

Modern Beamer Presentations with the MTHEME package

Matthias Vogelgesang
matthias.vogelgesang@gmail.com

v0.x.x

1 Introduction

Beamer is an awesome way to make presentations with LaTeX. But the stock themes do not necessarily look particularly nice and the custom themes often scream “Beamer” at first sight. The goal of MTHEME is to provide a modern Beamer theme with minimal visual noise. It provides section slides with a neat progress bar and It is intended to be used with [Fira Sans](#), a gorgeous typeface commissioned by Mozilla and designed by [Carrois](#). Hence to get the best results you should have installed the Fira typeface and use XeTeX to typeset your slides. Nevertheless this is no hard dependency. The theme also works fine with pdfTeX and the Computer Modern typeface.

The codebase is maintained on [GitHub](#). So if you have issues, find mistakes in the manual or want to contribute – to make the theme even better – get in touch there.

2 Getting Started

2.1 Installation

The `MTHHEME` uses Make as build system. Hence the installation is very straight forward. Simply type

```
$ make
$ make install
```

in the top directory and all the files will be created and installed on your computer. The complete list of make rules is as follows:

all

Build the theme, the manual and the demo presentation.

install

Install the theme into your local texmf folder.

uninstall

Remove the theme from your local texmf folder.

sty

Build the manual.

manual

Build the manual.

demo

Build the demo presentation.

ctan

Create a package for CTAN distribution.

2.2 Dependencies

- XeLaTeX
- [Fira Sans](#) and Mono font
- TikZ

Depending on the Linux distribution, the packaged name of **Fira Sans** might be **Fira Sans OT** instead of **Fira Sans**. In that case, you may have to edit `beamerfontthememetropolis.dtx`. You may also need to install Fira Sans; see the `contrib/` directory for more. Users of Debian or Ubuntu can also install this [.deb package](#) containing the theme files as well as the Fira Sans font files.

2.3 Pandoc

To use this theme with **Pandoc**-based presentations, you can run the following command

```
$ pandoc -t beamer --latex-engine=xelatex -V theme:m -o
  output.pdf input.md
```

2.4 A Minimal Example

To get started with the theme is very simple. The following code shows a minimal example of a Beamer presentation using the `MTHEME`.

```
\documentclass[10pt]{beamer}
\usetheme{m}                % load mtheme
\title{A modern beamer theme} % define title
\date{\today}              % define date
\author{Matthias Vogelgesang} % define author
\institute{Institute}      % define institute
\begin{document}
\maketitle                 % create titlepage
\section{First Section}    % create section
\begin{frame}{First Frame} % first frame
  Lorem ipsum dolor sit amet, ...
\end{frame}
\begin{frame}{Second Frame} % second frame
  Lorem ipsum dolor sit amet, ...
\end{frame}
```

```
\end{document}
```

3 Customization

3.1 Package options

The theme provides a number of options. To use any of the options below, call them when invoking `MTHEME` in the preamble of the slides, i.e.

```
\usetheme[<options>]{m}
```

<code>usetitleprogressbar</code>	Adds a thin progress bar similar to the section progress bar underneath each frame title.
<code>blockbg</code>	Adds background color to the blocks similar to other beamer themes.
<code>nooffset</code>	By default, the <code>MTHEME</code> adds <code>\vspace{2em}</code> after the <code>frametitle</code> to center content vertically on the frame. This option removes this additional space in order to get more content per slide.
<code>nosectionslide</code>	By default when using the <code>\section</code> command, a slide is created with just the title and the progress bar on it. This option prevents the creation of these additional slides.
<code>usetotalslideindicator</code>	By default, only the current page number is printed in the lower right corner. This option changes the slide numbering format to <code>#current/#total</code> .
<code>noslidenumbers</code>	Omits slide numbers entirely.
<code>darkcolors</code>	Makes the background dark and the foreground light.

3.2 Color Customization

The included metropolis color theme is used by default, but its colors can be easily changed to suit your tastes. All of the theme's styles are defined in terms of three beamer colors:

- `normal text` (dark fg, light bg)
- `alerted text` (colored fg, should be visible against dark or light)
- `example text` (colored fg, should be visible against dark or light)

An easy way to customize the theme is to redefine these colors using

```
\setbeamercolor{ ... }{ fg= ... , bg= ... }
```

in your preamble. For greater customization, you can redefine any of the other colors in `beamercolorthememetropolis`, including progress bar.

