynamic operations

TestCase12
Reading BufferedImage
TestCase13
Writing BufferedImage
TestCase14
GraphicsMagick
TestCase15
Using jpegtran
TestCase16
Asynchronous execution. Basic version.
TestCase16a
Asynchronous execution. This version uses `Executors.newSingleThreadExecutor()` to aquire a `ExecutorService`.
TestCase16b
Asynchronous execution. This version uses a `ProcessExecutor` to run at most two processes in parallel.
TestCase17
Using ufraw-batch
TestCase18
Using exiftool
TestCase19
Using dcraw
TestCase20
Setting search PATHs
TestCase21
Parallel processes
TestCase22
Using the BatchConverter-class