



# Ghostscript and MuPDF Status OpenPrinting summit April 2015

Michael Vrhel, Ph.D.  
Artifex Software Inc.  
San Rafael CA





## Outline

Ghostscript overview

Whats new with Ghostscript

Color overview in Ghostscript

MuPDF

GSView



## The Basics

Ghostscript is a document conversion and rendering engine.

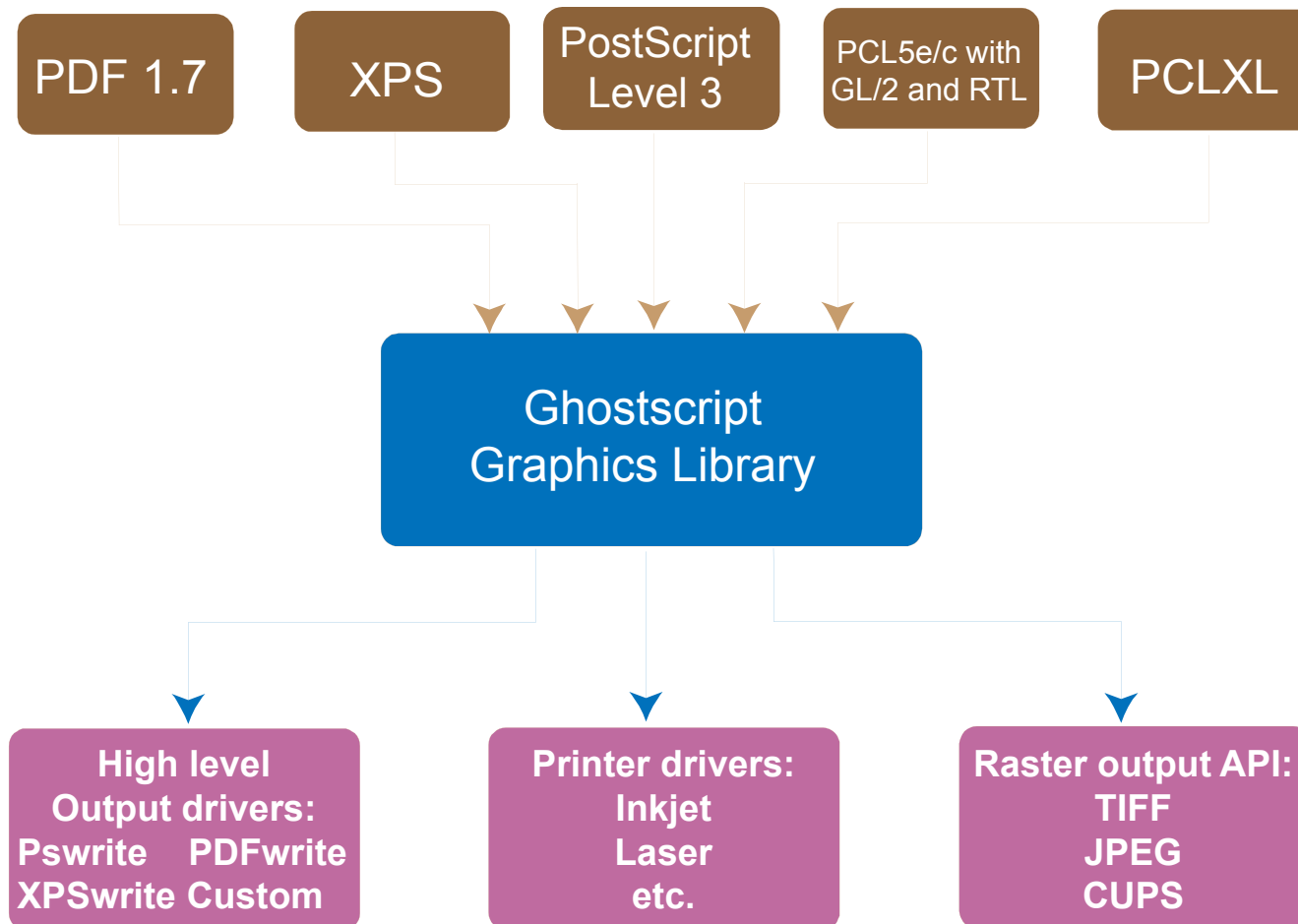
Written in C ANSI 1989 standard (ANS X3.159-1989)

Essential component of the Linux printing pipeline.

