

# Modern Beamer Presentations with the **NEO** package

v1.0 — 2017/10/01

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Getting Started</b>	<b>4</b>
2.1	Installing from GitLab . . . . .	4
2.2	A Minimal Example . . . . .	4
2.3	Dependencies . . . . .	5
2.4	Pandoc . . . . .	5
<b>3</b>	<b>Customization</b>	<b>6</b>
3.1	Package options . . . . .	6
3.1.1	Main theme . . . . .	6
3.1.2	Inner theme . . . . .	6
3.1.3	Outer theme . . . . .	7
3.1.4	Color theme . . . . .	7
3.1.5	Font theme . . . . .	7
3.2	Color Customization . . . . .	8
3.3	Font Customization . . . . .	8
3.3.1	Old style figures . . . . .	9
3.4	Commands . . . . .	9
3.4.1	Standout frames . . . . .	9
<b>4</b>	<b>pgfplots integration</b>	<b>9</b>
4.1	Styles . . . . .	9

4.2	Paul Tol colors	10
<b>5</b>	<b>Tips &amp; Tricks</b>	<b>10</b>
5.1	Backup Slides	10
<b>6</b>	<b>Known Issues</b>	<b>10</b>
6.1	Title formats	10
6.2	Interactions with other color themes	11
6.3	Notes on second screen	12
6.4	Standout frames with labels	12
6.5	Standout frames with Pandoc	13
<b>7</b>	<b>License</b>	<b>13</b>
<b>8</b>	<b>Implementation</b>	<b>13</b>
8.1	NEO parent theme	13
8.1.1	Package dependencies	14
8.1.2	Options	14
8.1.3	Component sub-packages	16
8.1.4	Custom commands	16
8.1.5	Process package options	17
8.2	NEO inner theme	17
8.2.1	Package dependencies	17
8.2.2	Options	18
8.2.3	Title page	18
8.2.4	Section page	22
8.2.5	Block environments	24
8.2.6	Lists and floats	27
8.2.7	Footnotes	27
8.2.8	Text and spacing settings	27
8.2.9	Standout frames	28
8.2.10	Process package options	29
8.3	NEO outer theme	29
8.3.1	Package dependencies	29
8.3.2	Options	30
8.3.3	Head and footline	32
8.3.4	Frametitle	34
8.3.5	Process package options	38

8.4	<b>NEO</b> font theme . . . . .	38
8.4.1	Package dependencies . . . . .	38
8.4.2	Load Fira fonts . . . . .	38
8.4.3	General font definitions . . . . .	41
8.4.4	Font style options . . . . .	41
8.4.5	Title format options . . . . .	42
8.4.6	Process package options . . . . .	48
8.5	<b>NEO</b> color theme . . . . .	48
8.5.1	Package dependencies . . . . .	48
8.5.2	Options . . . . .	48
8.5.3	Base colors . . . . .	49
8.5.4	Alias colors . . . . .	49
8.5.5	Base styles . . . . .	50
8.5.6	Derived colors . . . . .	51
8.5.7	Process package options . . . . .	54
8.6	Tol pgfplots theme . . . . .	54

## 1 Introduction

Beamer is an awesome way to make presentations with LaTeX, but its theme selection is surprisingly sparse. The stock themes share an aesthetic that can be a little cluttered, while the few distinctive custom themes available are often specialized for a particular corporate or institutional brand.

The goal of **NEO** is to provide a simple, modern Beamer theme suitable for anyone to use – it is based on the **metropolis** theme by **Matthias Vogelgesang** and **many other of contributors**. It tries to minimize noise and maximize space for content; the only visual flourish it offers is an (optional) progress bar added to each slide or to the section slides.

By default, **NEO** uses **Fira Sans**, a gorgeous typeface commissioned by Mozilla and designed by **Carrois**. For best results, you should use the Fira typeface distributed by this package and use  $\text{Xe}_{\text{L}}\text{TeX}$  to typeset your slides. However, **NEO** can also be used with other typefaces and  $\text{E}_{\text{T}}\text{X}$  build systems.

## 2 Getting Started

### 2.1 Installing from GitLab

If you want to use the cutting-edge development version of **NEO**, you can install it manually. Like any  $\text{\LaTeX}$  package, this involves four easy steps:

**Download the source** with a `git clone` of the **NEO repository** or as a **zip archive** of the latest development version.

