



Java Image Processing Survival Guide



Geförderte Wohnungen
77 m² | 3 Zimmer
1040 Wien
990,-



Maisonette mit Terrasse, Loggia
90 m² | 3 Zimmer
1030 Wien
1.290,-



Super Ausblick Provisionsfrei!!
75 m² | 2 Zimmer
1030 Wien
1.100,-



Sanierte Wohnung
102 m² | 2 Zimmer
1050 Wien
1.388,-

Deine Suche

- Preis bis 1.200,-
- ab 65 m²
- Wien
- 3. Bezirk, Landstraße

Willst du diese Suche speichern und neue Anzeigen **per Email** erhalten?

[SUCHAGENT AKTIVIEREN](#)

Volltextsuche

[Detailsuche](#) [Hilfe zur Suche](#)

Bundesland/Region

- 3. Bezirk, Landstraße
- [Alle Unterkategorien](#)

Gesamtmiete in €

[Alle zeigen](#)

350 – 799 6

800 – 899 29

138 Mietwohnungen in Wien

[ZUR KARTENANSICHT](#)[1](#) [2](#) [3](#) [4](#) [5](#) [weiter](#)Sortiert: [Datum](#)

Schöne Altbauwohnung im Schubertturm

3 Zimmer, Altbau, Balkon, Einbauküche, Etagenheizung, Parkettböden, Kellerabteil

**70 m²** **1.194,-**

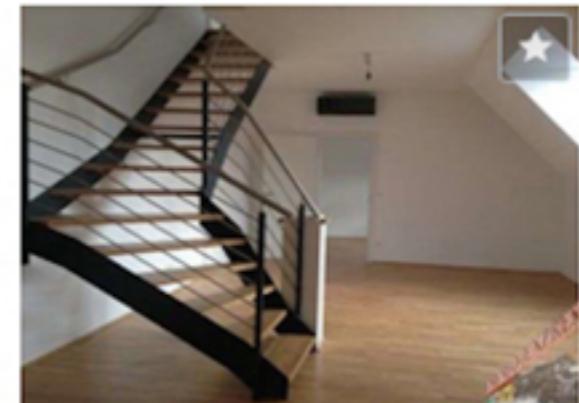
1030 Wien

69 m² **890,-**

1030 Wien

90 m² **900,-**

1030 Wien

**83 m²** **998,-**

1030 Wien

77 m² **1.099,-**

1030 Wien

102 m² **1.178,-**

1030 Wien

[Startseite](#) › [Immobilien](#) › [Wohnung mieten](#) › [Wien](#) › [Suchergebnis](#) › [Anzeige](#)

willhaben-Code: 123456899 | Zuletzt geändert: 23.07.2014

[◀ ZURÜCK](#)[WEITER ▶](#)

Wohnkomfort in Grünlage Nahe dem Hauptbahnhof mit Terrasse

1030 Wien, Landstraße 83m² 3 Zimmer Gesamtmiere € 998,- [ANZEIGE GEMERKT](#)

Kontaktdaten

**WOHNEN & WERT**

Firma Wohnen & Wert GmbH
Lehargasse 9/10, 1010 Wien
Name Ing. Anton Eibensteiner

[ANFRAGE SENDEN](#)

Telefon +43 1 596 60 20 41
Telefon 2 +43 699 17 55 66 41

[Weitere Anzeigen von diesem Anbieter](#)**€ 998,-**
Gesamtmiere

Nettomiete: € 738,71
Betriebskosten netto: € 168,56
USt. gesamt: € 90,73

Objektstandort

*“Siegfried, how difficult is it to
replace ImageMagick with a Java
library?”*



“Well, not too difficult”

The Customer's Approach

- ImageMagick native libraries
 - ▶ Handles image conversion and scaling
- Ghostscript for PDF preview generation
 - ▶ Invoked by ImageMagick
- JMagick to access ImageMagick libraries over Java Native Interface (JNI)

The Customer's Problems

- ImageMagick exceptions kills the JVM
 - ▶ Load balancer kills the next server
- JMagick is no longer maintained
 - ▶ Unable to set JPEG quality parameter
- Installing ImageMagick, JMagick, Ghostscript and friends can be tricky
 - ▶ Failed miserably for Mac OS X

Having Said That

- ImageMagick & Ghostscript are mature
- Handles many image formats and flavours
- Being used in production for eight years
- Handles 5.000.000 image uploads / month



**Does not look
so easy anymore**

We Need A Plan

- Java ImageIO for low-level image plumbing
- Java Advanced Imaging (JAI) for additional image formats
- Image scaling library to create previews and thumbnail images efficiently
- Apache PDFBox to create PDF previews

Java ImageIO

- *ImageReader* to create a *BufferedImage* from various image formats
 - ▶ JPEG, PNG, GIF & BMP
- *ImageWriter* to write a *BufferedImage* with various options
 - ▶ Setting image quality for JPEG
 - ▶ Set DPIs for PNG or JPEG

Java ImageIO

- Apply buffered image operations
 - ▶ Color conversion - grey-scaling
 - ▶ Affine transformation - scaling
 - ▶ Convolution operation - sharpening
- Provides a *Service Provider Interface (SPI)*
 - ▶ Adding additional image formats

