

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 `MTHOME` 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 `\@metropolis@titleformat`. 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{\@metropolis@titleformat}{}
% lowercase
\renewcommand{\@metropolis@titleformat}{\MakeLowercase}
% uppercase
\renewcommand{\@metropolis@titleformat}{\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.

Load the required packages.

```
1 \RequirePackage{etoolbox}
2 \RequirePackage{pgfopts}
3 \RequirePackage{ifxetex}
4 \RequirePackage{ifluatex}
```

8.1 Options

Set default values for options.

```
5 \newcommand{\@metropolis@setdefaults}{
6   \pgfkeys{/metropolis/.cd,}
7 }
```

8.2 Component sub-packages

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

```
8 \useinnertheme[sectionpage=none]{metropolis}
9 \useoutertheme[numbering=fraction, progressbar=head]{metropolis}
10 \usecolortheme{metropolis}
```

The **fira** font theme, which depends on **fontspec**, is only loaded if the document is being processed by Xe_ΛTeX or Lua_ΛTeX.

```
11 \ifboolexpr{bool {xetex} or bool {luatex}}{
12   \usefonttheme{metropolis}
13 }
```

```

14 \PackageWarning{beamerthemem}{%
15     You need to compile with XeLaTeX or LuaLaTeX to use the Fira fonts.
16 }
17 }

```

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

```

18 \AtEndPreamble{%
19     \ifpackageloaded{pgfplots}{%
20         \RequirePackage{pgfplotssthemetol}
21     }{}
22 }

```

8.3 Custom commands

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

<code>\@metropolis@titleformat</code> <code>metropolis@sectiontitleformat</code> <code>metropolis@frametitleformat</code> <code>metropolis@plaintitleformat</code>	Creates hooks to change the case format of the four different titles. <pre> 23 \def\@metropolis@titleformat#1{\MakeLowercase{#1}} 24 \def\@metropolis@sectiontitleformat#1{\@metropolis@titleformat{#1}} 25 \def\@metropolis@frametitleformat#1{\@metropolis@titleformat{#1}} 26 \def\@metropolis@plaintitleformat#1{\@metropolis@titleformat{#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](#).

```

27 \patchcmd{\sectionentry}
28     {\def\insertsectionhead{#2}}
29     {\def\insertsectionhead{\metropolissectiontitleformat{#2}}}
30     {}{}
31 \patchcmd{\beamer@section}
32     {\def\insertsectionhead{\hyperlink{Navigation\the\c@page}{#1}}}
33     {\def\insertsectionhead{\hyperlink{Navigation\the\c@page}{\metropolissectiontitle
34     }}}
35
36 \patchcmd{\beamer@@frametitle}

```



```

37 {\beamer@ifempty{#2}{}{}}%
38 \gdef\insertframetitle{{#2\ifnum\beamer@autobreakcount>0\relax{}}\space\usebea
tinuation}\fi}}%
39 \gdef\beamer@frametitle{#2}%
40 \gdef\beamer@shortframetitle{#1}%
41 }}
42 {\beamer@ifempty{#2}{}{}}%
43 \gdef\insertframetitle{{\@metropolis@frametitleformat{#2}\ifnum\beamer@autobr
tinuation}\fi}}%
44 \gdef\beamer@frametitle{#2}%
45 \gdef\beamer@shortframetitle{#1}%
46 }}
47 {}{}

```

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

```

48 \newcommand{\plain}[2][{}]{%
49 \begin{group
50 \setbeamercolor{background canvas}{use=palette primary,parent=palette pri-
mary}
51 \begin{frame}{#1}
52 \centering
53 \vfill
54 \vspace{1em}
55 \usebeamercolor[fg]{palette primary}
56 \usebeamerfont{section title}
57 \metropolisplaintitleformat{#2}
58 \vfill
59 \end{frame}
60 \endgroup
61 }

```

\mreducelistspacing

```

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

```

Process package options

```

63 \@metropolis@setdefaults
64 \ProcessPgfPackageOptions{/metropolis}

```

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.