**Compile the style files** by running `make sty` inside the downloaded directory. (Or run  $\text{\LaTeX}$  directly on `source/neotheme.ins`.)

**Move the resulting `*.sty` files** to the folder containing your presentation. To use **NEO** with many presentations, run `make install` or move the `*.sty` files to a folder in your  $\text{\TeX}$  path instead.

**Use the theme for your presentation** by declaring `\usetheme{neo}` in the preamble of your Beamer document.

**NEO** uses the Make build system to offer the following installation options for advanced users:

`make sty` builds the theme style files.

`make doc` builds this documentation manual.

`make demo` builds a demo presentation to test the features of **NEO**.

`make all` builds the theme and manual.

`make clean` removes the files generated by `make all`.

`make install` installs the theme into your local `texmf` folder.

`make uninstall` removes the theme from your local `texmf` folder.

### 2.2 A Minimal Example

The following code shows a minimal example of a Beamer presentation using **NEO**.

```
\documentclass{beamer}
\usetheme{neo}          % Use neo theme
\title{A minimal example}
```

```

\date{\today}
\author{Matthias Vogelgesang}
\institute{Centre for Modern Beamer Themes}
\begin{document}
  \maketitle
  \section{First Section}
  \begin{frame}{First Frame}
    Hello, world!
  \end{frame}
\end{document}

```

## 2.3 Dependencies

**NEO** depends on the beamer class and the following standard packages:

- tikz
- etoolbox
- ifxetex
- pgfopts
- calc
- ifluatex

For best results, we recommend installing the fonts **Fira Sans** and **Fira Mono** and compiling with **NEO** using Xe<sub>La</sub>TeX or Lua<sub>La</sub>TeX. These are optional dependencies; **NEO** is compatible with (e.g.) pdf<sub>La</sub>TeX and will fall back to standard fonts if **Fira Sans** or **Fira Mono** is not installed.

The packaged name of **Fira Sans** is **Fira Sans OT** in some Linux distributions; this case is automatically handled by **NEO**.

## 2.4 Pandoc

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

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

## 3 Customization

### 3.1 Package options

The theme provides a number of options, which can be set using a key=value interface. The primary way to set options is to provide a comma-separated list of option-value pairs when loading **NEO** in the preamble:

```
\usetheme[option1=value1, option2=value2, ...]{neo}
```

Options can be changed at any time — even mid-presentation! — with the `\neoset` macro.

```
\neoset{option1=newvalue1, option2=newvalue2, ...}
```

The list of options is structured as shown in the following example.

option key *list of possible values* ..... default

A short description of the option.

#### 3.1.1 Main theme

titleformat *regular, smallcaps, allsmallcaps, allcaps* ..... regular

Changes the format of titles, subtitles, section titles, frame titles, and the text on “standout” frames. The available options produce Regular, SMALLCAPS, ALLSMALLCAPS, or ALLCAPS titles. Please refer to Section 6.1 for known issues with these options.

titleformat plain *regular, smallcaps, allsmallcaps, allcaps* ..... regular

Changes the format of “standout” frames (see titleformat, above).

#### 3.1.2 Inner theme

sectionpage *none, simple, progressbar* ..... progressbar

Adds a slide at the start of each section (simple) with an optional thin progress bar below the section title (progressbar). The none option disables the section page.

subsectionpage *none, simple, progressbar* ..... none

Optionally adds a slide at the start of each subsection. If enabled with the simple or progressbar options, the style of the section page will be updated to match the style of the subsection page. Note that section slides and subsection slides can appear consecutively if both are enabled; you may want to use this option together with sectionpage=none depending on the section structure of your presentation.

### 3.1.3 Outer theme

numbering *none, counter, fraction* ..... counter

Controls whether the frame number at the bottom right of each slide is omitted (none), shown (counter) or displayed as a fraction of the total number of frames (fraction).

progressbar *none, head, frametitle, foot* ..... none

Optionally adds a progress bar to the top of each frame (head), the bottom of each frame (foot), or directly below each frame title (frametitle).

### 3.1.4 Color theme

block *transparent, fill* ..... transparent

Optionally adds a light grey background to block environments like theorem and example.

background *dark, light, white* ..... light

Provides the option to have a dark background and light foreground instead of the reverse. White is similar to light but uses a pure white background.

