Bespoke Multipurpose Class

BMC

*v*0.1

Pretentiousness Given Form

tecosaur
Of Github

09th June 2019

Preamble

Like with most things I didn't start out with the intent to end up this way. Initially I had a slowly growing template that I used for most documents; every so often I'd discover a package that did something I liked, or a setting that I preferred to be non-default. *Every* time that happened I'd want to go through the current documents I was working on and apply the latest revelations. Then when revisiting old documents I'd want to get them 'up to scratch'. There would always be the odd document I forgot about, or line missed, and so I quickly became tired of this process.

After realising that if I made a class and shoved it in my `texmf` directory that I'd be able to as many improvements as I like and they'd all be applied when I recompiled, *as well* as make initial configuration greatly simplified — I couldn't see a reason not to do it.

This class is very much written with my personal taste, and specific use case in mind. While I try to keep things general it is very much built around my particular perspective. As such it is reasonable to think that to the community as a whole the self-importance in the name is a tad exaggerated or undeserved. Considering that is also designed to not just convey information but also designed to visually impress, the tagline "Pretentiousness given form" seems somewhat appropriate.

I'm pleased to say that I consider this a project a success (in those respects). As I have largely drawn upon snippets of LaTeX floating around online I though the least I could do is give others that same opportunity. As such here you have an overview of my personal class designed to work for all of the documents I produce. In other words a *bespoke, multipurpose class* — or BMC for short.

Enjoy!

tecosaur

Contents

1 What This Does

Summary

1.1 Typography

1.1.1 Typefaces

This package loads three typefaces.

1. IBM Plex Serf
2. IBM Plex Sans
3. IBM Plex Mono

I wanted a selection where serif, sans, and mono all mix well. Ideally with a few weight variants. Additionally I wanted a typographic style that meshed well with the large class of documents I indented to use this for. IBM Plex seems like a good fit (For more info see section 3.2). For all three of these a linespread of 1.15 is used.

Typeface `\selectfont`	Bold `b`	Semibold `sb`	Medium `mb`	Text `tx`	Regular `m`	Light `l`	Extra L `el`	Thin `t`
Plex Serif	**Words**	**Words**	**Words**	Words	Words	Words	Words	Words
Plex Serif	***Words***	***Words***	***Words***	*Words*	*Words*	*Words*	*Words*	*Words*
Plex Sans	**Words**	**Words**	**Words**	Words	Words	Words	Words	Words
Plex Sans	***Words***	***Words***	***Words***	*Words*	*Words*	*Words*	*Words*	*Words*
Condensed	**Words**	**Words**	**Words**	Words	Words	Words	Words	Words
Condensed	***Words***	***Words***	***Words***	*Words*	*Words*	*Words*	*Words*	*Words*
Plex Mono	**Words**	**Words**	**Words**	Words	Words	Words	Words	Words
Plex Mono	***Words***	***Words***	***Words***	*Words*	*Words*	*Words*	*Words*	*Words*

1.1.2 Roman Numerals

While biblatex does provide handy roman numeral command, it's nice to have them available regardless. Hence this class provides them if they aren't already available. To get upper case roman numerals use `\RN{1978}` to produce MCMLXXVIII, and `\Rn{1978}` to produce mcmlxxviii.

```
1037  \providecommand*{\RN}[1]{\expandafter\@slowromancap\romannumeral #1@}
1038  \providecommand*{\Rn}[1]{\romannumeral#1\relax}
```

1.1.3 Fake Small Caps

Some fonts (such as IBM Plex) are not kind enough to provide small caps. Simply using downscaled capitals is a barbaric and decidedly inferior solution. So `\fauxsc{}` is

defined which, while not as nice as *true* small caps, is a darn sight better than just reducing the font size.

Barbaric Solution:	SMALL CAPS
`\fauxsc`:	SMALL CAPS

1.1.4 Penalties

The class sets new penalties.

```
482 \@beginparpenalty=10000 % don't like it when a paragraph title is on a
    ↪ different page to the start of the content
483 \hyphenpenalty=500 % not a huge fan of hyphens, but they are worthwhile
484 \righthyphenmin=4 % min letters post-hyphen
485 \lefthyphenmin=4 % min letters pre-hyphen
```

1.1.5 Captions

Caption labels are made to be upright sans-serif in the ‘text’ style, while captions are italic in the style of the body. When captions flow beyond a single line, ragged right alignment is used.

```
520 \setkomafont{caption}{\itshape\color{text}}
521 \setkomafont{captionlabel}{\fontfamily{\headingsFont}\fontseries{tx}
    ↪ \selectfont\upshape\color{text}}
522 \captionsetup{justification=raggedright,singlelinecheck=true}
```

1.2 Colour

1.2.1 Theme Colours

This class makes use of the following defined colours.