Java Advanced Imaging

- Adds TIFF and JPEG2000 support
- Last stable release in 2006
- Libraries are not on Maven Central
- Many dead documentation links
- Consists actually of three libraries
 - ▶ `jai_imageio.jar` provides Java ImageIO integration

Image Scaling

- Many options when using Java 2D API
 - ▶ `Image.getScaledInstance()`
 - ▶ `Graphics.drawImage()`
 - ▶ `Graphics2D.drawImage()`
 - ▶ `BufferedImageOps`
 - ▶ `AffineTransforms`

Image Scaling Quality

- Scaling algorithm being used
 - ▶ Nearest Neighbour, Bilinear, Bicubic Interpolation, Lanczos Re-sampling, ...
- Image optimisations
 - ▶ Sharpening, Anti-aliasing, Image Auto-Correction
- JPEG quality settings

Image Scaling Libraries

- Open Source image scaling libraries
 - ▶ *imgscalr*
 - ▶ *thumbnailator*
 - ▶ *java-image-scaling*
- Libraries have similar options, performance and result image quality

Apache PDF Box

- PDF Document Creation
- PDF document manipulation
 - ▶ Merging
 - ▶ Splitting
- PDF text extraction
- PDF to image conversion
 - ▶ Uses ImageIO SPI under the hood

Apache PDF Box

```
List<BufferedImage> pdfToImage(  
    Object source, int from,  
    int to, int dpi)  
{  
    PDDocument pdDocument = loadPDDocument(source);  
    List<BufferedImage> result = new ArrayList<BufferedImage>();  
    List<PDPage> pages = pdDocument.getDocumentCatalog().getAllPages();  
  
    for (int i = from - 1; i < to && i < pages.size(); i++)  
    {  
        PDPage page = pages.get(i);  
        BufferedImage image = page.convertToImage(BufferedImage.TYPE_INT_RGB, dpi);  
        result.add(image);  
    }  
    return result;  
}
```

Hitting
Real-life
Test Data



What's Wrong?!



PNG



JPEG

Alpha-Channels



- Alpha channel stores transparency for GIF and PNG
- ImageIO uses a RGBA color model when saving as JPEG
- Resulting images have a red tint or are all black
- Need to convert to RGB using `Graphics2D.drawImage()`

Alpha-Channels

```
BufferedImage bufferedImage = createBufferedImage(sourceImageFile);
int width = bufferedImage.getWidth();
int height = bufferedImage.getHeight();
final int imageType = BufferedImage.TYPE_INT_RGB;
BufferedImage rgbBufferedImage = new BufferedImage(width, height, imageType);
Graphics2D graphics = rgbBufferedImage.createGraphics();
graphics.drawImage(bufferedImage, 0, 0, null);
graphics.dispose();
```

What's Wrong?!

```
if(imageSourceFile.length() > IMAGE_FILE_SIZE_LIMIT) {  
    throw new ImageConversionException("The image is too big ...");  
}
```

What's Wrong?!

```
if(imageSourceFile.length() > IMAGE_FILE_SIZE_LIMIT) {  
    throw new ImageConversionException("The image is too big ...");  
}
```

```
temp> ll preview.png  
-rw-rw-rw-@ 1 sgoeschl staff 44024 Apr 10 20:55 preview.png
```

```
java.lang.OutOfMemoryError: Java heap space  
at java.awt.image.DataBufferByte.<init>(DataBufferByte.java:76)  
at java.awt.image.MultiPixelPackedSampleModel.createDataBuffer(MultiPixelPackedSampleModel.java:204)  
at java.awt.image.Raster.createWritableRaster(Raster.java:941)  
at javax.imageio.ImageTypeSpecifier.createBufferedImage(ImageTypeSpecifier.java:1073)  
at javax.imageio.ImageReader.getDestination(ImageReader.java:2896)  
at com.sun.imageio.plugins.png.PNGImageReader.readImage.PNGImageReader.java:1280)  
at com.sun.imageio.plugins.png.PNGImageReader.read.PNGImageReader.java:1577)  
at javax.imageio.ImageIO.read(ImageIO.java:1448)  
at javax.imageio.ImageIO.read(ImageIO.java:1308)
```

File Size versus Image Size

- Relying on file size leads to *Decompression Bomb Vulnerability*
 - ▶ Images are usually compressed
 - ▶ 19.000×19.000 uni-color PNG uses 44 KByte disk space but up to one GB memory
- Read dimension from image metadata

Determining Image Size

```
ImageInputStream in = ImageIO.createImageInputStream(imageFile);

final Iterator<ImageReader> readers = ImageIO.getImageReaders(in);

if (readers.hasNext())
{
    ImageReader reader = readers.next();

    try
    {
        reader.setInput(in);
        return new Dimension(reader.getWidth(0), reader.getHeight(0));
    } finally {
        reader.dispose();
    }
}
```

What's Wrong?!