Load required packages.

```
65 \RequirePackage{calc}
66 \RequirePackage{pgfopts}
67 \RequirePackage{tikz}
```

9.1 Options

block This option controls the block style.

```
68 \pgfkeys{
69   /metropolis/inner/block/.cd,
70   .is choice,
71   transparent/.code=\setlength{\@metropolis@blockskip}{0ex},
72   fill/.code=\setlength{\@metropolis@blockskip}{1ex},
73 }
```

sectionpage The **sectionpage** option defines the behaviour of the sectionpage.

```
74 \pgfkeys{
75   /metropolis/inner/sectionpage/.cd,
76   .is choice,
77   none/.code=\@metropolis@sectionpage@none,
78   progressbar/.code=\@metropolis@sectionpage@progressbar,
79 }
```

Set default values for options.

```
80 \newcommand{\@metropolis@inner@setdefaults}{}
```

```

81 \pgfkeys{/metropolis/inner/.cd,
82   sectionpage=progressbar,
83   block=transparent,
84 }
85 }

```

9.2 Title page

`title page` Template for the title page.

```

86 \setbeamertemplate{title page}{
87   \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.

```

88   \ifx\inserttitlegraphic\@empty\else{%
89     \vbox to 0pt {
90       \vspace*{2em}
91       \usebeamercolor[fg]{titlegraphic}%
92       \inserttitlegraphic%
93     }%
94     \nointerlineskip%
95   }
96   \fi
97   \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.

```

98   \ifx\inserttitle\@empty\else{%
99     \raggedright%
100    \linespread{1.0}%
101    \usebeamerfont{title}%
102    \usebeamercolor[fg]{title}%
103    \@metropolis@titleformat{\inserttitle}%
104    \par%
105    \vspace*{0.5em}
106  }}
107  \fi

```

```

108 \ifx\insertsubtitle\@empty\else{%
109   \usebeamerfont{subtitle}%
110   \usebeamercolor[fg]{subtitle}%
111   \insertsubtitle%
112   \par%
113   \vspace*{0.5em}
114 }%
115 \fi

```

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

```

116 \begin{tikzpicture}
117   \usebeamercolor{title separator}
118   \draw[fg] (0, 0) -- (\textwidth, 0);
119 \end{tikzpicture}%
120 \par%
121 \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](#).

```

122 \ifx\beamer@shortauthor\@empty\else{%
123   \usebeamerfont{author}%
124   \usebeamercolor[fg]{author}%
125   \insertauthor%
126   \par%
127   \vspace*{0.25em}
128 }%
129 \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`.

```

130 \ifx\insertdate\@empty\else{%
131   \usebeamerfont{date}%
132   \usebeamercolor[fg]{date}%
133   \insertdate%

```

```

134     \par%
135   }}
136   \fi
137   \ifx\insertinstitute\@empty\else{%
138     \vspace*{3mm}
139     \usebeamerfont{institute}%
140     \usebeamercolor[fg]{institute}%
141     \insertinstitute%
142     \par%
143   }}
144   \fi
145   \vfill
146   \vspace*{1mm}
147 \end{minipage}
148 }

```

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`

```

149 \def\maketitle{%
150   \ifbeamer@inframe
151     \titlepage
152   \else
153     \frame[plain]{\titlepage}
154   \fi
155 }
156 \def\titlepage{%
157   \usebeamertemplate{title page}
158 }

