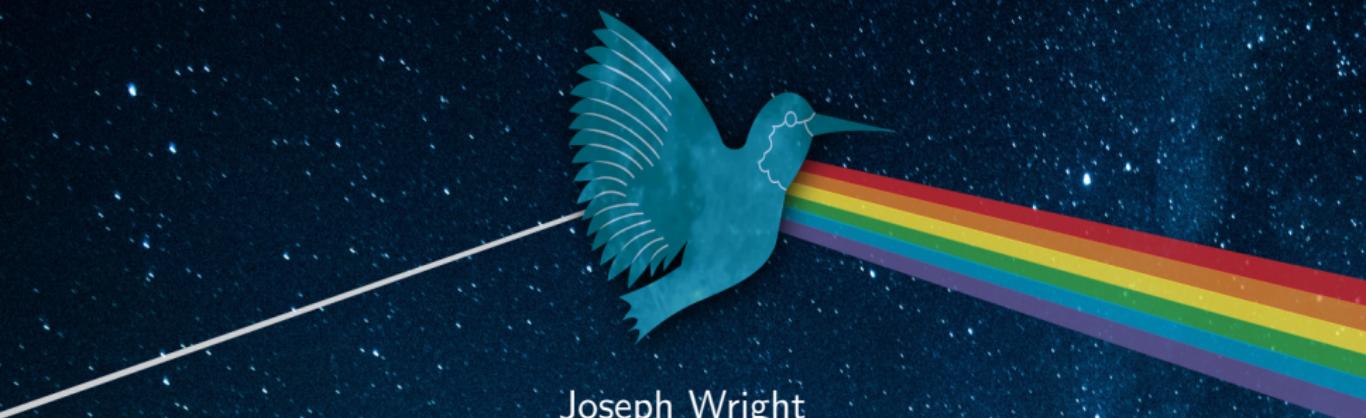


Any colo(u)r you like



Joseph Wright

LATEX Project

TUG 2021

## Acknowledgements

- David Carlisle (`color`)
- Uwe Kern (`xcolor`)
- Javier Bezos (`colorspace`)
- Heiko Oberdiek (`pdfrender`, `luacolor`)

# Background

- $\text{\TeX}$  requires specials for color

# Background

- $\text{\TeX}$  requires specials for color
- $\text{\LaTeX} 2\epsilon$  color package does that well

# Background

- $\text{\TeX}$  requires specials for color
- $\text{\LaTeX} 2\epsilon$  color package does that well
- But there is more to color

# Background

- $\text{\TeX}$  requires specials for color
- $\text{\LaTeX} 2\epsilon$  color package does that well
- But there is more to color
- Aim to bring ideas together

# Background

- $\text{\TeX}$  requires specials for color
- $\text{\LaTeX} 2\epsilon$  color package does that well
- But there is more to color
- Aim to bring ideas together
- Also use new(er) engine features

## Abstracting engine color

From an explicit model

```
\color_select:nn { rgb }  
{ 0.123 , 0.456 , 0.678 }
```

## Abstracting engine color

From an explicit model

```
\color_select:nn { rgb }
{ 0.123 , 0.456 , 0.678 }
```

or from a named color

```
\color_select:n { blue }
```

# Mixtures

Most people would prefer

```
\color_select:n { red!20!blue }
```

over

```
\color_select:nn { rgb }
```

```
 { 0.2 , 0.0 , 0.8 }
```

# Color models

There are various color models

- Grayscale (gray)
- Red–green–blue (rgb)
- Cyan–magenta–yellow–black (cmyk)

# Color models

There are various color models

- Grayscale (gray)
- Red–green–blue (rgb)
- Cyan–magenta–yellow–black (cmyk)
- Hue-saturation-brightness (hsb)
- HTML-format (HTML)
- Integer red–green–blue (RGB)

## Mixing models

When we mix, the first *native* model wins

```
\color_set:nn { test } { red!25!cyan }
\color_show:n { test }
```

# Mixing models

When we mix, the first *native* model wins

```
\color_set:nn { test } { red!25!cyan }
\color_show:n { test }
```

gives

The color foo has the properties:

```
> model => rgb
> rgb => 0.25 0.75 0.75.
```

## Converting models

As we've seen the package can convert between models.

## Converting models

As we've seen the package can convert between models.

We can provide explicit data for multiple models

```
\color_set:nnn { test } { cmyk / rgb }
{ 0.1 , 0.2 , 0.3 , 0.4 / 0.1, 0.2 , 0.3 }
```

# Converting models

As we've seen the package can convert between models.

We can provide explicit data for multiple models

```
\color_set:nnn { test } { cmyk / rgb }
{ 0.1 , 0.2 , 0.3 , 0.4 / 0.1, 0.2 , 0.3 }
```

And we can export the information

```
\color_export:nnN { red!33!green!10 }
{ HTML } \l_tmpa_tl
```

# Converting models

As we've seen the package can convert between models.

We can provide explicit data for multiple models

```
\color_set:nnn { test } { cmyk / rgb }
{ 0.1 , 0.2 , 0.3 , 0.4 / 0.1, 0.2 , 0.3 }
```

And we can export the information

```
\color_export:nnN { red!33!green!10 }
{ HTML } \l_tmpa_tl
> \l_tmpa_tl=macro:
->EEF7E6.
```

# Color spaces and spots

In print, we might need defined inks: *spot* colors

```
\color_model_new:nnn
{
    BarTone
}
{
    Separation
{
    name = BarTone~555~GN ,
    alternative-model = rgb ,
    alternative-values = { 0.123 , 0.456, 0.789 }
}
```

# Color spaces and spots

We might also want device-independent CIELAB colors

```
\color_model_new:nnn
{ BarToneCIELAB }
{ Separation }
{
    name = BarTone~555~GN ,
    alternative-model = CIELAB ,
    alternative-values = {50, -30, -40},
    illuminant = d65
}
```

# Color spaces and spots

Or DeviceN spaces

```
\color_model_new:nnn { AllIn } { DeviceN }
{
    names =
        { BarTone , cyan , magenta , yellow , black }
}
```

## Fills, strokes and stacks

- pdfT<sub>E</sub>X and LuaT<sub>E</sub>X have multiple color stacks
- So does up-to-date dvipdfmx
- These are really graphic state stacks!
- We can use them to (better) manage fill and stroke color

## (Toward) color profiles

```
\color_iccprofile_apply:nn { rgb }
{ sRGB_v4_ICC_preference.icc }
```

or

```
\color_model_new:nnn { ICC } { ICCBased }
{ file = sRGB_v4_ICC_preference.icc }
```

Still pull requests at present!

# Avoiding whatsits

In  $\text{\LaTeX}$  we can use attributes to manage color

Likely to use that for l3color on  $\text{\LaTeX}$  generally:

# Avoiding whatsits

In  $\text{\LaTeX}$  we can use attributes to manage color

Likely to use that for l3color on  $\text{\LaTeX}$  generally:  $\text{\LaTeX} 2\epsilon$  a bigger question!

# Summary

- Single set of interfaces
- Take existing ideas and bring in new features
- A chance to explore what's possible