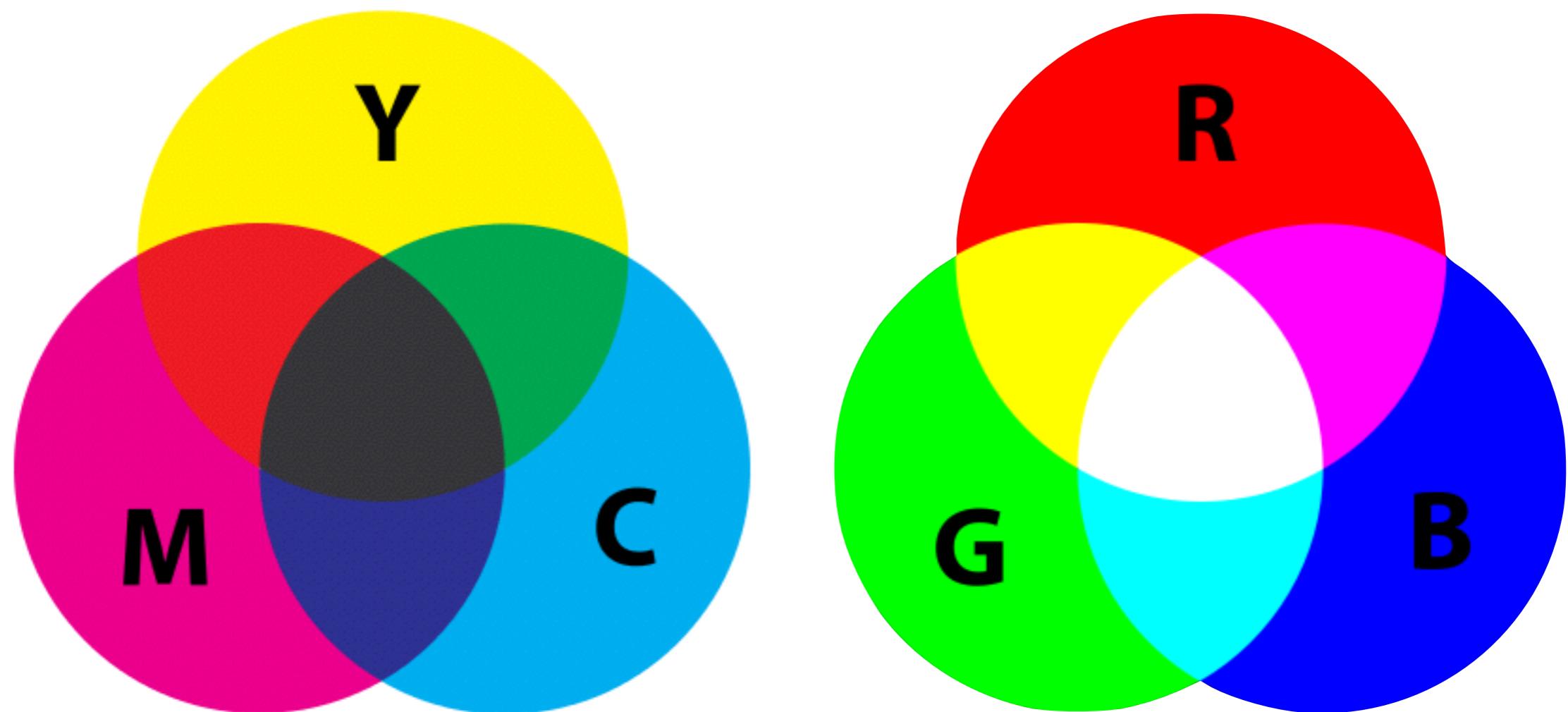


JPEG

```
javax.imageio.IIOException: Unsupported Image Type
```

```
at com.sun.imageio.plugins.jpeg.JPEGImageReader.readInternal(JPEGImageReader.java:1063)
at com.sun.imageio.plugins.jpeg.JPEGImageReader.read(JPEGImageReader.java:1034)
at javax.imageio.ImageIO.read(ImageIO.java:1448)
at javax.imageio.ImageIO.read(ImageIO.java:1308)
```

CMYK Color Model



CMYK Color Model

- CMYK is a subtractive color model
- Used in color printing and hence Photoshop users
- Java ImageIO JPEGImageReader will not read CMYK images
- Conversion to RGB is non-trivial

Old post, but for future reference:

Inspired by this question and links found here, I've written a JPEGImageReader plugin for ImageIO that supports CMYK color models (both with original color model, or implicitly converted to RGB on read). The reader also does proper color conversion, using the ICC profile embedded in the JPEG stream, in contrast to other solutions mentioned here.

It's plain Java and does not require JAI. The source code and binary distributions is freely available at github.com/haraldk/TwelveMonkeys.

Once you have it installed, it allows you to read CMYK JPEGs using `ImageIO.read(...)` like this:

```
File cmykJPEGFile = new File(/*path*/);
BufferedImage image = ImageIO.read(cmykJPEGFile);
```

i.e.: In most cases, it's not necessary to modify your code.

share improve this answer

edited Aug 7 '14 at 12:46

answered Apr 22 '13 at 14:04



haraldK

8,350 ● 3 ● 13 ● 36



12 MONKEYS

Source: <https://fanart.tv/movie/63/12-monkeys/>

TwelveMonkeys History

- As all good ideas, it started after a long night at the pub...
- Initially developed for a CMS Swing client
- Read support for OSX clipboard data and Photoshop images
 - ▶ PICTImageReader
 - ▶ PSDImageReader

TwelveMonkeys History

- Later used in various other projects, including an online bookstore
- Solving many real-world JPEG issues
 - ▶ CMYK color space
 - ▶ AdobeRGB ICC profile support
 - ▶ Broken ICC profiles handling
 - ▶ Inconsistent JPEG metadata

Why Use TwelveMonkeys?