3.3 Title Case Formatting

The main title, section titles, frame titles and plain frame titles are all formatted according to the custom command `\mthemetitleformat`. By default, this is equivalent to `\MakeLowercase{#1}`, hence setting the titles in small capitals. You can change this behaviour in your preamble. For example:

```
% camel case
\renewcommand{\mthemetitleformat}{}
% lowercase
\renewcommand{\mthemetitleformat}{\MakeLowercase}
% uppercase
\renewcommand{\mthemetitleformat}{\MakeUppercase}
```

Be aware that these formatting macros will be replaced with theme options in the future.

3.4 Commands

The `\plain{title=[]}{<body>}` command sets a slide in plain dark colors which can be useful to focus attention on a single sentence or image.

3.5 Paul Tol's colors: a `pgfplots` theme

A good presentation uses colors that are

- distinct from each other as much as possible, and
- distinct from black and white,
- under many different lighting and display environments, and
- to color-blind viewers,
- all while matching well together.

In a [technical note](#) for SRON, Paul Tol proposed a palette of colors satisfying these constraints. The sub-package `pgfplotsthemetol` defines palettes for `pgfplots` charts based on Tol's work. Use the `mlineplot` key to plot line data and `mbarplot` or horizontal `mbarplot` to plot bar charts.

4 Known Issues

5 License

The theme itself is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#). This means that if you change the theme and re-distribute it, you must retain the copyright notice header and license it under the same CC-BY-SA license. This does not affect the presentation that you create with the theme.

6 Contributors

For a full list of contributors please visit the [GitHub Repository](#).

7 Implementation

8 Implementation: METROPOLIS main theme

The primary job of this package is to load the component sub-packages of the METROPOLIS theme and route the theme options accordingly. It also provides some custom commands and environments for the user.

Options

```
1 \newif\if@useTitleProgressBar
2 \@useTitleProgressBarfalse
3 \DeclareOptionBeamer{usetitleprogressbar}{
4   \@useTitleProgressBartrue
5 }
```

usetotalslideindicator

```
6 \newif\if@useTotalSlideIndicator
7 \@useTotalSlideIndicatorfalse
8 \DeclareOptionBeamer{usetotalslideindicator}{
9   \@useTotalSlideIndicatortrue
10 }
```

noslidenumbers

```
11 \newif\if@noSlideNumbers
12 \@noSlideNumbersfalse
13 \DeclareOptionBeamer{noslidenumbers}{
14   \@noSlideNumberstrue
15 }
```

nosectionslide

```
16 \newif\if@noSectionSlide
17 \@noSectionSlidefalse
18 \DeclareOptionBeamer{nosectionslide}{
19   \@noSectionSlidetrue
20 }
```

nooffset

```

21 \newlength{\@mtheme@voffset}
22 \setlength{\@mtheme@voffset}{2em}
23 \DeclareOptionBeamer{nooffset}{
24   \setlength{\@mtheme@voffset}{0em}
25 }

```

blockbg

```

26 \newif\if@beamer@metropolis@blockbg
27 \@beamer@metropolis@blockbgfalse
28 \DeclareOptionBeamer{blockbg}{
29   \@beamer@metropolis@blockbgtrue
30 }

```

darkcolors

```

31 \DeclareOptionBeamer{darkcolors}{
32   \PassOptionsToPackage{darkcolors}{beamercolorthememetropolis}%
33 }

```

Unknown option error handling

```

34 \DeclareOptionBeamer*{
35   \PackageWarning{beamerthemem}{Unknown option '\CurrentOption'}%
36 }
37 \ProcessOptionsBeamer

```

8.1 Component sub-packages

Having processed the options, we can now load the component sub-packages of the theme.

```

38 \useinnertheme{metropolis}
39 \useoutertheme{metropolis}
40 \usecolortheme{metropolis}

```

The `fira` font theme, which depends on `fontspec`, is only loaded if the document is being processed by Xe_{La}TeX or Lua_{La}TeX.

```

41 \RequirePackage{etoolbox}
42 \RequirePackage{ifxetex,ifluatex}
43 \ifboolexpr{bool {xetex} or bool {luatex}}{

```



```

44 \usefonttheme{metropolis}
45 }{
46 \PackageWarning{beamerthemem}{%
47   You need to compile with XeLaTeX or LuaLaTeX to use the Fira fonts.
48 }
49 }

```

The `tol` theme for `pgfplots` is only loaded if `pgfplots` is used.

```

50 \AtEndPreamble{%
51   \@ifpackageloaded{pgfplots}{%
52     \RequirePackage{pgfplotsthemetol}
53   }}
54 }

```

8.2 Custom commands

We define custom commands in this package as their proper usage may depend on multiple sub-packages.