```

9.3 Section page

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

```

159 \newcommand{\@metropolis@sectionpage@none}{
160   \AtBeginSection{
161     % intenionally empty
162   }
163 }
164 \defbeamertemplate{section page}{progressbar}{
165   \vspace{2em}
166   \centering
167   \begin{minipage}{22em}
168     \usebeamercolor[fg]{section title}
169     \usebeamerfont{section title}
170     \insertsectionhead\[-1ex]
171     \usebeamertemplate*{progress bar in section page}
172   \end{minipage}
173   \par
174 }
175 \newcommand{\@metropolis@sectionpage@progressbar}{
176   \setbeamertemplate{section page}[progressbar]
177   \AtBeginSection{
178     \ifbeamer@inframe
179       \sectionpage
180     \else
181       \frame[plain,c]{\sectionpage}
182     \fi
183   }
184 }

```

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`.

```

185 \newlength{\metropolis@progressonsectionpage}
186 \setbeamertemplate{progress bar in section page}{
187   \setlength{\metropolis@progressonsectionpage}{%
188     \textwidth * \ratio{\insertframenum pt}{\inserttotalframenum pt}%
189   }%
190   \begin{tikzpicture}
191     \draw[bg, fill=bg] (0,0) rectangle (\textwidth, 0.4pt);
192     \draw[fg, fill=fg] (0,0) rectangle (\metropolis@progressonsectionpage, 0.4pt);
193   \end{tikzpicture}%

```

```
194 }
```

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

```
195 \def\inserttotalframenumber{100}
```

9.4 Block environments

```
196 \newlength{\@metropolis@blockskip}
197 \setbeamertemplate{block begin}{%
198   \vspace*{1ex}
199   \begin{beamercolorbox}{%
200     ht=2.4ex,
201     dp=1ex,
202     leftskip=\@metropolis@blockskip,
203     rightskip=\@metropolis@blockskip}{block title}
204     \usebeamerfont*{block title}\insertblocktitle%
205   \end{beamercolorbox}%
206   \vspace*{-1pt}
207   \usebeamerfont{block body}%
208   \begin{beamercolorbox}{%
209     dp=1ex,
210     leftskip=\@metropolis@blockskip,
211     rightskip=\@metropolis@blockskip,
212     vmode]{block body}%
213 }
214 \setbeamertemplate{block end}{%
215   \end{beamercolorbox}
216   \vspace*{0.2ex}
217 }
```

Alerted block environment

```

218 \setbeamertemplate{block alerted begin}{%
219   \vspace*{1ex}
220   \begin{beamercolorbox}[%
221     ht=2.4ex,
222     dp=1ex,
223     leftskip=\@metropolis@blockskip,
224     rightskip=\@metropolis@blockskip]{block title alerted}
225     \usebeamerfont*{block title alerted}\insertblocktitle%
226   \end{beamercolorbox}%
227   \vspace*{-1pt}
228   \usebeamerfont{block body alerted}%
229   \begin{beamercolorbox}[%
230     dp=1ex,
231     leftskip=\@metropolis@blockskip,
232     rightskip=\@metropolis@blockskip,
233     vmode]{block body}%
234   }
235 \setbeamertemplate{block alerted end}{%
236   \end{beamercolorbox}
237   \vspace*{0.2ex}
238 }

```

Example block environment

```

239 \setbeamertemplate{block example begin}{%
240   \vspace*{1ex}
241   \begin{beamercolorbox}[%
242     ht=2.4ex,
243     dp=1ex,
244     leftskip=\@metropolis@blockskip,
245     rightskip=\@metropolis@blockskip]{block title example}
246     \usebeamerfont*{block title example}\insertblocktitle%
247   \end{beamercolorbox}%
248   \vspace*{-1pt}
249   \usebeamerfont{block body example}%
250   \begin{beamercolorbox}[%
251     dp=1ex,
252     leftskip=\@metropolis@blockskip,
253     rightskip=\@metropolis@blockskip,
254     vmode]{block body}%
255   }

```



```

256 \setbeamertemplate{block example end}{%
257   \end{beamercolorbox}
258   \vspace*{0.2ex}
259 }