- Uses standard ImageIO API
- Actively maintained project
- No native code, easy install
- Available on Maven Central
- Open source, with liberal license (BSD)



TwelveMonkeys Road Map

- Improved JPEG support
 - ▶ Lossless JPEG
 - ▶ Arithmetic coding
 - ▶ CMYK write support
- Improved TIFF support
 - ▶ Read/write support for CCITT T4 and T6
 - ▶ Metadata

TwelveMonkeys Road Map

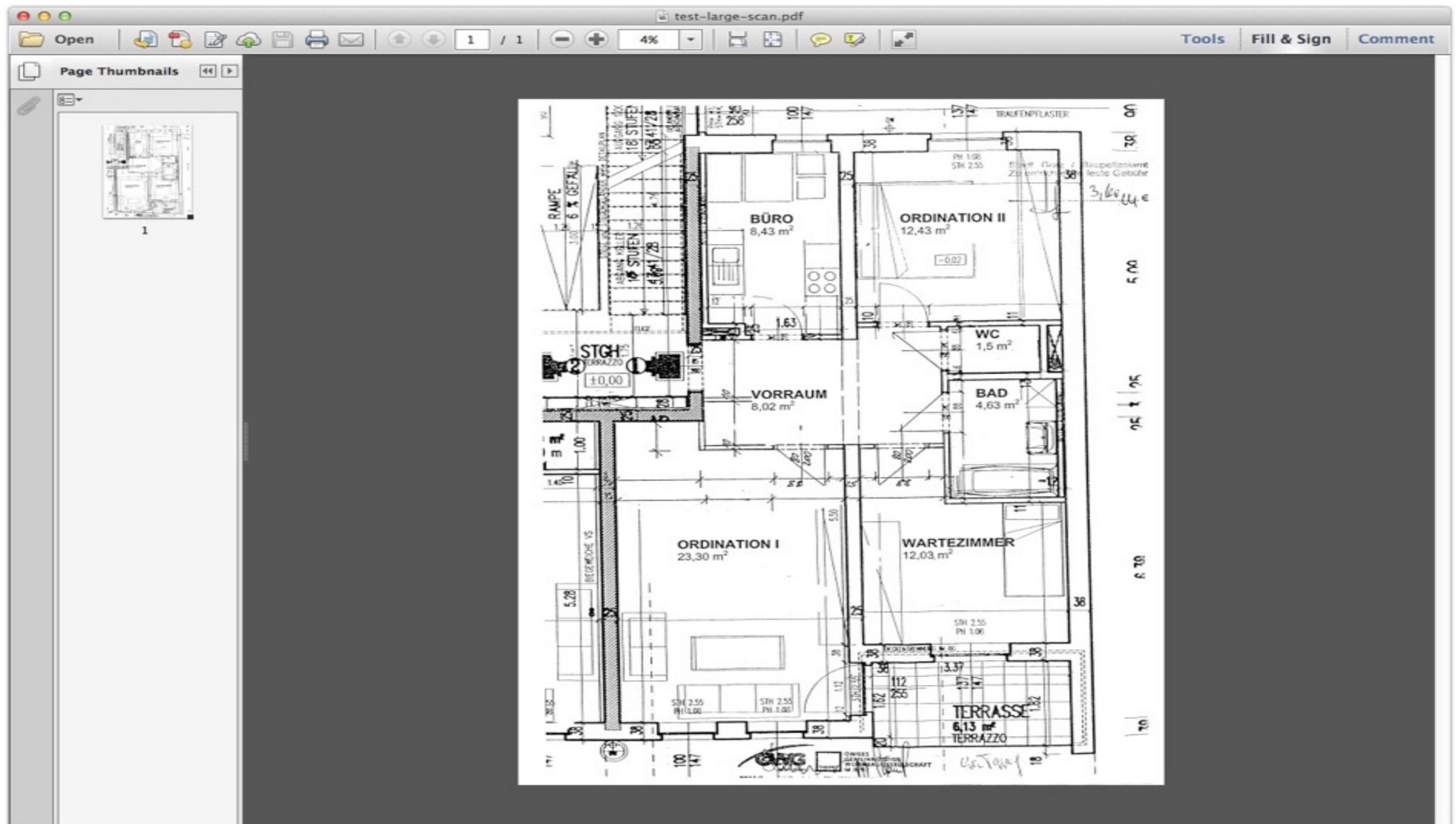
- Camera RAW plugins
 - ▶ CR2 (Canon)
 - ▶ NEF (Nikon)
 - ▶ TIFF/EP and DNG (ISO/Adobe)