<pre> \mthemetitleformat \mthemesectiontitleformat \mthemeframetitleformat \mthemeplaintitleformat </pre>	<pre> Creates hooks to change the case format of the four different titles. 55 \def\mthemetitleformat#1{\MakeLowercase{#1}} 56 \def\mthemesectiontitleformat#1{\mthemetitleformat{#1}} 57 \def\mthemeframetitleformat#1{\mthemetitleformat{#1}} 58 \def\mthemeplaintitleformat#1{\mthemetitleformat{#1}} </pre>
---	---

To give users the option to `\MakeUppercase` or `\MakeLowercase` the section title and frame title we need to patch the commands `\sectionentry`, `\beamer@section` and `\beamer@@frametitle`. This solution was suggested by Enrico Gregorio in an answer to [this StackExchange question](#).

```

59 \patchcmd{\sectionentry}
60   {\def\insertsectionhead{#2}}
61   {\def\insertsectionhead{\mthemesectiontitleformat{#2}}}
62   {}{}
63 \patchcmd{\beamer@section}
64   {\def\insertsectionhead{\hyperlink{Navigation\the\c@page}{#1}}}
65   {\def\insertsectionhead{\hyperlink{Navigation\the\c@page}{\mthemesectiontitleform
66   {}{}

```

```

67
68 \patchcmd{\beamer@frametitle}
69   {\beamer@ifempty{#2}}{}{%
70     \gdef\insertframetitle{#{2}\ifnum\beamer@autobreakcount>0\relax{ }\space\usebeamer@
71     continuation}\fi}}%
72   \gdef\beamer@frametitle{#2}%
73   \gdef\beamer@shortframetitle{#1}%
74   }}
75   {\beamer@ifempty{#2}}{}{%
76     \gdef\insertframetitle{{\mthemeframetitleformat{#2}\ifnum\beamer@autobreakcount>0\relax{ }\space\usebeamer@
77     continuation}\fi}}%
78   \gdef\beamer@frametitle{#2}%
79   \gdef\beamer@shortframetitle{#1}%
80   }}
81   {}{}

```

`\plain` Creates a plain frame with dark background, suitable for displaying images or a few words.

```

80 \newcommand{\plain}[2][{}]{%
81   \begin{group}
82     \setbeamercolor{background canvas}{use=palette primary,parent=palette primary}
83     \begin{frame}{#1}
84       \centering
85       \vfill
86       \vspace{1em}
87       \usebeamercolor[fg]{palette primary}
88       \usebeamerfont{section title}
89       \mthemeplaintitleformat{#2}
90       \vfill
91     \end{frame}
92   \end{group}
93 }

```

`\mreducelistspacing`

```

94 \newcommand{\mreducelistspacing}{\vspace{-\topsep}}

```

9 Implementation: METROPOLIS inner theme

A `beamer` inner theme dictates the style of the frame elements traditionally set in the “body” of each slide. These include:

- title, part, and section pages;
- itemize, enumerate, and description environments;
- block environments including theorems and proofs;
- figures and tables; and
- footnotes and plain text.

9.1 Title page

`title page` Template for the title page.

```
95 \RequirePackage{tikz}
96 \setbeamertemplate{title page}{
97   \begin{minipage}[b][\paperheight]{\textwidth}
```

If the user has set a `titlegraphic`, we set it in a zero-height box so it doesn’t change the position of other elements.

```
98   \ifx\inserttitlegraphic\@empty\else{%
99     \vbox to 0pt {
100       \vspace*{2em}
101       \usebeamercolor[fg]{titlegraphic}%
102       \inserttitlegraphic%
103     }%
104     \nointerlineskip%
105   }
106   \fi
107   \vfill%
```

We set the title and subtitle, but only if they are defined by the user. If `\subtitle` is empty, for example, it won’t leave a blank space on the title slide.

```
108   \ifx\inserttitle\@empty\else{ {%
109     \raggedright%
110     \linespread{1.0}%
111     \usebeamerfont{title}%
```

```

112     \usebeamercolor[fg]{title}%
113     \mthemetitleformat{\inserttitle}%
114     \par%
115     \vspace*{0.5em}
116   }}
117   \fi
118   \ifx\insertsubtitle\@empty\else{%
119     \usebeamerfont{subtitle}%
120     \usebeamercolor[fg]{subtitle}%
121     \insertsubtitle%
122     \par%
123     \vspace*{0.5em}
124   }}
125   \fi