Dual AGPL/Proprietary licensed. Artifex owns the copyright.

Source and documentation available at [www.ghostscript.com](http://www.ghostscript.com)

# Graphical Overview





## Devices

Understanding devices is a major key to understanding Ghostscript.

Devices can have high-level functionality. e.g. pdfwrite can handle text, images, patterns, shading, fills, strokes and transparency directly.

Devices may be set up to handle only certain high-level operations.

Graphics library has “default” operations. e.g. text turns into bitmaps, images decomposed into rectangles.

In embedded environments, calls into hardware can be made.

Raster devices require the graphics library to do all the rendering.



## Relevant Changes to GS since last meeting....

Ghostscript now supports the PDF security handler revision 6 (9.15)

The pdfwrite and ps2write (and related) devices can now be forced to "flatten" glyphs into "basic" marking operations `-dNoOutputFonts` (9.15)

PostScript programs can now determine if a page contains color markings by reading the `pageneutralcolor` state from the device. Only accurate when in `clist` mode. `-dMaxBitmap=0 -dGrayDetection=true` (9.15)



## Relevant Changes to GS since last meeting....

pdfwrite device supports Link annotations with GoTo and GoToR actions (9.15)

pdfwrite device supports BMC/BDC/EMC pdfmarks (9.15)

pdfwrite device fully supports the ICC color management. Use of -dUseCIEColor is **strongly** discouraged. (9.16)

"LockColorants" command line option for tiffsep and psdcmymk devices.

-dLockColorants will restrict those devices to only the colorants

list on the command line

```
-c "<< /SeparationColorNames [ /Cyan /Magenta /Yellow /Black /Violet /Orange]  
/SeparationOrder [ /Cyan /Magenta /Yellow /Black /Violet /Orange]>>  
setpagedevice") (9.16)
```



## Relevant Changes to GS since last meeting....

New URW+ fonts in which the families NimbusMono, NimbusRoman and NimbusSans have been augmented with production quality Greek and Cyrillic glyphs. (9.16)

Improvements in xpswrite device in terms of support for text and images. (9.16)

Number of bug fixes in named color replacement for separation and DeviceN colors. (9.16)





# Ghostscript Color Architecture

- Easy to interface different CMM with Ghostscript.
- ALL color spaces defined in terms of ICC profiles.
- Linked transformations and internally generated profiles cached.
- Easily accessed manager for ICC profiles.
- Easy to specify default profiles for DeviceGray, DeviceRGB and DeviceCMYK.
- Devices communicate their ICC profiles and have their ICC profile set.
- Operates efficiently in a multithreaded environment.
- Handles named colors with ICC named color profile or proprietary format.
- ICC Color management of Device-N colors or customizable spot handling.
- Includes object type (e.g. image, graphic, text) and rendering intent into the computation of the linked transform. Maintained with transparency.