Hitting Production



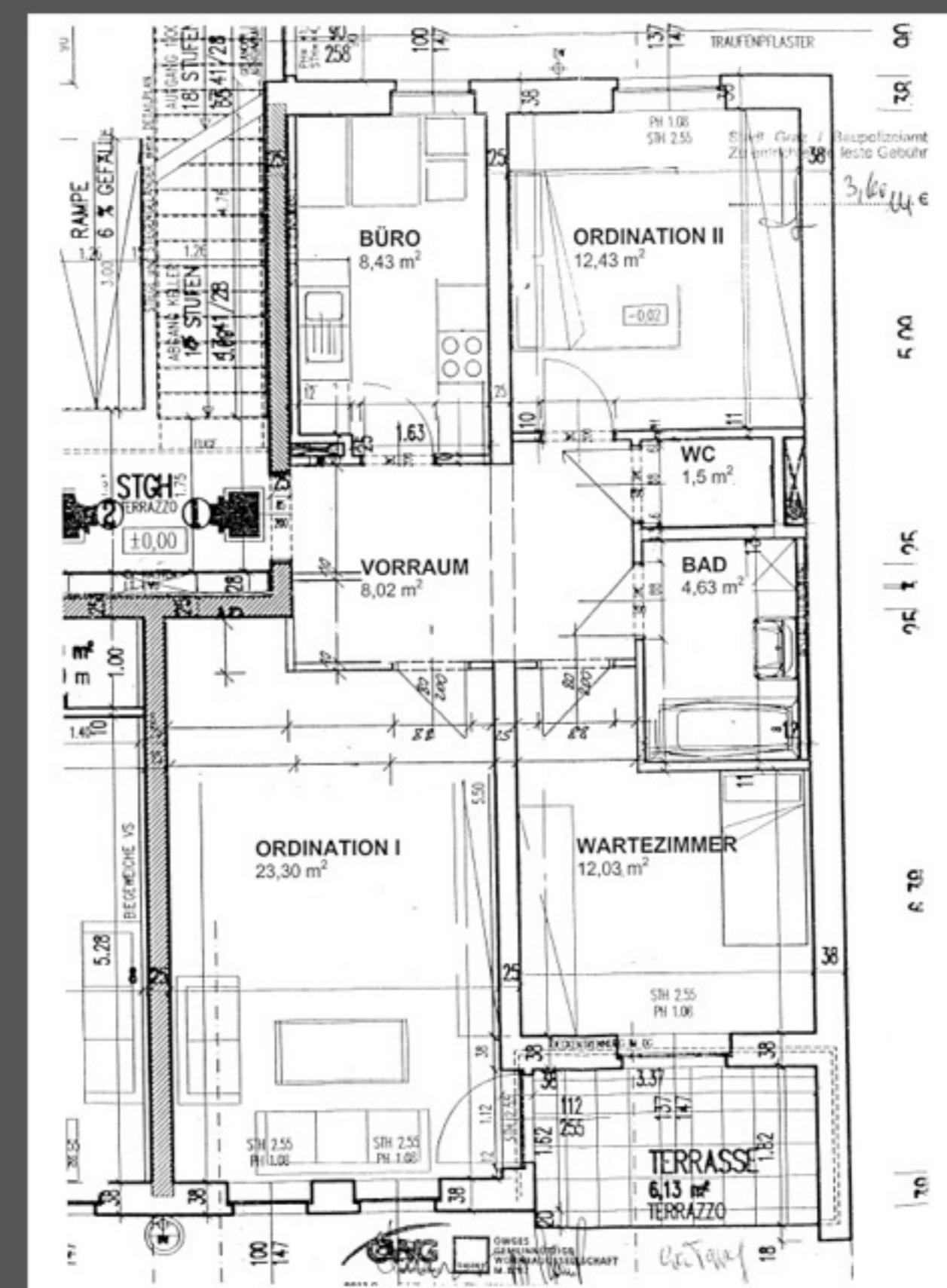
Source : www.wikipedia.org

What's Wrong?!



ImageIO Segment Violation

```
#  
# A fatal error has been detected by the Java Runtime Environment:  
#  
# SIGSEGV (0xb) at pc=0x00007fce89c42353, pid=20878, tid=140525025629952  
#  
# JRE version: Java(TM) SE Runtime Environment (7.0_45-b18) (build 1.7.0_45-b18)  
# Java VM: Java HotSpot(TM) 64-Bit Server VM (24.45-b08 mixed mode linux-amd64 compressed o  
ops)  
# Problematic frame:  
# C  IntArgbPreToIntRgbSrcOverMaskBlit+0x2a3  
#  
Stack: [0x00007fce882c1000,0x00007fce883c2000],  sp=0x00007fce883b8428,  free space=989k  
Native frames: (J=compiled Java code, j=interpreted, Vv=VM code, C=native code)  
C  [libawt.so+0x77353]  IntArgbPreToIntRgbSrcOverMaskBlit+0x2a3  
C  [libawt.so+0x45017]  Java_sun_java2d_loops_TransformHelper_Transform+0xeb7  
j  sun.java2d.loops.TransformHelper.Transform(Lsun/java2d/loops/MaskBlit;Lsun/java2d/Surfac  
eData;Lsun/java2d/SurfaceData;Ljava.awt/Composite;Lsun/java2d/pipe/Region;Ljava.awt/geom/Af  
fineTransform;IIIIIIII[III)V+0  
j  sun.java2d.pipe.DrawImage.renderImageXform(Lsun/java2d/SunGraphics2D;Ljava.awt/Image;Ljava  
/awt/geom/AffineTransform;IIIIILjava/awt/Color;)V+505  
j  sun.java2d.pipe.DrawImage.transformImage(Lsun/java2d/SunGraphics2D;Ljava.awt/Image;IILja  
va/awt/geom/AffineTransform;I)V+366  
j  sun.java2d.pipe.DrawImage.transformImage(Lsun/java2d/SunGraphics2D;Ljava.awt/Image;Ljava  
/awt/geom/AffineTransform;Ljava/awt/image/ImageObserver;)Z+17  
j  sun.java2d.pipe.ValidatePipe.transformImage(Lsun/java2d/SunGraphics2D;Ljava.awt/Image;Ljava  
/awt/geom/AffineTransform;Ljava/awt/image/ImageObserver;)Z+17  
j  sun.java2d.SunGraphics2D.drawImage(Ljava/awt/Image;Ljava/awt/geom/AffineTransform;Ljava/  
awt/image/ImageObserver;)Z+111  
j  org.apache.pdfbox.pdfviewer.PageDrawer.drawImage(Ljava/awt/Image;Ljava/awt/geom/AffineTr  
ansform;)V+35
```