### 3.1.5 Font theme

titleformat title *regular, smallcaps, allsmallcaps, allcaps* ..... regular

titleformat subtitle  
titleformat section  
titleformat frame

Individually controls the format of titles, subtitles, section titles, and frame titles (see titleformat, above).

## 3.2 Color Customization

The included **NEO** 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 stock beamer colors. In addition to the stock colors the theme defines a number of **NEO** specific colors, which can also be redefined to your liking.

```
\setbeamercolor{progress bar}{ ... }  
\setbeamercolor{title separator}{ ... }  
\setbeamercolor{progress bar in head/foot}{ ... }  
\setbeamercolor{progress bar in section page}{ ... }
```

## 3.3 Font Customization

The default font for **NEO** is Fira. This can be easily changed using the standard font selection commands of the `fontspec` package. So if you prefer, for example, the **Ubuntu** font family, just add the following two commands after loading the **NEO** theme.

```
\setsansfont{Ubuntu}  
\setmonofont{Ubuntu Mono}
```

If you are expecting to present in a large room or with an underpowered projector, you may want to change the font to a heavier weight of Fira to maximize readability.

```
\setsansfont[BoldFont={Fira Sans SemiBold}]{Fira Sans Book}
```



### 3.3.1 Old style figures

The regular fontspec mechanism for changing glyph appearance applies also to this theme. If you want to have old style figures in the text but regular lined figures for math, you could add the following to your preamble:

```
\usefonttheme{professionalfonts}    % required for mathspec
\usepackage{mathspec}
\setsansfont[BoldFont={Fira Sans},
             Numbers={OldStyle}]{Fira Sans Light}
\setmathsfon(Digits)[Numbers={Lining, Proportional}]{Fira
  Sans Light}
```

## 3.4 Commands

### 3.4.1 Standout frames

The **NEO** inner theme offers a custom frame format with large, centered text and an inverted background — perfect for focusing attention on single sentence or image. To use it, add the key `standout` to the frame:

```
\begin{frame}[standout]
  Thank you!
\end{frame}
```

## 4 pgfplots integration

**NEO** comes with a set of pre-defined pgfplots styles and a color theme based on Paul Tol's color scheme.

### 4.1 Styles

Pass the following style keys to the axis environment to get the appropriate effect:

<code>mlineplot</code>	Plot regular line charts with reduced axis frames, less intrusive legend and subdued grid.
------------------------	--

<code>mbarplot</code>	Plot vertical bar charts in a similar way as <code>mlineplot</code> but reduce grid usage.
<code>horizontal mbarplot</code>	Plot horizontal bar charts.
<code>disable thousands separator</code>	Helper style to remove thousands separator.

## 4.2 Paul Tol colors

A good presentation uses colors that are distinct from each other as much as possible as well as from black and white, can be discerned item under different lighting and display environments and by color-blind viewers, while matching well together.

In a [technical note](#) for SRON, Paul Tol proposed a palette of colors satisfying these constraints. The sub-package `pgfplots-themetol` defines palettes for `pgfplots` charts based on Tol's work.

# 5 Tips & Tricks

## 5.1 Backup Slides

Speakers will often include extra slides at the end of their presentation to refer to during audience questions. One easy way to do this is to include the `appendixnumberbeamer` package in your preamble and call `\appendix` before your backup slides.

**NEO** will automatically turn off slide numbering and progress bars for slides in the appendix.

# 6 Known Issues

## 6.1 Title formats

Be aware that not every font supports small caps, so the `smallcaps` or `allsmallcaps` options may not work if you use a font other than Fira Sans. In particular, the Computer Modern sans-serif typeface, which is used when **NEO** is compiled with pdf<sub>La</sub>TeX, does not have a small-caps variant.

The title format options `allsmallcaps` and `allcaps` are quite nice from an aesthetic point of view, but their use of `\MakeLowercase` and `\MakeUppercase` can cause unexpected problems. For example:

- Some commands, like `\`, do not work inside `\MakeLowercase` and `\MakeUppercase`. (See [#125](#))
- Only alphabetic characters are affected by `\MakeLowercase`, so numerals and punctuation remain at full height. This can spoil some of the aesthetic benefits of `allsmallcaps`. (See [#33](#))
- `\MakeLowercase` and `\MakeUppercase` apply to math mode and `\scshape` does not. This can easily introduce mathematical errors that are hard to catch.
- It is impossible to typeset symbols which are encoded as uppercase letters in a different font. In particular, `\mathbb` and `\mathcal` letters will be replaced by other math glyphs. (See [#153](#))

The `allsmallcaps` and `allcaps` options are safe to use if your titles contain only alphabetic characters and do not require the expansion of any macros.