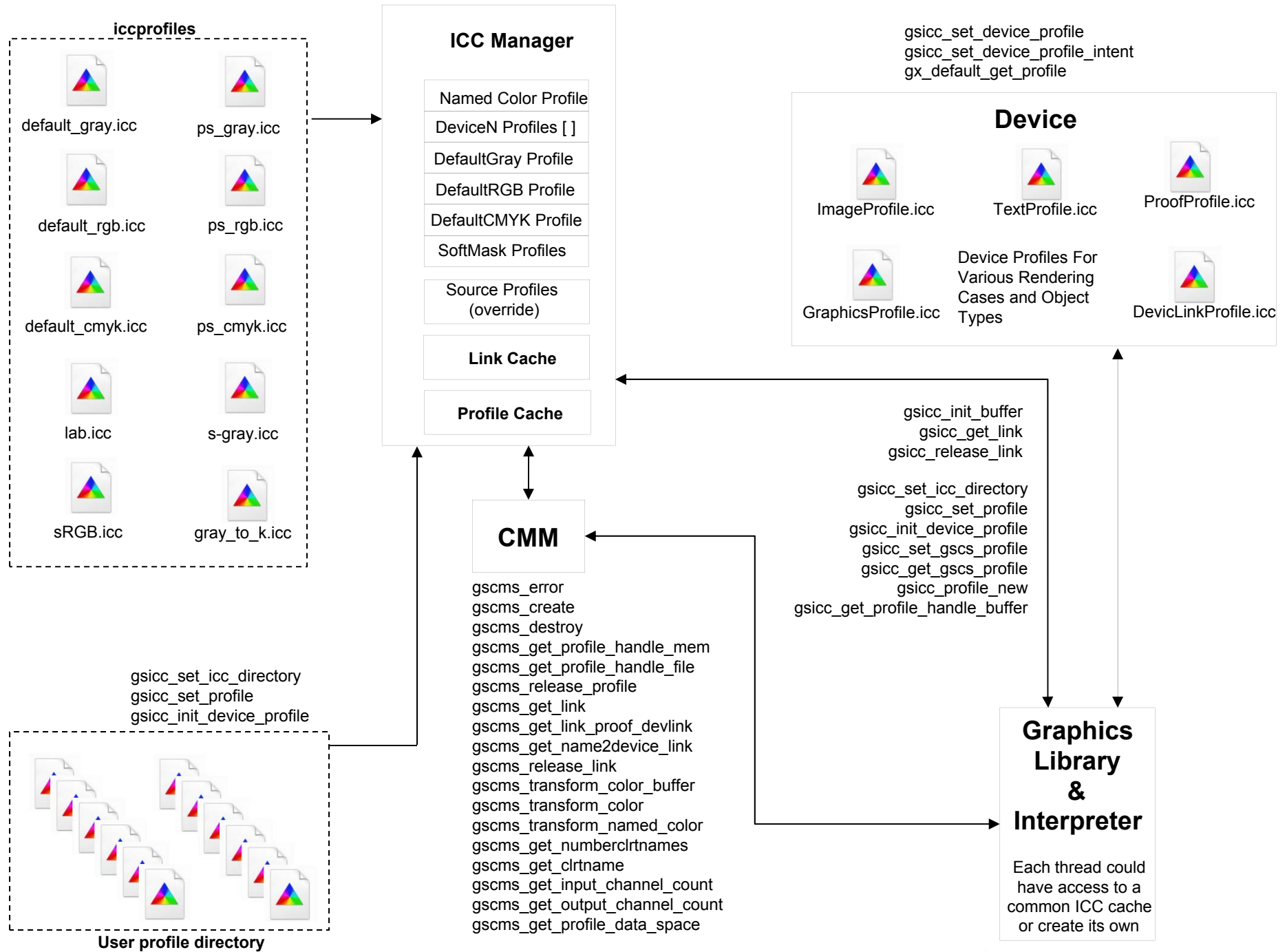
# Ghostscript Color Architecture

- Ability to override document embedded ICC profiles with Ghostscript's default ICC profiles.
- Easy to specify unique **source** ICC profiles to use with CMYK and RGB graphic, image and text objects.
- Easy to specify unique **destination** ICC profiles to use with graphic, image and text objects.
- Easy to specify different rendering intents (perceptual, colorimetric, saturation, absolute colorimetric) and black point comp. for graphic, image and text objects.
- Control to force gray source colors to black ink only for devices that support black ink (e.g. CMYK).