```

A horizontal rule (drawn in TikZ) separates the title and subtitle from the author, date, and institution.

```

126   \begin{tikzpicture}
127     \usebeamercolor{title separator}
128     \draw[fg] (0, 0) -- (\textwidth, 0);
129   \end{tikzpicture}%
130   \par%
131   \vspace*{1em}%

```

Like the title and subtitle, we display the author only when it is defined. But beamer's definition of `\insertauthor` is always nonempty, so we have to test another macro initialized by `\author{...}` to see if the user has defined an author. This solution was suggested by Enrico Gregorio in an answer to [this Stack Exchange question](#).

```

132   \ifx\beamer@shortauthor\@empty\else{%
133     \usebeamerfont{author}%
134     \usebeamercolor[fg]{author}%
135     \insertauthor%
136     \par%
137     \vspace*{0.25em}
138   }}
139   \fi

```

The date and institute are set after the author, again provided they are nonempty.

Note that the default date in \TeX is `\today`, not `\empty`.

```
140 \ifx\insertdate\@empty\else{%
141   \usebeamerfont{date}%
142   \usebeamercolor[fg]{date}%
143   \insertdate%
144   \par%
145 }}
146 \fi
147 \ifx\insertinstitute\@empty\else{%
148   \vspace*{3mm}
149   \usebeamerfont{institute}%
150   \usebeamercolor[fg]{institute}%
151   \insertinstitute%
152   \par%
153 }}
154 \fi
155 \vfill
156 \vspace*{1mm}
157 \end{minipage}
158 }
```

Normal people should use `\maketitle` or `\titlepage` instead of using the `title page` beamer template directly. Beamer already defines these macros, but we patch them here to make the title page `[plain]` by default, remove `\@thanks`, and ensure the title frame number doesn't count.

`\maketitle` Inserts the title frame, or causes the current frame to use the `title page` template.
`\titlepage` plate.

```
159 \def\maketitle{%
160   \ifbeamer@inframe
161     \titlepage
162   \else
163     \frame[plain]{\titlepage}
164   \fi
165 }
166 \def\titlepage{%
167   \usebeamertemplate{title page}
168 }
```

9.2 Section page

`section page` Template for the section title slide at the beginning of each section.

```
169 \setbeamertemplate{section page}{
170   \vspace{2em}
171   \centering
172   \begin{minipage}{22em}
173     \usebeamercolor[fg]{section title}
174     \usebeamerfont{section title}
175     \insertsectionhead\[-1ex]
176     \usebeamertemplate*{progress bar in section page}
177   \end{minipage}
178   \par
179 }
180 \if@noSectionSlide\else%
181   \AtBeginSection{
182     \ifbeamer@inframe
183       \sectionpage
184     \else
185       \frame[plain,c]{\sectionpage}
186     \fi
187   }
188 \fi
```

`progress bar in section page` Template for the progress bar displayed by default on the section page. This code is duplicated in large part in the outer theme's template `progress bar in head-
/foot`.

```
189 \RequirePackage{calc}
190 \newlength{\metropolis@progressonsectionpage}
191 \setbeamertemplate{progress bar in section page}{
192   \setlength{\metropolis@progressonsectionpage}{%
193     \textwidth * \ratio{\insertframenumber pt}{\inserttotalframenumber pt}}%
194   }%
195   \begin{tikzpicture}
196     \draw[bg, fill=bg] (0,0) rectangle (\textwidth, 0.4pt);
197     \draw[fg, fill=fg] (0,0) rectangle (\metropolis@progressonsectionpage, 0.4pt);
198   \end{tikzpicture}%
199 }
```

The above code assumes that `\insertframenum` is less than or equal to `\inserttotalframenum`. However, this is not true on the first compile; in the absence of an `.aux` file, `\inserttotalframenum` defaults to 1. This behaviour could cause fatal errors for long presentations, as `\metropolis@progressonsectionpage` would exceed T_EX's maximum length (16383.99999pt, roughly 5.75 metres or 18.9 feet). To avoid this, we increase the default value for `\inserttotalframenum`; presentations with over 4000 slides will still break on first compile, but users in that situation likely have deeper problems to solve.

```
200 \def\inserttotalframenum{100}
```