## 6.2 Interactions with other color themes

**NEO** can be used along with any other Beamer color theme, such as `crane` or `seahorse`. If you wish to do this, it is usually best to include the **NEO** sub-packages individually so the **NEO** color theme is never loaded. This will prevent conflicts between the **NEO** color theme and your preferred theme.

For example, overriding the color theme as follows may not work as expected because `\usetheme{neo}` loads the **NEO** color theme, which defines a relationship between the `frametitle` background and the primary palette of the theme. Since `seahorse` assumes a different relationship between its palettes, the result is a grey, rather than `periwinkle`, `frametitle` background.

```
\usetheme{neo}
\usecolortheme{seahorse}
```

The correct colors are chosen if the **NEO** outer, inner, and font themes are loaded separately:

```
\useoutertheme{neo}
```

```

\useinnertheme{neo}
\usefonttheme{neo}
\usecolortheme{seahorse}    % or your preferred color theme

```

Please note that **NEO** may not use all the colors defined in your favourite Beamer color theme. In particular, **NEO** does not set a background color for the title; this will cause issues when using color themes like `whale` which set a white foreground for the title.

### 6.3 Notes on second screen

If you use the `[show notes on second screen]` option built in to Beamer and compile with  $\text{\LaTeX}$ , text on slides following the first section slide may be rendered in white instead of the regular colour. This is due to a [bug](#) in Beamer or  $\text{\LaTeX}$  itself. You can work around it either by compiling with  $\text{\LuaTeX}$  or by adding the following code to your preamble to reset the text color on each slide.

```

\makeatletter
\def\beamer@frametnotesbegin{% at beginning of slide
    \usebeamercolor[fg]{normal text}
    \gdef\beamer@noteitems{}%
    \gdef\beamer@notes{}%
}
\makeatother

```

### 6.4 Standout frames with labels

Because the `standout` frame option creates a group to restrict the colour change to a single slide, labels defined after calling `standout` will stay local to the group. In other words, the following may result in a “label undefined” error.

```

\begin{frame}[standout, label=conclusion]{Conclusion}
    Awesome slide
\end{frame}

```

To fix this problem, change the order of the keys in the frame.

```
\begin{frame}[label=conclusion, standout]{Conclusion}
  Awesome slide
\end{frame}
```

This error can be unwittingly triggered if you export your slides from Emacs Org mode, which automatically adds labels after frame options. Alex Branham [offers](#) the following solution for Org mode users, using `org-set-property`.

```
* Start of a frame
:PROPERTIES:
:BEAMER_opt: label=conclusion,standout
:END:
```

## 6.5 Standout frames with Pandoc

With Pandoc versions prior 1.17.2 it was not possible to create standout frames because Pandoc only supported a specific list of frame attributes thus ignoring additional attributes such as `{.standout}`.

## 7 License

**NEO** 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 any presentations that you create with the theme.

## 8 Implementation

### 8.1 **NEO** parent theme

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

### 8.1.1 Package dependencies

```
1 \RequirePackage{etoolbox}
2 \RequirePackage{pgfpages}
3 \RequirePackage{pgfopts}
```

### 8.1.2 Options

Most options are passed off to the component sub-packages.

```
4 \pgfkeys{/neo/.cd,
5   .search also={
6     /neo/inner,
7     /neo/outer,
8     /neo/color,
9     /neo/font,
10  }
11 }
```

`titleformat plain` Controls the formatting of the text on standout “plain” frames.

```
12 \pgfkeys{
13   /neo/titleformat plain/.cd,
14   .is choice,
15   regular/.code={%
16     \let\neo@plaintitleformat\@empty%
17     \setbeamerfont{standout}{shape=\normalfont}%
18   },
19   smallcaps/.code={%
20     \let\neo@plaintitleformat\@empty%
21     \setbeamerfont{standout}{shape=\scshape}%
22   },
23   allsmallcaps/.code={%
24     \let\neo@plaintitleformat\MakeLowercase%
25     \setbeamerfont{standout}{shape=\scshape}%
26     \PackageWarning{beamerthemeneo}{%
27       Be aware that titleformat plain=allsmallcaps can lead to problems%
28     }
29   },
30   allcaps/.code={%
31     \let\neo@plaintitleformat\MakeUppercase%
32     \setbeamerfont{standout}{shape=\normalfont}%