Modifying these colours in the preamble affects the entire document.

1.2.2 Colour Palette

While xcolor and latex do already come with some 'nice' shades, nice colour themes may be found at https://flatuicolors.com. These shades do not use the pretentious names listed, we just call them what they are (e.g. nephritis → green). Instead of overwriting the pre-existing colour, these colours have been differentiated by capitalisation, i.e. "Green" instead of 'green'.

1.2.3 Functional Colours

This package has a few special colours that describe a particular aspect of a document, such as `href` and `inlinemath`. For more information see subsection 2.3.1.

1.3 Mathematics

This class makes a few additions, and one or two modifications to Mathematics.

1.3.1 Modifications

Less/greater than or equal The less than or equal, and greater than or equal symbols are changed as such:

$$\begin{array}{ccc} \leq & \text{old} & \geq \\ \leqslant & \text{new} & \geqslant \end{array}$$

Inline math colour After interspersing maths and text a fair bit I've begun to think there's some merit to the Beamer 'make all maths a different colour' approach. So I've redefined the LaTeX inline math command such that `\(a^x + bx + c\)` now becomes $ax^2 + bx + c$. Avoiding this is easy, just change the colour of `inlinemath` in the preamble like so `\colorlet{inlinemath}{text}` and you won't notice this exists. For once-offs I've defined a stared variant `\(* a^x + bx + c\)` which produces the normal $ax^2 + bx + c$.

```
\renewrobustcmd{\(}{\@ifstar\@inlinemath\@@inlinemath}
\DeclareRobustCommand{\@inlinemath}{\relax\ifmmode\@badmath\else$\fi}
\DeclareRobustCommand{\@@inlinemath}{\relax\ifmmode\@badmath\else$\fi
  \color{inlinemath}}
```

Matrix environment The default for matrices (using `\begin{bmatrix}` or similar) is left aligned values, with no option to change this. This class adds an optional parameter to change the alignment, (`\begin{bmatrix}[r]`), and defaults to right aligned.

Old

$$\begin{bmatrix} 3 & -2 \\ -1 & 7 \end{bmatrix}$$

New

$$\begin{bmatrix} 3 & -2 \\ -1 & 7 \end{bmatrix}$$

1.3.2 Additions

Deliminators

`\abs{x}`	→	$\lvert x \rvert$
`\norm{x}`	→	$\lVert x \rVert$

Sets†

`\RR`	→	$\mathbb{R}$	`\RR[n]`	→	$\mathbb{R}^n$
`\NN`	→	$\mathbb{N}$	`\NN[n]`	→	$\mathbb{N}^n$
`\ZZ`	→	$\mathbb{Z}$	`\ZZ[n]`	→	$\mathbb{Z}^n$
`\QQ`	→	$\mathbb{Q}$	`\QQ[n]`	→	$\mathbb{Q}^n$
`\CC`	→	$\mathbb{C}$	`\CC[n]`	→	$\mathbb{C}^n$

Differential d

`\dd`†	→	d

Stats Operators

`\Var`	→	Var
`\Cov`	→	Cov
`\E`	→	E

Others

`\qed`†	→	■
`\qedhere`†	→	□
`\Lap`	→	$\mathcal{L}$

1.4 Code

This package spends a few lines tweaking the minted and tcolorbox config to get code blocks to look rather nice.

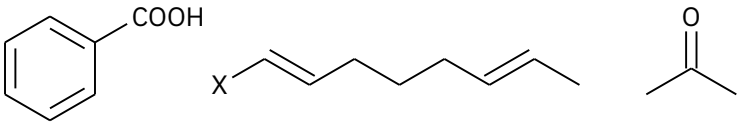
†Can also be used outside of math mode.