9.3 Block environments

```
201 \newlength{\leftrightskip}
202 \if@beamer@metropolis@blockbg
203   \setlength{\leftrightskip}{1ex}
204 \else
205   \setlength{\leftrightskip}{0ex}
206 \fi
207 \setbeamertemplate{block begin}{%
208   \vspace*{1ex}
209   \begin{beamercolorbox}[%
210     ht=2.4ex,
211     dp=1ex,
212     leftskip=\leftrightskip,
213     rightskip=\leftrightskip]{block title}
214     \usebeamerfont*{block title}\insertblocktitle%
215   \end{beamercolorbox}%
216   \vspace*{-1pt}
217   \usebeamerfont{block body}%
218   \begin{beamercolorbox}[%
219     dp=1ex,
220     leftskip=\leftrightskip,
221     rightskip=\leftrightskip,
222     vmode]{block body}%
223 }
224 \setbeamertemplate{block end}{%
225   \end{beamercolorbox}
226   \vspace*{0.2ex}
```

227 }

Alerted block environment

```
228 \setbeamertemplate{block alerted begin}{%
229   \vspace*{1ex}
230   \begin{beamercolorbox}[%
231     ht=2.4ex,
232     dp=1ex,
233     leftskip=\leftskip,
234     rightskip=\rightskip]{block title alerted}
235     \usebeamerfont*{block title alerted}\insertblocktitle%
236   \end{beamercolorbox}%
237   \vspace*{-1pt}
238   \usebeamerfont{block body alerted}%
239   \begin{beamercolorbox}[%
240     dp=1ex,
241     leftskip=\leftskip,
242     rightskip=\rightskip,
243     vmode]{block body}%
244 }
245 \setbeamertemplate{block alerted end}{%
246   \end{beamercolorbox}
247   \vspace*{0.2ex}
248 }
```

Example block environment

```
249 \setbeamertemplate{block example begin}{%
250   \vspace*{1ex}
251   \begin{beamercolorbox}[%
252     ht=2.4ex,
253     dp=1ex,
254     leftskip=\leftskip,
255     rightskip=\rightskip]{block title example}
256     \usebeamerfont*{block title example}\insertblocktitle%
257   \end{beamercolorbox}%
258   \vspace*{-1pt}
259   \usebeamerfont{block body example}%
260   \begin{beamercolorbox}[%
261     dp=1ex,
262     leftskip=\leftskip,
```



```

263     rightskip=\leftrightskip,
264     vmode]{block body}%
265 }
266 \setbeamertemplate{block example end}{%
267   \end{beamercolorbox}
268   \vspace*{0.2ex}
269 }

```

9.4 Itemize/enumerate environments

```

270 \setlength{\leftmargini}{1em}
271 \setlength{\leftmarginii}{1em}
272 \setlength{\leftmarginiii}{1em}
273 \setbeamertemplate{itemize item}{\textbullet}
274 \setbeamertemplate{itemize subitem}{\textbullet}
275 \setbeamertemplate{itemize subsubitem}{\textbullet}

```

9.5 Figures and tables

```

276 \setbeamertemplate{caption label separator}{: }
277 \setbeamertemplate{caption}[numbered]

```

9.6 Footnotes

```

278 \setbeamertemplate{footnote}{%
279   \parindent 0em\noindent%
280   \raggedright
281   \usebeamercolor{footnote}\hbox to 0.8em{\hfil\insertfootnotemark}\insertfootnotetext}
282 }

```

9.7 General text

```

283 \mode<all>
284 \setlength{\parskip}{0.5em}
285 \linespread{1.15}

```

10 Implementation: METROPOLIS outer theme

A `beamer` outer theme dictates the style of the frame elements traditionally set outside the body of each slide: the head, footline, and frame title.

This customization will be removed in a future version.

```
286 \def\mthemetitleformat{\scshape\MakeLowercase}
```

10.1 Head and footline

All good `beamer` presentations should already remove the navigation symbols, but `METROPOLIS` removes them automatically (just in case).

```
287 \setbeamertemplate{navigation symbols}{}
```

The only element in the footline by default is the frame number. It can optionally be omitted or displayed as a fraction of the total frames.

```
288 \setbeamertemplate{footline}{%
289   \begin{beamercolorbox}[%
290     wd=\textwidth,
291     ht=3ex,
292     dp=3ex,
293     leftskip=0.3cm,
294     rightskip=0.3cm
295   ]{footline}%
296   \hfill\usebeamerfont{page number in head/foot}%
297   \if@noSlideNumbers%
298     %Purposefully left blank to display no slide number.%
299   \else%
300     \if@useTotalSlideIndicator%
301     \insertframenumbers/\inserttotalframenumbers%
302     \else%
303     \insertframenumbers%
304     \fi%
305   \fi%
306 \end{beamercolorbox}%
307 }
```