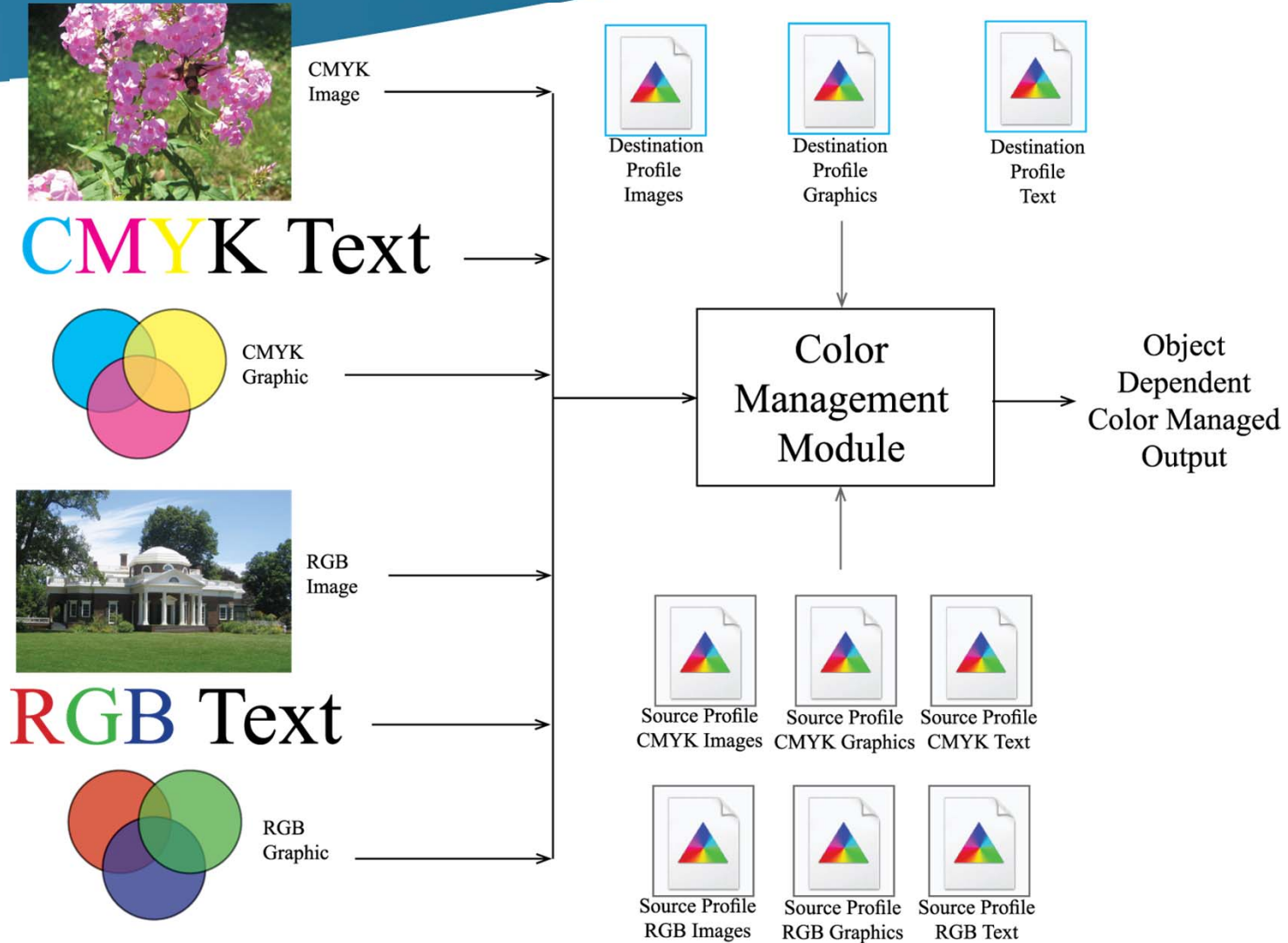


# Ghostscript Color Architecture

- Make use of PDF output intent ICC profile.
- Use an NCLR ICC output profile when rendering to a separation device.
- Make use of device link ICC profiles for direct mapping of source colors to the device color space.
- Ability to make use of device link ICC profiles for retargeting from SWOP/Fogra standard color space to a specific device color space.



# Object Dependent Color Management





## Special Color Handling

The color transformation operations have a simple API.