```

9.5 Itemize/enumerate environments

```

260 \setlength{\leftmargini}{1em}
261 \setlength{\leftmarginii}{1em}
262 \setlength{\leftmarginiii}{1em}
263 \setbeamertemplate{itemize item}{\textbullet}
264 \setbeamertemplate{itemize subitem}{\textbullet}
265 \setbeamertemplate{itemize subsubitem}{\textbullet}

```

9.6 Figures and tables

```

266 \setbeamertemplate{caption label separator}{: }
267 \setbeamertemplate{caption}[numbered]

```

9.7 Footnotes

```

268 \setbeamertemplate{footnote}{%
269   \parindent 0em\noindent%
270   \raggedright
271   \usebeamercolor{footnote}\hbox to 0.8em{\hfil\insertfootnotemark}\insertfootnotetext
272 }

```

9.8 General text settings

```

273 \mode<all>
274 \setlength{\parskip}{0.5em}
275 \linespread{1.15}

```

Process package options

```

276 \@metropolis@inner@setdefaults
277 \ProcessPgfPackageOptions{/metropolis/inner}

```

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.

Load required packages.

```
278 \RequirePackage{calc}
279 \RequirePackage{pgfopts}
```

10.1 Options

numbering This option controls the page numbering.

```
280 \pgfkeys{
281   /metropolis/outer/numbering/.cd,
282   .is choice,
283   none/.code=\setbeamertemplate{frame numbering}[none],
284   counter/.code=\setbeamertemplate{frame numbering}[counter],
285   fraction/.code=\setbeamertemplate{frame numbering}[fraction],
286 }
```

progressbar This option controls the progressbar.

```
287 \pgfkeys{
288   /metropolis/outer/progressbar/.cd,
289   .is choice,
290   none/.code=\setbeamertemplate{progress bar in head/foot}[none],
291   head/.code=\setbeamertemplate{progress bar in head/foot}[head],
292 }
```

frametitleoffset This option controls the frame title offset.

```
293 \pgfkeys{
294   /metropolis/outer/.cd,
295   frametitleoffset/.code=\setlength{\@metropolis@voffset}{#1},
296   noframetitleoffset/.code=\setlength{\@metropolis@voffset}{0em},
297 }
```

Set default values for options.

```
298 \newcommand{\@metropolis@outer@setdefaults}{
299   \pgfkeys{/metropolis/outer/.cd,
300     numbering=counter,
301     progressbar=none,
302     frametitleoffset=2em,
```

```

303 }
304 }

```

10.2 Head and footline

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

```

305 \setbeamertemplate{navigation symbols}{}

```

Template for the frame number. Can be omitted, shown or displayed as a fraction of the total frames.

```

306 \defbeamertemplate{frame numbering}{none}{
307   % intentionally empty
308 }
309 \defbeamertemplate{frame numbering}{counter}{
310   \insertframenumbers
311 }
312 \defbeamertemplate{frame numbering}{fraction}{
313   \insertframenumbers/\inserttotalframenumbers
314 }

```

Define additional space between frame title and content. By default 2em.

```

315 \newlength{\@metropolis@voffset}

```

The only element in the footline by default is the frame number.

```

316 \setbeamertemplate{footline}{%
317   \begin{beamercolorbox}[%
318     wd=\textwidth,
319     ht=3ex,
320     dp=3ex,
321     leftskip=0.3cm,
322     rightskip=0.3cm
323   ]{footline}%
324   \hfill\usebeamerfont{page number in head/foot}%
325   \usebeamertemplate*{frame numbering}
326   \end{beamercolorbox}%
327 }