1 / 1



3,79%



SU

-

RAMPE

3,00 6% GEFAELLE

SIEGE und STIEGENGELÄNDER siehe DETAILPLAN

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

3,79%



1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

1.26

BÜRO

8,43 m²

ORDINATION II

12,43 m²

-0,02

STG

1.75

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

1.0

General Info	
File name:	test-large-scan.pdf
Document type:	Portable Document Format (PDF)
File size:	331 KB (330.946 bytes)
PDF version:	1.2
Page count:	1
Page size:	359,84 × 493,61 cm
Title:	-
Author:	-
Subject:	-
PDF Producer:	-
Content creator:	-
Creation date:	-
Modification date:	-

ImageIO Segment Violation

- PDF with a size of 360×495 cm
- Passed 300 DPI to PDBox assuming A4 size
- Native ImageIO code bombs and kills JVM
- Load balancer propagates the problem
- Calculate DPI value used for PDFBox

ImageIO Segment Violation

```
int imageType = BufferedImage.TYPE_INT_RGB;
PDPage page = pages.get(i);
PDRectangle cropBox = page.findCropBox();
float width = cropBox.getWidth();
float height = cropBox.getHeight();
int currResolution = calculateResolution(resolution, width, height);
BufferedImage image = page.convertToImage(imageType, currResolution);

/**
 * Calculate the resolution being used assuming that the DPI are intended
 * for an A4 page.
 */
protected int calculateResolution(int dpi, float cropBoxWidth, float cropBoxHeight)
{
    int result;

    float maxPoints = Math.max(cropBoxWidth, cropBoxHeight);
    float pointsForRequestedResolution = 29.7f * dpi / 2.54f;
    result = Math.round((pointsForRequestedResolution * DPI_72 / maxPoints));
    return result;
}
```

What's Wrong?!

```
java.lang.NoClassDefFoundError: Could not initialize class javax.imageio.ImageIO
```

```
java.lang.NullPointerException
at com.twelvemonkeys.imageio.color.ColorSpaces.getColorSpace(ColorSpaces.java:265)
at com.twelvemonkeys.imageio.plugins.jpeg.JPEGImageReader.getImageTypes(JPEGImageReader.java:232)
at com.twelvemonkeys.imageio.plugins.jpeg.JPEGImageReader.readImageAsRasterAndReplaceColorProfile
at com.twelvemonkeys.imageio.plugins.jpeg.JPEGImageReader.read(JPEGImageReader.java:337)
at javax.imageio.ImageIO.read(ImageIO.java:1422)
...
```

ImageIO & Web Apps

- ImageIO Plugin Registry is VM global and doesn't play nicely with web apps
 - ▶ Need to invoke `ImageIO.scanForPlugins()` to discover plugins
 - ▶ Class-loader issue & resource leaks
- TwelveMonkeys `IIOProviderContextListener`
 - ▶ Dynamic loading/unloading of plugins

ImageIO & Web Apps

```
<web-app>
  <listener>
    <display-name>
      ImageIO service provider loader/unloader
    </display-name>
    <listener-class>
      com.twelvemonkeys.servlet.image.IIOProviderContextListener
    </listener-class>
  </listener>
</web-app>
```

What's Wrong?!

```
Caused by: java.awt.color.CMMException: LCMS error 13: Couldn't link the profiles
at sun.java2d.cmm.lcms.LCMS.createNativeTransform(Native Method)
at sun.java2d.cmm.lcms.LCMS.createTransform(LCMS.java:156)
at sun.java2d.cmm.lcms.LCMSTransform.doTransform(LCMSTransform.java:155)
at sun.java2d.cmm.lcms.LCMSTransform.colorConvert(LCMSTransform.java:467)
at java.awt.image.ColorConvertOp.filter(ColorConvertOp.java:571)
at com.sun.imageio.plugins.jpeg.JPEGImageReader.acceptPixels(JPEGImageReader.java:1268)
at com.sun.imageio.plugins.jpeg.JPEGImageReader.readImage(Native Method)
at com.sun.imageio.plugins.jpeg.JPEGImageReader.readInternal(JPEGImageReader.java:1236)
at com.sun.imageio.plugins.jpeg.JPEGImageReader.read(JPEGImageReader.java:1039)
at javax.imageio.ImageIO.read(ImageIO.java:1448)
at javax.imageio.ImageIO.read(ImageIO.java:1308)
```

ImageIO & JDK 8

- New Color Management System
- Kodak CMS replaced by LittleCMS (good!)
- Affect ICC profile handling (bad!)
- Affect ColorConvertOp performance
- You can get the old CMS back (for now)

`-Dsun.java2d.cmm=sun.java2d.cmm.kcms.KcmsServiceProvider`

What's Wrong?!