A link (which transforms colors from one space to another) has a structure called `gscms_procs_t` which contains three functions.

```
void (*gscms_trans_color_proc_t) (gx_device*, gsicc_link_t*,  
                                   void *inputcolor, void *outputcolor, int num_bytes);
```

```
void (*gscms_trans_buffer_proc_t) (gx_device*, gsicc_link_t*,  
                                   gsicc_bufferdesc_t *input_buff_desc,  
                                   gsicc_bufferdesc_t *output_buff_desc,  
                                   void *inputbuffer, void *outputbuffer);
```

```
void (*gscms_link_free_proc_t) (gsicc_link_t *icclink);
```



## Special Color Handling Continued

It is simple to replace these procedures with other ones to do special color handling/customization with minimal code writing.

Ghostscript includes three examples:

**gsicc\_nocm.c**

This contains code which uses 255-X style color management.

**gsicc\_monitorcm.c**

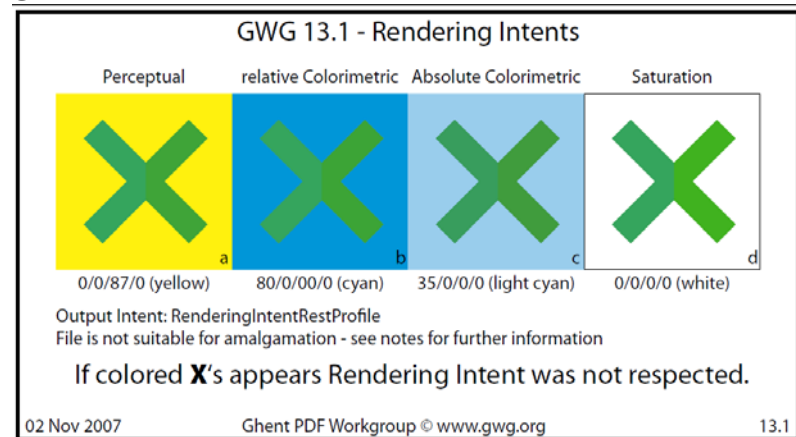
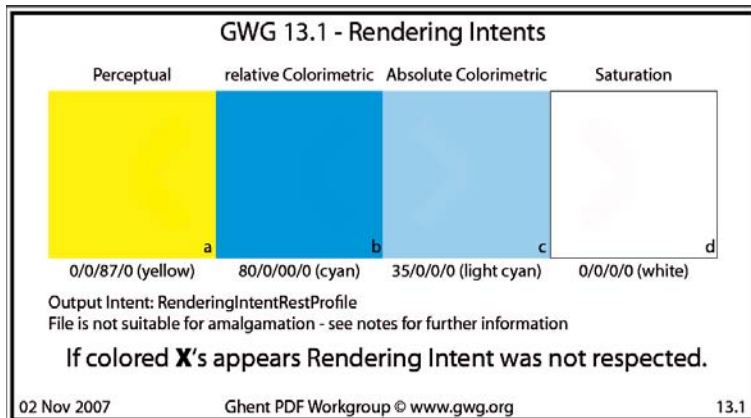
This contains code to monitor the colors encountered during management. Used to monitor for non-gray content.

**gsicc\_replacecm.c**

This contains code to replace the incoming colors in some “special” way.

# PDF Output Rendering Intent

GS supports PDF Output Rendering Intent usage Passes all Ghent tests and these are included in the testing suite.



OutputIntents array (Optional; PDF 1.4) An array of output intent dictionaries describing the color characteristics of output devices on which the document might be rendered (see "Output Intents" on page 970).





## PDF Output Rendering Intent

`-dUsePDFX3Profile = #`

Where # defines which output intent to use in the order that they occur in the document. If no number specified, first one encountered is used.

If no profile is present in the intent dictionary, a warning is displayed and the rendering intent is ignored.

If the output intent ICC profile does not match the process color model of the output device, then the output intent ICC profile is used as a proofing profile.