```

10.3 Frametitle

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

```
328 \setbeamertemplate{frametitle}{%
329   \nointerlineskip
330   \begin{beamercolorbox}[%
331     wd=\paperwidth,
332     leftskip=0.3cm,
333     rightskip=0.3cm,
334     ht=2.5ex,
335     dp=1.5ex
336   ]{frametitle}
337   \insertframetitle%
338   \end{beamercolorbox}%
339   \usebeamertemplate*{progress bar in head/foot}
340   \vspace{\@metropolis@voffset}
341 }
```

progress bar in head/foot 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**.

```
342 \newlength{\metropolis@progressinheadfoot}
343 \defbeamertemplate{progress bar in head/foot}{none}{}
344 \defbeamertemplate{progress bar in head/foot}{head}{
345   \nointerlineskip
346   \setlength{\metropolis@progressinheadfoot}{%
347     \paperwidth * \ratio{\insertframenumber pt}{\inserttotalframenumber pt}}%
348   }%
349   \begin{beamercolorbox}[
350     wd=\paperwidth,
351     ht=0.4pt,
352     dp=0pt]{progress bar in head/foot}
353     \begin{tikzpicture}
354       \draw[bg, fill=bg] (0,0) rectangle (\paperwidth, 0.4pt);
355       \draw[fg, fill=fg] (0,0) rectangle (\metropolis@progressinheadfoot, 0.4pt);
356     \end{tikzpicture}%
357   \end{beamercolorbox}
358 }
```

Process package options

```
359 \@metropolis@outer@setdefaults
360 \ProcessPgfPackageOptions{/metropolis/outer}
```

11 Implementation: Fira font theme

Font Definitions

```
361 \RequirePackage[no-math]{fontspec}
362 \defaultfontfeatures{Mapping=tex-text}
363 \setsansfont[BoldItalicFont={Fira Sans Italic},%
364             ItalicFont={Fira Sans Light Italic},%
365             BoldFont={Fira Sans}]{Fira Sans Light}
366 \setmonofont{Fira Mono}
367 \newfontfamily\ExtraLight{Fira Sans ExtraLight}
368 \newfontfamily\Light{Fira Sans Light}
369 \newfontfamily\Book{Fira Sans}
370 \newfontfamily\Medium{Fira Sans Medium}
371 \AtBeginEnvironment{tabular}{%
372     \setsansfont[BoldFont={Fira Sans},%
373                 Numbers={Monospaced}]{Fira Sans Light}%
374 }
```

Font Assignment

```
375 \setbeamerfont{title}{family=\Book, size=\Large, shape=\scshape}
376 \setbeamerfont{author}{family=\ExtraLight, size=\small}
377 \setbeamerfont{date}{family=\ExtraLight, size=\small}
378 \setbeamerfont{section title}{family=\Book, size=\Large, shape=\scshape}
379 \setbeamerfont{block title}{family=\Book, size=\normalsize}
380 \setbeamerfont{block title alerted}{family=\Book, size=\normalsize}
381 \setbeamerfont{subtitle}{family=\Light, size=\fontsize{12}{14}}
382 \setbeamerfont{frametitle}{family=\Book, size=\large, shape=\scshape}
383 \setbeamerfont{caption}{size=\small}
384 \setbeamerfont{caption name}{family=\Book}
385 \setbeamerfont{description item}{family=\Book}
386 \setbeamerfont{page number in head/foot}{size=\scriptsize}
```

Bibliography

```
387 \setbeamerfont{bibliography entry author}{family=\Light, size=\normalsize}
388 \setbeamerfont{bibliography entry title}{family=\Book, size=\normalsize}
389 \setbeamerfont{bibliography entry location}{family=\Light, size=\normalsize}
390 \setbeamerfont{bibliography entry note}{family=\Light, size=\small}
391 \linespread{1.15}
```

12 Implementation: METROPOLIS color theme

Load required packages.

```
392 \RequirePackage{pgfopts}
```

12.1 Options

block This option controls whether the blocks are filled or transparent.

```
393 \pgfkeys{
394   /metropolis/color/block/.cd,
395   .is choice,
396   transparent/.code=\@metropolis@block@transparent,
397   fill/.code=\@metropolis@block@fill,
398 }
```

colors Defines whether the background shall be dark and the foreground be light or vice versa

```
399 \pgfkeys{
400   /metropolis/color/background/.cd,
401   .is choice,
402   dark/.code=\@metropolis@colors@dark,
403   light/.code=\@metropolis@colors@light,
404 }
```