For example:

```tex
\section{Code}

This package spends a few lines tweaking
the minted and tcolorbox config to get code
blocks to look rather nice.

For example:
\begin{minted}[escapeinside=,highlightlines={8,17}]{tex}
    \section{Code}

    This package spends a few lines tweaking
    the minted and tcolorbox config to get code
    blocks to look rather nice.

    For example:
    ...
\end{minted}
```

1.5 Chemistry

When the `chem` option is used, `mhchem` is loaded with the configuration, however `chemfig` undergos a few modifications to make the results look cleaner.




```tex
\setchemfig{
    chemfig style={line width=0.06642 em},  % 'Line Width'
    angle increment=30,
    double bond sep=0.35700 em,  % 'Bond Spacing'
    atom sep=1.78500 em,  % 'Fixed Length'
    bond offset=0.18265 em  % 'Margin Width'
}
\renewcommand*\printatom[1]{\small\ensuremath{\mathsf{#1}}}
```

1.6 Links and Metadata

Both the `hyperref` and `hyperxmp` packages are used. The widely used hyperref package of course provides hyperlinks. This is ~~abused~~ used to add a few extra links; specificly every page number is a link to the TOC, and the text of every header links to the relevant chapter page. This allows you to jump all over the document in just a few clicks.

The hyperxmp package is rather handy for setting a few fields of pdf metadata. Using the `\title`, `\author`, and `\subtitle` attributes it sets the relevant metadata fields.

2 Boring Info

Summary

2.1 Class Options

This class builds off `scrartcl`, any other options than those listed here will just be passed through.

2.1.1 Main Styling

`dark` Switches to a dark version of the style

`solid` Uses style with solid title page, and wide stripes on chapter pages, with solid colour bar at top of pages

`stripe` Uses plain background on title page, and thin stripes on chapter pages

`article` Use `scrartcl` class instead of `scrrept`

`notes` Move the margins to make room for notes

2.1.2 Fonts

Body Text

`serif` Use serif font as main

`sans` Use sans font as main

`mono` Use mono font as main

Math

`math-serif` Same as with math option but forcing a serif font

`math-sans` Same as with math option but forcing a sans font

`math-mono` Same as with math option but forcing a mono font

2.1.3 Headings

These options set the style of the following components

- `\chapter` through to `\subparagraph`
- The page head, and page number
- Caption labels

`headings-serif` Not a default.

`headings-sans` Default when either `serif` or `sans` options are given.

`headings-mono` Default when either `mono` option is given.

2.1.4 Package Related

`chem` Load and configure `mhchem` and `chemfig` packages

`code` Load and configure minted package

`plot` Load and configure `pgfplots` package

`math` Load and configure some mathematical packages, and set font to match main text font (also see `math-serif` etc.)

2.2 Packages Used

2.2.1 Overview

Category	Packages				
General	etoolbox titlesec xcolor	xpatch titletoc tikz	Silence framed hyperref	ifdraft textpos hyperxmp	geometry calc scrlayer-scrpage
Text	microtype multicol	setspace	plex-serif	plex-sans	plex-mono
Table	booktabs	tabularx	longtable		
Graphics	graphicx	grffile	subcaption	caption	
infoBulle	infoBulle	marginInfoBulle	fontawesome5		
Chemistry	mhchem	chemfig			
Code	minted	tcolorbox			
Math	amsmath mathtools	amssymb mathastext	mathdesign pgfplots	xfrac	cancel

2.2.2 General Packages

`etoolbox` Provides LaTeX frontends to some of the new primitives provided by e-TeX as well as some rather useful some generic tools — namely,

- Robust definitions
- Command Patching
- Command Protection
- Arithmetic counters and lengths
- Document Hooks
- Environment Hooks

`xpatch` Extends the command patching provided by etoolbox

`Silence` Allows me to ignore expected warnings.

`ifdraft` To make it easy to change things up a bit more than usual for draft mode.

`scrlayer-scrpage` To allow for those lovely headers and footers.

`geometry` Loaded with options,

```
507 a4paper, ignoreheadfoot, left=\leftmargin, right=\rightmargin, top=2cm,
    ↪ bottom=3.5cm, headsep=1cm
```

`titlesec` Allows for customisation of `\chapter` etc. Was originally used for all section commands, but now all except for `\chapter` have been transitioned to KOMA-script.

`titletoc` Allows significant tweaking to how the table of contents looks.

`framed` Facilitate the definition of new environments that take multi-line material, wrap it with some non-breakable formatting (some kind of box or decoration) and allow page breaks in the material

`textpos` Facilitates placement of boxes at absolute positions on the LaTeX page. Loaded with options `absolute,overlay`

`hyperref` Used to produce all sorts of hyperlinks in a document. Loaded with option `pdfa`

`hyperxmp` Improves metadata setting with hyperref.

`calc` Adds infix expressions to perform arithmetic on the arguments of the LaTeX commands `\setcounter`, `\addtocounter`, `\setlength`, and `\addtolength`

`xcolor` Provides all sorts of colour use and mixing capabilities.