```

```

33     \PackageWarning{beamerthemeneo}{%
34         Be aware that titleformat plain=allcaps can lead to problems%
35     }
36 },
37 }

```

`titleformat` Sets a standard format for titles, subtitles, section titles, frame titles, and the text on standout “plain” frames.

```

38 \pgfkeys{
39   /neo/titleformat/.code=\pgfkeysalso{
40     font/titleformat title=#1,
41     font/titleformat subtitle=#1,
42     font/titleformat section=#1,
43     font/titleformat frame=#1,
44     titleformat plain=#1,
45   }
46 }

```

Shortcut option names as aliases to the corresponding key=value options.

```

47 \pgfkeys{/neo/.cd,
48   noslidenumbers/.code=\pgfkeysalso{outer/numbering=none},
49   usetotalslideindicator/.code=\pgfkeysalso{outer/numbering=fraction},
50   nosectionslide/.code=\pgfkeysalso{inner/sectionpage=none},
51   darkcolors/.code=\pgfkeysalso{color/background=dark},
52   whitebg/.code=\pgfkeysalso{color/background=white},
53   blockbg/.code=\pgfkeysalso{color/block=fill, inner/block=fill},
54   light/.code=\pgfkeysalso{font/style=light},
55   book/.code=\pgfkeysalso{font/style=book},
56   regular/.code=\pgfkeysalso{font/style=regular},
57 }

```

Set default values for options.

```

58 \newcommand{\neo@setdefaults}{
59   \pgfkeys{/neo/.cd,
60     titleformat plain=regular,
61   }
62 }

```

To avoid generating externalized figures of the progressbar we have to disable them with “tikzexternalenable” and “tikzexternaldisable”. However, if the “external” library is not loaded we would get undefined control sequence problems, hence we define them as no-ops if they are not defined yet.

```
63 \providecommand{\tikzexternalenable}{}
64 \providecommand{\tikzexternaldisable}{}

```

### 8.1.3 Component sub-packages

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

```
65 \useinnertheme{neo}
66 \useoutertheme{neo}
67 \usecolortheme{neo}
68 \usefonttheme{neo}

```

The tol theme for pgfplots is only loaded if pgfplots is used, pdfpcnotes always

```
69 \AtEndPreamble{%
70   \RequirePackage{pdfpcnotes}
71   \@ifpackageloaded{pgfplots}{%
72     \RequirePackage{pgfplotsthemetol}
73   }{}
74 }

```

### 8.1.4 Custom commands

The parent theme defines custom commands as their proper usage may depend on multiple sub-packages.

`\neoset` Allows the user to change options midway through a presentation.

```
75 \newcommand{\neoset}[1]{\pgfkeys{/neo/.cd,#1}}

```

`\plain` Creates a plain frame with dark background, suitable for displaying images or a few words. The format of the text can be set with the `titleformat plain` option.



```

76 \def\neo@plaintitleformat#1{#1}
77 \newcommand{\plain}[2][\]{%
78   \PackageWarning{beamerthemeneo}{%
79     The syntax ‘\plain’ may be deprecated in a future version of neo.
80     Please use a frame with [standout] instead.
81   }
82   \begin{frame}[standout]{#1}
83     \neo@plaintitleformat{#2}
84   \end{frame}
85 }

```

`\mreducelistspacing`

```

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

```

### 8.1.5 Process package options

```

87 \neo@setdefaults
88 \ProcessPgfOptions{/neo}

```

## 8.2 NEO 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.

### 8.2.1 Package dependencies

```

89 \RequirePackage{etoolbox}
90 \RequirePackage{keyval}
91 \RequirePackage{calc}
92 \RequirePackage{pgfopts}
93 \RequirePackage{pgfpages}
94 \RequirePackage{tikz}

```

## 8.2.2 Options

`sectionpage` Optionally add a slide marking the beginning of each section.

```
95 \pgfkeys{
96   /neo/inner/sectionpage/.cd,
97   .is choice,
98   none/.code=\neo@disablesectionpage,
99   simple/.code={\neo@enablesectionpage
100                  \setbeamertemplate{section page}[simple]},
101   progressbar/.code={\neo@enablesectionpage
102                      \setbeamertemplate{section page}[progressbar]},
103 }
```