The screenshot shows a blog interface with a sidebar on the right and five posts listed vertically on the left.

Posts:

- 1: Acer Iconia A700 Tablet im Test – Alternative zum Nexus 10?
Date: 25 JAN/13
Image: Acer Iconia A700 tablet screen
- 2: Blank page after upgrading Infinite Scroll plugin from 2.0b2 to latest version
Date: 9 DEC/12
Image: WordPress logo
- 3: Western Digital 3TB Festplatte WD30EZRX – Green vs. Black Label
Date: 23 NOV/12
Image: Western Digital 3TB Festplatte WD30EZRX
- 4: (partially obscured)
- 5: (partially obscured)

Sidebar:

- Pages: About, Datenschutz, Impressum
- Categories: Administration, Android, Computer, Data, Life, Mac, Review, Software, Wordpress
- Recent Comments:
 - Ereakie: habe win7 32-bit und 3TB bei dem WD-HDD laufen darauf immer, MIT HDD anderer hersteller gab es Probleme im Betrieb. Western Digital 3TB Festplatte WD30EZRX, Green vs. Black Label - 12 hours ago
 - Danish: Ahh, I see. Thanks again for taking the time to respond. I thought you were still using the plugin mentioned in this post ;) Infinite Scroll with LightHouse and another social button - 2 months ago
 - Von Gana: I'm simply using the plain vanilla dropdown plugin. Almost

**Thousands of
Incompatible
File Formats**

TIFF

- Tagged Image File Format
- Supports dozen of image, storage and compression formats
- It is practically impossible to correctly process all real-life TIFF image files
- Baseline TIFF as minimal functionality
- But “TIFF 6.0, Part 2:TIFF Extensions” are commonly used in real-life

Baseline TIFF

- Multi-page support
- Image Types - bilevel, grayscale, palette-color and RGB full-color images
- Compression schemes - uncompressed, CCITT Group 3 encoding, PackBits
- Little and Big Endian byte order
- Handling of optional fields

TIFF 6.0 Extensions

- Additional compression schemes - CCITT T.4 & CCITT T.6 bi-level encoding, LZW, JPEG-based compression
- Additional image types - CMYK, YCbCr, HalftoneHints, Tiled Images, CIE L*a*b*
- TIFF 6.0 Extensions partially supported by JAI, Apache Commons Imaging & TwelveMonkeys

What's Wrong?!

VOLVO

TRANSPORTMELDING

Nr. 00000000
COPY

Geleverde Route:
Bestemming M: 0 - 00000000
Na: 00000000
Van: 00000000

Ongelv.: 00000000
Aanvlg.: 0000

Transporttijd: 00000000 - 00000000 | TM: datum + uur: 0000-00-00 00:00:00

Leidendeerd: 00000000 | ETA: 0000-00-00 00:00:00

Type: TRA | Transportwijze: ROAD

Gemerken: 0

Vege/ Lading	Lijnnummer	Uit	Bestemming	Bestemming suffix	Punt	Datum	Punt	Tpl. Col.	Bte/ gewicht	Volume kub.mtr
1	VET	VET	07A					15	1389,99	11,674
1	VET	VET	07B					6	574,00	5,229
2	KN-VET	KN-VET	07B					56	7000,00	54,320