`tikz` It's tikz. You can't draw anything without it.

2.2.3 Text

`microtype` Always good to have. It simply makes text look better, specificity it applies the following,

- Character protrusion
- Font expansion
- Adjustment of interword spacing and kerning
- Letterspacing

Configured with,

```
215 activate={true,nocompatibility},final,tracking=true,kerning=true,spacing=true
```

`plex-serif`

`plex-sans`

`plex-mono`

`setspace` Provides an easy way to set line spacing with commands such as `\doublespacing` and `\setstretch{1.25}`.

`multicol` Split text into multiple columns (up to 10).

2.2.4 Table-related

`booktabs` Contribues different width `\hline` variants.

`tabularx` Adds the `tabularx` environment which has its width explicitly set, `X` column type which automatically determines its width based on its contents.

`longtable` Provides a good way of allowing tables to spread over multiple pages.

2.2.5 Graphics and Figures

`graphicx` Makes loading images (`includegraphics`) work well.

`grffile` This fixes the fix allowed filenames of graphicx.

`caption` Povides many ways to customise the captions in floating environments like figure and table, and cooperates with many other packages. Facilities include rotating captions, sideways captions, continued captions (for tables or figures that come in several parts). Loaded with option `hypcap=true`

`subcaption` Allows for typeseting of sub-figures and sub-tables.

2.2.6 infoBulle

`fontawesome5` Fontawesome 5, need I say any more?

`infoBulle`

`marginInfoBulle`

2.2.7 Chemistry

`mhchem` Useful for simple inline chemistry.

`chemfig` Useful for chemical diagrams.

2.2.8 Code

`minted` Configured as follows,

```
\setminted{
    frame=none,
    % framesep=2mm,
    baselinestretch=1.2,
    fontsize=\footnotesize,
    highlightcolor=page!95!text!80!primary,
    linenos,
    breakanywhere=true,
    breakautoindent=true,
    breaklines=true,
    tabsize=4,
    xleftmargin=3em,
    autogobble=true,
    obeytabs=true,
    python3=true,
    texcomments=true,
    framesep=2mm,
    breakbefore=\\\.+,
    breakafter=\,
}
```

`tcolorbox` Used for prettifying the `minted` environment. Loaded with option `many`

2.2.9 Math Related

These packages are loaded by the `math` option (or one of its derivatives).

`amsmath,amssymb` Extends the math commands and symbols in latex.

`mathdesign` To use the Utopia font for math symbols.

`xfrac` Allows split level fractions a/b better than `{}^{a}/{}_{b}` can produce.

`cancel` Allows for easy canceling within math like so $-\not{\pi}$ and $\not{\pi}^{0}$. Loaded with option `makeroom`

`mathtools` Provides a varienty of enhancements to make math *even* better.

- Extensible symbols, such as brackets, arrows, harpoons, etc.;
- Various symbols such as `\coloneqq` (≔);
- Easy creation of new tag forms;
- Showing equation numbers only for referenced equations;
- Extensible arrows, harpoons and hookarrows;
- Starred versions of the amsmath matrix environments for specifying the column alignment;
- More building blocks: multlined, cases-like environments, new gathered environments;
- Maths versions of `\makebox`, `\llap`, `\rlap` etc.;
- Cramped math styles; and more…

`mathastext` Uses relevant plex font for maths letters. Uses options `basic,italic,symbolgreek`.

`pgfplots` Loaded by the `plot` option.

2.3 Configuration

2.3.1 Colours

Name	Default (Light)	Default (Dark)
`text`	#000000	#FCFCFC
`page`	#FFFFFF	#222222
`href`	tertiary	secondary
`primaryVariant`	primary!75!Cream >twheel,-3,360	〃
`inlinemath`	secondary!50!text	tertiary!50!text
infoBulle		
⋮	⋮	⋮

Name	Default (Light)	Default (Dark)
⋮	⋮	⋮
`infoBulleBackground`	page!90!text	〃
`infoBulleText`	text	〃
`marginInfoBulleBackground`	page	〃
`marginInfoBulleText`	text	〃
`criticalColor`	Red	〃
`questionColor`	Purple	〃
`informationColor`	Green	〃
`checkColor`	Blue	〃
`warningColor`	Orange	〃
`tipsColor`	Purple	〃
`exampleColor`	Blue	〃
`mathematicalColor`	Orange	〃
`codeColor`	Grey	〃

3 The Whys

Summary

3.1 Layout

3.2 Typefaces

3.3 Colour