

# Print Mode Specs

(this isn't a finished specification, it was just came to mind quickly, so it's just a quick and dirty braindump at this point)

Darktable current has no way to print images...

My current workflow is:

1. Export from Darktable to 16 TIFF (AdobeRGB)
2. transform to printer profile using tificc to move the image to the printers native colorspace
3. open the transformed image in Eye of GNOME, and sent to my printer  
(Eye of GNOME allow simple adjustments like leaving an unprinted border)

Current Linux has some color management print infrastructure via colord, but currently there is no integration with the (GTK) print dialog, so can setup a single *printer* associated profile which is transparently used (there is no explicitly sign of this in the print dialog). So this is major nuisance if you regular print use different paper types (not so uncommon, since there is a huge selection of fine-art papers available on the market)...

Another disadvantage of solving this print color management via cups is that print profile rendering intents are inherently tied to the source working space. For relative colorimetric this isn't an issue since practically there is no gamut mapping happening (making this the only *safe* rendering intent to use without user interaction/verification). Perceptual (and to less importance Saturation) rendering intents are gamut mapped and need to mapped to a specific source gamut (typically AdobeRGB), this source gamut however is as far as I'm aware never stored in the printer icc, so there is no way to cups to "see" which the matching source gamut would be. Thus this is always something that the end-user needs to manually control, if he wants to use anything but relative colorimetric.

All issues considered it would make sense to implement a print mode in Darktable on the assumption someone is willing to invest the required effort.

The following items are in my opinion critical:

1. Print preview where you can see a small "paper" rendering with your image with some simple controls so the printable area can be shrunk to leave a unprinted border. When a print profile below has been specified it also makes a LOT of sense to softproof this preview by default (possibly including paper white simulation).
2. Just like export in 1.2, allow the application of a "export/print/-time-only" style (think for fancy borders/watermarks)
3. output sharpening (this should probably do clever stuff with paper size/viewing distance to come up with unsharp mask parameters)
4. output print profile (sRGB by default for non profiled printers)
5. print profile rendering intent

From a color management point of view we'd be effectively:

1. Doing an normal export at 16bit bitmap in AdobeRGB (if a print profile has been specified)
2. Transform that bitmap to the print colorspace
3. Send to the printer keeping in mind the border  
In the future it's likely there will be a mechanism we could use to tag our files, so we can reliably suppress colord/cups' color management and override it with our own (for increase flexibility, keeping in mind the concerns expressed earlier)

The reason I want to use AdobeRGB as a connection space for the print profile has a number of reasons:

1. AdobeRGB has a gamut that's large enough that most printers gamut's will be covered
2. AdobeRGB has a nice smooth 2.2 gamma, without the weird notch sRGB has
3. Most paper manufacturer seem to deliver profiles that are gamut mapped against AdobeRGB, subtly indicating that this may very well be common practise in the industry.

Some may wonder why we can't transform directly to the printer own RGB colorspace, and we can't for several reasons:

1. We have modules operating on output rgb, which may not have the expection result in a printer's rgb colorspace
2. The rendering intent/gamut mapping reasons as expressed above.

It has been suggested that adding PDF export would be nice too, but without a fully compliant PDF/X implementation I'm not sure I see the value in that, and thus might better be left to specialized tools like Scribus.

## Related artwork:

Just dugged out from somewhere deep: some UI mockups by Roberto concerning the print view:

[darktable%201%20mode%20print%20module%20advanced.jpg](#)

[darktable%201%20mode%20print%20module%20basic.jpg](#)

[print-module.jpg](#)

### Files

---

[print-module.jpg](#)

79.3 KB

06/12/2014

Pascal Obry