Set default values for options.

```
405 \newcommand{\@metropolis@color@setdefaults}{
406   \pgfkeys{/metropolis/color/.cd,
```

```

407     background=light,
408     block=fill,
409 }
410 }

```

12.2 Base colors

```

411 \definecolor{mDarkBrown}{HTML}{604c38}
412 \definecolor{mDarkTeal}{HTML}{23373b}
413 \definecolor{mLightBrown}{HTML}{EB811B}
414 \definecolor{mLightGreen}{HTML}{14B03D}

```

12.3 Base styles

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

```

415 \newcommand{\@metropolis@colors@dark}{
416   \setbeamercolor{normal text}{%
417     fg=black!2,
418     bg=mDarkTeal
419   }
420 }
421 \newcommand{\@metropolis@colors@light}{
422   \setbeamercolor{normal text}{%
423     fg=mDarkTeal,
424     bg=black!2
425   }
426 }
427 \setbeamercolor{alerted text}{%
428   fg=mLightBrown
429 }
430 \setbeamercolor{example text}{%
431   fg=mLightGreen
432 }

```

12.4 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`.

```
433 \setbeamercolor{titlelike}{%  
434   use=normal text,  
435   parent=normal text  
436 }  
437 \setbeamercolor{structure}{%  
438   fg=normal text.fg  
439 }
```

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.

```
440 \setbeamercolor{palette primary}{%  
441   use=normal text,  
442   fg=normal text.bg,  
443   bg=normal text.fg  
444 }  
445 \setbeamercolor{frametitle}{%  
446   use=palette primary,  
447   parent=palette primary  
448 }
```

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`.

```
449 \setbeamercolor{progress bar}{%  
450   use=alerted text,  
451   fg=alerted text.fg,  
452   bg=normal text.bg!50!normal text.fg  
453 }  
454 \setbeamercolor{title separator}{  
455   use=progress bar,
```



```

456   parent=progress bar
457 }
458 \setbeamercolor{progress bar in head/foot}{%
459   use=progress bar,
460   parent=progress bar
461 }
462 \setbeamercolor{progress bar in section page}{
463   use=progress bar,
464   parent=progress bar
465 }

```

Blocks

```

466 \newcommand{\@metropolis@block@transparent}{
467   \setbeamercolor{block title}{use=normal text, parent=normal text}
468 }
469 \newcommand{\@metropolis@block@fill}{
470   \setbeamercolor{block title}{%
471     use=normal text,
472     fg=normal text.fg,
473     bg=normal text.bg!80!fg
474   }
475 }
476 \setbeamercolor{block title alerted}{%
477   use={block title, alerted text},
478   bg=block title.bg,
479   fg=alerted text.fg
480 }
481 \setbeamercolor{block title example}{%
482   use={block title, example text},
483   bg=block title.bg,
484   fg=example text.fg
485 }
486 \setbeamercolor{block body alerted}{use=block body, parent=block body}
487 \setbeamercolor{block body example}{use=block body, parent=block body}
488 \setbeamercolor{block body}{
489   use={block title, normal text},
490   bg=block title.bg!50!normal text.bg
491 }

```

Footnotes

```
492 \setbeamercolor{footnote}{fg=normal text.fg!90}  
493 \setbeamercolor{footnote mark}{fg=.
```

Process package options

```
494 \@metropolis@color@setdefaults  
495 \ProcessPgfPackageOptions{/metropolis/color}  
  
496 \mode<all>
```