10.2 Frametitle

`frametitle` Template for the frame title, which is optionally underlined with a progress bar.

```

308 \setbeamertemplate{frametitle}{%
309   \nointerlineskip
310   \begin{beamercolorbox}[%
311     wd=\paperwidth,
312     leftskip=0.3cm,
313     rightskip=0.3cm,
314     ht=2.5ex,
315     dp=1.5ex
316   ]{frametitle}
317   \insertframetitle%
318   \end{beamercolorbox}%
319   \if@useTitleProgressBar
320     \nointerlineskip
321     \usebeamertemplate*{progress bar in head/footer}
322   \fi
323   \vspace{\@mtheme@voffset}
324 }

```

`progress bar in head/footer` Template for the progress bar optionally displayed below the frame title on each page. Much of this code is duplicated in the inner theme's template `progress bar in section page`.

```

325 \RequirePackage{calc}
326 \newlength{\metropolis@progressinheadfoot}
327 \setbeamertemplate{progress bar in head/footer}{
328   \setlength{\metropolis@progressinheadfoot}{%
329     \paperwidth * \ratio{\insertframenum pt}{\inserttotalframenum pt}%
330   }%
331   \begin{beamercolorbox}[wd=\paperwidth,ht=0.4pt,dp=0pt]{progress bar in head-
332     /foot}
333     \begin{tikzpicture}
334       \draw[bg, fill=bg] (0,0) rectangle (\paperwidth, 0.4pt);
335       \draw[fg, fill=fg] (0,0) rectangle (\metropolis@progressinheadfoot, 0.4pt);
336     \end{tikzpicture}%
337   \end{beamercolorbox}

```

11 Implementation: Fira font theme

Font Definitions

```
338 \RequirePackage[no-math]{fontspec}
339 \defaultfontfeatures{Mapping=tex-text}
340 \setsansfont[BoldItalicFont={Fira Sans Italic},%
341             ItalicFont={Fira Sans Light Italic},%
342             BoldFont={Fira Sans}]{Fira Sans Light}
343 \setmonofont{Fira Mono}
344 \newfontfamily\ExtraLight{Fira Sans ExtraLight}
345 \newfontfamily\Light{Fira Sans Light}
346 \newfontfamily\Book{Fira Sans}
347 \newfontfamily\Medium{Fira Sans Medium}
348 \AtBeginEnvironment{tabular}{%
349     \setsansfont[BoldFont={Fira Sans},%
350                 Numbers={Monospaced}]{Fira Sans Light}%
351 }
```

Font Assignment

```
352 \setbeamerfont{title}{family=\Book, size=\Large, shape=\scshape}
353 \setbeamerfont{author}{family=\ExtraLight, size=\small}
354 \setbeamerfont{date}{family=\ExtraLight, size=\small}
355 \setbeamerfont{section title}{family=\Book, size=\Large, shape=\scshape}
356 \setbeamerfont{block title}{family=\Book, size=\normalsize}
357 \setbeamerfont{block title alerted}{family=\Book, size=\normalsize}
358 \setbeamerfont{subtitle}{family=\Light, size=\fontsize{12}{14}}
359 \setbeamerfont{frametitle}{family=\Book, size=\large, shape=\scshape}
360 \setbeamerfont{caption}{size=\small}
361 \setbeamerfont{caption name}{family=\Book}
362 \setbeamerfont{description item}{family=\Book}
363 \setbeamerfont{page number in head/foot}{size=\scriptsize}
```

Bibliography

```
364 \setbeamerfont{bibliography entry author}{family=\Light, size=\normalsize}
365 \setbeamerfont{bibliography entry title}{family=\Book, size=\normalsize}
366 \setbeamerfont{bibliography entry location}{family=\Light, size=\normalsize}
367 \setbeamerfont{bibliography entry note}{family=\Light, size=\small}
368 \linespread{1.15}
```

12 Implementation: METROPOLIS color theme

Options

darkcolors

```
369 \newif\if@beamer@metropolis@darkcolors
370 \@beamer@metropolis@darkcolorsfalse
371 \DeclareOptionBeamer{darkcolors}{
372   \@beamer@metropolis@darkcolorstrue
373 }
```