## More Information

[http://www.ghostscript.com/doc/current/GS9\\_Color\\_Management.pdf](http://www.ghostscript.com/doc/current/GS9_Color_Management.pdf)



MuPDF





# What is MuPDF?



- **A core set of libraries**
  - + Entirely written in C
  - + Very portable
    - We've done: Windows/Linux/MacOS/iOS/Android
    - Third parties: BB10/QNX/others
- **Various example tools that use these libraries:**
  - + Simple viewers for Linux/Android/MacOS/iOS/Windows/WinRT.
  - + Command line tools for rendering PDF pages.
  - + Command line tools for manipulating PDF files
    - page extraction
    - decompression
    - repair
    - resource extraction
- **Licensing the same as Ghostscript.** Dual AGPL/Proprietary licensed. Artifex owns the copyright.



## Ghostscript or MuPDF?



<http://twiki.ghostscript.com/do/view/Ghostscript/GhostscriptOrMuPDF>

**For most printing applications - use Ghostscript.**

- \* Postscript
- \* PCL
- \* Spot colors
- \* Extreme level of color management control
- \* Massive range of output devices



# Ghostscript or MuPDF?



**For screen use or embedded devices - use MuPDF.**

## **Fast**

- + PDF Parser in C.
- + AA Rendering designed in from the ground up.

## **Small**

- + Much smaller ROM footprint.

## **Simple**

- + No complex garbage collector to maintain
- + Small set of dependent libraries
- + Simpler to port

## **Interactive features**

- + More suitable for building viewers
- + Searching
- + Zooming
- + Form filling
- + Transitions



# Features of MuPDF



## **Not just "PDF" - Other formats too:**

- + XPS/OXPS
- + CBZ/JPEG/PNG/TIFF/EPUB (version 2 DRM-free)
- + Extensible system

## **"Device" interface**

- + Separates interpretation from rendering
- + Allows display lists
  - interpret page once, render many times at different zooms
- + Allows format conversions (more later)



# Features of MuPDF



## **Clever memory management**

- + Caching of objects (both raw and decoded)
- + Memory scavenging (throw objects away just in time)

## **Optional Multi-core/threading support**

- + Not tied to any one threading implementation
  - All we need is locks
- + Interpretation happens on 1 thread
- + Rendering can happen on many.
  - Thumbnails rendered in the background.
  - Banded rendering of pages.
  - Resources decoded on one thread can be used by others.





# Features of MuPDF



## Form filling

- + Javascript for validation
  - Not tied to any one javascript implementation
  - Thin veneer to Googles 'v8' engine supplied
  - muJS is our own small javascript library (<http://www.mujs.com/>)
- + Ability to save files back with data in them.



# Features of MuPDF



## **Reflow View**

- + Pages extracted to HTML (text and images). New layout engine in version 1.7
- + Rudimentary layout detection (tables, indents, etc.)

## **Digital Signatures**

- + Verify signatures
- + Sign documents
- + Re-sign documents after form filling



# GSView



## GSView

*a user friendly viewer for Postscript, PDF, XPS, EPUB<sup>1</sup>, CBZ, JPEG, and PNG*

GSView leverages the viewing capabilities of MuPDF, along with the conversion capabilities of Ghostscript to provide fast and high quality on-screen experience, and high quality export and conversion features.

GSView is currently in its beta testing phase.

[www.gsview.com](http://www.gsview.com)

# Bug Tracking

<http://bugs.ghostscript.com/>

**Bugzilla – Main Page** version 4.2.5

Home | New | Browse | Search |  Search [?] | Reports | Preferences  
| Administration | Log out michael.vrhel@artifex.com

## Welcome to Bugzilla



[File a Bug](#)



[Search](#)



[User Preferences](#)

[Quick Search help](#)

[Bugzilla User's Guide](#) | [Release Notes](#)

---

Home | New | Browse | Search |  Search [?] | Reports | Preferences  
| Administration | Log out michael.vrhel@artifex.com

My Bugs | All Mine | My Closed | My Customer Bugs | regresssions\_mjv  
customer bugs



## More Information



Repositories located at  
[git://git.ghostscript.com](https://git.ghostscript.com)

MuPDF and Ghostscript discussions on IRC freenode #ghostscript channel

Additional information at [www.mupdf.com](http://www.mupdf.com) [www.ghostscript.com](http://www.ghostscript.com)



Thank you for your attention!



michael.vrhel (at) artifex.com