13 Implementation: Tol pgfplots theme

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

```
497 \definecolor{TolDarkPurple}{HTML}{332288}  
498 \definecolor{TolDarkBlue}{HTML}{6699CC}  
499 \definecolor{TolLightBlue}{HTML}{88CCEE}  
500 \definecolor{TolLightGreen}{HTML}{44AA99}  
501 \definecolor{TolDarkGreen}{HTML}{117733}  
502 \definecolor{TolDarkBrown}{HTML}{999933}  
503 \definecolor{TolLightBrown}{HTML}{DDCC77}  
504 \definecolor{TolDarkRed}{HTML}{661100}  
505 \definecolor{TolLightRed}{HTML}{CC6677}  
506 \definecolor{TolLightPink}{HTML}{AA4466}  
507 \definecolor{TolDarkPink}{HTML}{882255}  
508 \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.

```
509 \pgfplotscreateplotcyclelist{mbarplot cycle}{%  
510 {draw=TolDarkBlue, fill=TolDarkBlue!70},  
511 {draw=TolLightBrown, fill=TolLightBrown!70},  
512 {draw=TolLightGreen, fill=TolLightGreen!70},
```

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

```

513 {draw=TolDarkPink,    fill=TolDarkPink!70},
514 {draw=TolDarkPurple,  fill=TolDarkPurple!70},
515 {draw=TolDarkRed,     fill=TolDarkRed!70},
516 {draw=TolDarkBrown,   fill=TolDarkBrown!70},
517 {draw=TolLightRed,    fill=TolLightRed!70},
518 {draw=TolLightPink,   fill=TolLightPink!70},
519 {draw=TolLightPurple, fill=TolLightPurple!70},
520 {draw=TolLightBlue,   fill=TolLightBlue!70},
521 {draw=TolDarkGreen,   fill=TolDarkGreen!70},
522 }

```

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

```

523 \pgfplotscreateplotcyclelist{mlineplot cycle}{%
524   {TolDarkBlue, mark=*, mark size=1.5pt},
525   {TolLightBrown, mark=square*, mark size=1.3pt},
526   {TolLightGreen, mark=triangle*, mark size=1.5pt},
527   {TolDarkBrown, mark=diamond*, mark size=1.5pt},
528 }

```

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.

```

529 \pgfplotsset{
530   compat=1.9,

```

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

```

531 mlineplot/.style={
532   mbaseplot,
533   xmajorgrids=true,
534   ymajorgrids=true,
535   major grid style={dotted},
536   axis x line=bottom,
537   axis y line=left,
538   legend style={
539     cells={anchor=west},
540     draw=none
541   },

```

```

542     cycle list name=mlineplot cycle,
543 },

```

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**.

```

544 mbarplot base/.style={
545     mbaseplot,
546     bar width=6pt,
547     axis y line*=none,
548 },
549 mbarplot/.style={
550     mbarplot base,
551     ybar,
552     xmajorgrids=false,
553     ymajorgrids=true,
554     area legend,
555     legend image code/.code={%
556         \draw[#1] (0cm,-0.1cm) rectangle (0.15cm,0.1cm);
557     },
558     cycle list name=mbarplot cycle,
559 },
560 horizontal mbarplot/.style={
561     mbarplot base,
562     xmajorgrids=true,
563     ymajorgrids=false,
564     xbar stacked,
565     area legend,
566     legend image code/.code={%
567         \draw[#1] (0cm,-0.1cm) rectangle (0.15cm,0.1cm);
568     },
569     cycle list name=mbarplot cycle,
570 },

```

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

```

571 mbaseplot/.style={
572     legend style={
573         draw=none,
574         fill=none,

```

```

575     cells={anchor=west},
576 },
577 x tick label style={
578     font=\footnotesize
579 },
580 y tick label style={
581     font=\footnotesize
582 },
583 legend style={
584     font=\footnotesize
585 },
586 major grid style={
587     dotted,
588 },
589 axis x line*=bottom,
590 },
591 disable thousands separator/.style={
592     /pgf/number format/.cd,
593     1000 sep={}
594 },
595 }

```