Unknown option error handling

```
374 \DeclareOptionBeamer*{%
375   \PackageWarning{beamercolorthememetropolis}{Unknown option `\'CurrentOption'}%
376 }
377 \ProcessOptionsBeamer
```

12.1 Base colors

```
378 \definecolor{mDarkBrown}{HTML}{604c38}
379 \definecolor{mDarkTeal}{HTML}{23373b}
380 \definecolor{mLightBrown}{HTML}{EB811B}
381 \definecolor{mLightGreen}{HTML}{14B03D}
```

12.2 Base styles

All colors in the METROPOLIS theme are derived from the definitions of `normal text`, `alerted text`, and `example text`.

```
382 \if@beamer@metropolis@darkcolors
383   \setbeamercolor{normal text}{%
384     fg=black!2,
385     bg=mDarkTeal
386   }
387 \else
388   \setbeamercolor{normal text}{%
389     fg=mDarkTeal,
390     bg=black!2
391   }
```

```

392 \fi
393 \setbeamercolor{alerted text}{%
394   fg=mLightBrown
395 }
396 \setbeamercolor{example text}{%
397   fg=mLightGreen
398 }

```

12.3 Derived colors

The titles and structural elements (e.g. `itemize` bullets) are set in the same color as `normal text`. This would ideally be done by setting `normal text` as a parent style, which we do to set `titlelike`, but this doesn't work for `structure` as its foreground is set explicitly in `beamercolorthemedefault.sty`.

```

399 \setbeamercolor{titlelike}{%
400   use=normal text,
401   parent=normal text
402 }
403 \setbeamercolor{structure}{%
404   fg=normal text.fg
405 }

```

The “primary” palette should be used for the most important navigational elements, and possibly of other elements. The METROPOLIS theme uses it for frame titles and slides.

```

406 \setbeamercolor{palette primary}{%
407   use=normal text,
408   fg=normal text.bg,
409   bg=normal text.fg
410 }
411 \setbeamercolor{frametitle}{%
412   use=palette primary,
413   parent=palette primary
414 }

```

The METROPOLIS inner or outer themes optionally display progress bars in various locations. Their color is set by `progress bar` but the two different kinds can be

customized separately. The horizontal rule on the title page is also set based on the progress bar color and can be customized with `title separator`.

```
415 \setbeamercolor{progress bar}{%
416   use=alerted text,
417   fg=alerted text.fg,
418   bg=normal text.bg!50!normal text.fg
419 }
420 \setbeamercolor{title separator}{
421   use=progress bar,
422   parent=progress bar
423 }
424 \setbeamercolor{progress bar in head/foot}{%
425   use=progress bar,
426   parent=progress bar
427 }
428 \setbeamercolor{progress bar in section page}{
429   use=progress bar,
430   parent=progress bar
431 }
```

Blocks

```
432 \if@beamer@metropolis@blockbg
433   \setbeamercolor{block title}{%
434     use=normal text,
435     fg=normal text.fg,
436     bg=normal text.bg!80!fg
437   }
438 \else
439   \setbeamercolor{block title}{use=normal text, parent=normal text}
440 \fi
441 \setbeamercolor{block title alerted}{%
442   use={block title, alerted text},
443   bg=block title.bg,
444   fg=alerted text.fg
445 }
446 \setbeamercolor{block title example}{%
447   use={block title, example text},
448   bg=block title.bg,
449   fg=example text.fg
```

```

450 }
451 \setbeamercolor{block body alerted}{use=block body, parent=block body}
452 \setbeamercolor{block body example}{use=block body, parent=block body}
453 \setbeamercolor{block body}{
454   use={block title, normal text},
455   bg=block title.bg!50!normal text.bg
456 }

```

Footnotes

```

457 \setbeamercolor{footnote}{fg=normal text.fg!90}
458 \setbeamercolor{footnote mark}{fg=.}
459 \mode<all>

```

13 Implementation: Tol pgfplots theme

Paul Tol's 12-color palette¹ is as follows:

```

460 \definecolor{TolDarkPurple}{HTML}{332288}
461 \definecolor{TolDarkBlue}{HTML}{6699CC}
462 \definecolor{TolLightBlue}{HTML}{88CCEE}
463 \definecolor{TolLightGreen}{HTML}{44AA99}
464 \definecolor{TolDarkGreen}{HTML}{117733}
465 \definecolor{TolDarkBrown}{HTML}{999933}
466 \definecolor{TolLightBrown}{HTML}{DDCC77}
467 \definecolor{TolDarkRed}{HTML}{661100}
468 \definecolor{TolLightRed}{HTML}{CC6677}
469 \definecolor{TolLightPink}{HTML}{AA4466}
470 \definecolor{TolDarkPink}{HTML}{882255}
471 \definecolor{TolLightPurple}{HTML}{AA4499}