Betrekking DK/NOK: Later DK/NOK | TOTAAL: 75 8963,99 71,223

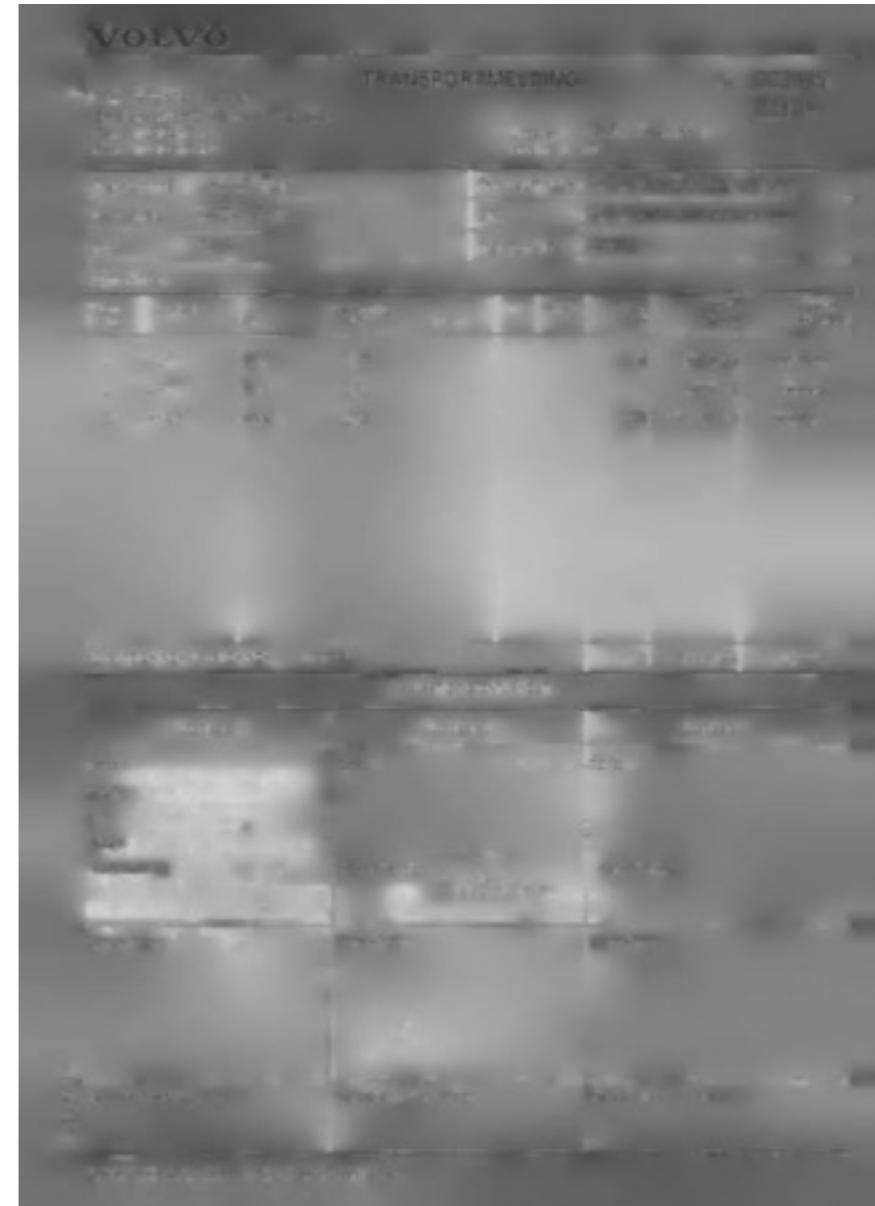
BEVESTIGD OBTWAGST

Lijnnummer 1	Lijnnummer 2	Lijnnummer 3
Tpl.col. Datum Unit Naam Handtekening	Tpl.col. Datum Unit Naam Handtekening	Tpl.col. Datum Unit Naam Handtekening
Gemerken: <i>Hartog, KDA</i>	Gemerken: <i>Svenn Groenendijk, KDA</i>	Gemerken:
Handtekening vervoerder	handtekening vervoerder	handtekening vervoerder

F VOLVO Inc. 2001

FASCO-FOR HERZ 2, 2001-06-DIVERG, VERW. COV. 1 2880001

PDF



JPEG

PDFBox & JBIG2

- PDF contains a JBIG2 image
- JBIG2 was added to PDF 1.4 in 2004
- PDFBox relies on ImageIO to convert the source image into a JPEG
- No suitable ImageIO plugin was found and PDFBox returned a JPEG nevertheless

PDFBox & JBIG2

```
ERROR JBIG2Filter - Can't find  
an ImageIO plugin to decode the  
JBIG2 encoded datastream.  
ERROR PDPPixelMap - Something  
went wrong ... the pixelmap  
doesn't contain any data.  
WARN PDXObjectImage - masking  
getRGBImage returned NULL
```

PDFBox & JBIG2

- Use Levigo's JBIG2 ImageIO plugin
 - ▶ <https://github.com/levigo/jbig2-imageio>
 - ▶ Uses GPLv3 (unfortunately)
 - ▶ Contact the vendor for other licensing options

A photograph of three students sleeping in a lecture hall. In the foreground, a woman with dark hair, wearing a white shirt and a gold watch, rests her head on her arms which are propped up on a wooden desk. In the middle ground, a man in a dark blue shirt is also sleeping with his head down. In the background, another student in a red jacket is visible, also asleep. The room has rows of wooden desks and chairs.

Things to take home

Things To Take Home

- You won't handle all of your real-life images & PDFs without issues
 - ▶ Get as many test documents as possible
 - ▶ Be prepared to hit the learning curve
- Stay clear of TIFF if possible
- Never use the term “TIFF” - use “Baseline TIFF” or “Baseline TIFF with some extensions” instead

Things To Take Home

- `TwelveMonkeys` for image processing
 - ▶ Scaling, color conversion, cropping, water-marking
 - ▶ Real-life JPEG support
- JAI and/or Apache Imaging for additional TIFF support
- Check out dedicated image scaling libraries if performance is critical

[github.com/sgoeschl/
java-image-processing-
survival-guide](https://github.com/sgoeschl/java-image-processing-survival-guide)

[github.com/haraldk/
TwelveMonkeys](https://github.com/haraldk/TwelveMonkeys)

Questions & Answers