```

To use these colors, we describe “cycle lists” from which PGF chooses styles for the different series in a chart.

mbarplot cycle Colors and styles intended for bar charts with up to 12 series.

```

472 \pgfplotscreateplotcyclelist{mbarplot cycle}{%
473   {draw=TolDarkBlue, fill=TolDarkBlue!70},

```

¹Tol actually describes several palettes; these colours are taken from the bottom row of Figure 3 in his technical note.


```

474 {draw=TolLightBrown, fill=TolLightBrown!70},
475 {draw=TolLightGreen, fill=TolLightGreen!70},
476 {draw=TolDarkPink, fill=TolDarkPink!70},
477 {draw=TolDarkPurple, fill=TolDarkPurple!70},
478 {draw=TolDarkRed, fill=TolDarkRed!70},
479 {draw=TolDarkBrown, fill=TolDarkBrown!70},
480 {draw=TolLightRed, fill=TolLightRed!70},
481 {draw=TolLightPink, fill=TolLightPink!70},
482 {draw=TolLightPurple, fill=TolLightPurple!70},
483 {draw=TolLightBlue, fill=TolLightBlue!70},
484 {draw=TolDarkGreen, fill=TolDarkGreen!70},
485 }

```

`mlineplot cycle` Colors and styles intended for line charts with up to 4 series.

```

486 \pgfplotscreateplotcyclelist{mlineplot cycle}{%
487 {TolDarkBlue, mark=*, mark size=1.5pt},
488 {TolLightBrown, mark=square*, mark size=1.3pt},
489 {TolLightGreen, mark=triangle*, mark size=1.5pt},
490 {TolDarkBrown, mark=diamond*, mark size=1.5pt},
491 }

```

However, the above cycle lists are not applied automatically. We still need to define styles — `mlineplot` and `mbarplot` — that the user can apply to the axis of a `pgfplots` chart to use the colors. We'll also take the opportunity to adjust the display of chart axes when these styles are used.

```

492 \pgfplotsset{
493   compat=1.9,

```

`mlineplot` A style to apply to the axis of a PGF line plot.

```

494 mlineplot/.style={
495   mbaseplot,
496   xmajorgrids=true,
497   ymajorgrids=true,
498   major grid style={dotted},
499   axis x line=bottom,
500   axis y line=left,
501   legend style={
502     cells={anchor=west},

```

```

503     draw=none
504   },
505   cycle list name=mlineplot cycle,
506 },

```

mbarplot A style to apply to the axis of a PGF bar chart. **mbarplot** uses vertical bars by default, while **horizontal mbarplot** has horizontal bars as the name implies. Their shared properties are factored out into the internal style **mbarplot base**.

```

507 mbarplot base/.style={
508   mbaseplot,
509   bar width=6pt,
510   axis y line*=none,
511 },
512 mbarplot/.style={
513   mbarplot base,
514   ybar,
515   xmajorgrids=false,
516   ymajorgrids=true,
517   area legend,
518   legend image code/.code={%
519     \draw[#1] (0cm,-0.1cm) rectangle (0.15cm,0.1cm);
520   },
521   cycle list name=mbarplot cycle,
522 },
523 horizontal mbarplot/.style={
524   mbarplot base,
525   xmajorgrids=true,
526   ymajorgrids=false,
527   xbar stacked,
528   area legend,
529   legend image code/.code={%
530     \draw[#1] (0cm,-0.1cm) rectangle (0.15cm,0.1cm);
531   },
532   cycle list name=mbarplot cycle,
533 },

```

mbaseplot Adjusts the appearance of the axes in a PGF chart.

```

534 mbaseplot/.style={
535   legend style={

```

```

536     draw=none,
537     fill=none,
538     cells={anchor=west},
539   },
540   x tick label style={
541     font=\footnotesize
542   },
543   y tick label style={
544     font=\footnotesize
545   },
546   legend style={
547     font=\footnotesize
548   },
549   major grid style={
550     dotted,
551   },
552   axis x line*=bottom,
553 },
554 disable thousands separator/.style={
555   /pgf/number format/.cd,
556     1000 sep={}
557 },
558 }

```