`subsectionpage` Optionally add a slide marking the beginning of each subsection.

```
104 \pgfkeys{
105   /neo/inner/subsectionpage/.cd,
106   .is choice,
107   none/.code=\neo@disablesubsectionpage,
108   simple/.code={\neo@enablesubsectionpage
109                  \setbeamertemplate{section page}[simple]},
110   progressbar/.code={\neo@enablesubsectionpage
111                      \setbeamertemplate{section page}[progressbar]},
112 }
```

`\neo@inner@setdefaults` Set default values for inner theme options.

```
113 \newcommand{\neo@inner@setdefaults}{
114   \pgfkeys{/neo/inner/.cd,
115     sectionpage=progressbar,
116     subsectionpage=none
117   }
118 }
```

## 8.2.3 Title page

`title page` Template for the title page. Each element is only typset if it is defined by the user. If `\subtitle` is empty, for example, it won't leave a blank space on the title slide.

```

119 \setbeamertemplate{title page}{
120   \begin{minipage}[b][0.95\paperheight]{\textwidth}
121     \vfill%
122     \ifx\inserttitle\@empty\else\usebeamertemplate*{title}\fi
123     \ifx\insertsubtitle\@empty\else\usebeamertemplate*{subtitle}\fi
124     \usebeamertemplate*{title separator}

```

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](#).

```

125   \ifx\insertdate\@empty\else\usebeamertemplate*{date}\fi
126   \ifx\beamer@shortauthor\@empty\else\usebeamertemplate*{author}\fi
127   \ifx\insertinstitute\@empty\else\usebeamertemplate*{institute}\fi
128   \vfill
129   \begin{minipage}[b][0.25\paperheight][t]{\textwidth}
130 % The lower part of the title page background contains a white area which
131 % covers this whole minipage. Thus switch the text color back to normal
132     \neo@colors@light%
133     \usebeamercolor[fg]{normal text}%
134     \ifx\inserttitlegraphic\@empty\else\usebeamertemplate*{title graphic}\fi
135   \end{minipage}
136 \end{minipage}
137 }

```

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. In addition, `\maketitle` used outside of a frame will load a predefined background image, which can be changed using the optional argument: `\maketitle{extern}` for `images/titlepage-extern.pdf`

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

```

138 \renewcommand{\maketitle}[1][extern]{%
139   \ifbeamer@inframe
140     \titlepage

```

```

141 \else
142 {
143   \usebackgroundtemplate{
144     \tikzexternaldisable%
145     \begin{tikzpicture}
146       \node[anchor=north west,inner sep=0,outer sep=0] at (0, \paperheight) {\i
147       \fill[nWhite] (0,0) rectangle (\paperwidth, 0.3\paperheight);
148     \end{tikzpicture}%
149     \tikzexternalenable%
150   }
151   \frame[plain,noframenumbering]{
152     \neo@colors@dark
153     \setbeamercolor{title separator}{
154       fg=black!20,
155       bg=normal text.fg
156     }
157     \titlepage
158   }
159 }
160 \fi
161 }
162 \def\titlepage{%
163   \usebeamertemplate{title page}
164 }

```

**title graphic** Set the title graphic in a zero-height box, so it doesn't change the position of other elements.

```

165 \setbeamertemplate{title graphic}{
166   \vbox to 0pt {
167     \vspace*{2em}
168     \inserttitlegraphic%
169   }%
170   \nointerlineskip%
171 }

```

**title** Set the title on the title page.

```

172 \setbeamertemplate{title}{
173   \raggedright%
174   \linespread{1.0}%

```

```

175 \inserttitle%
176 \par%
177 \vspace*{0.5em}
178 }

```

**subtitle** Set the subtitle on the title page.

```

179 \setbeamertemplate{subtitle}{
180 \raggedright%
181 \insertsubtitle%
182 \par%
183 \vspace*{0.5em}
184 }

```

**title separator** Template to set the title graphic in a zero-height box. (It won't change the position of other elements.)

```

185 \newlength{\neo@titleseparator@linewidth}
186 \setlength{\neo@titleseparator@linewidth}{0.4pt}
187 \setbeamertemplate{title separator}{
188 \tikzexternaldisable%
189 \begin{tikzpicture}
190 \fill[fg] (0,0) rectangle (\textwidth, \neo@titleseparator@linewidth);
191 \end{tikzpicture}%
192 \tikzexternalenable%
193 \par%
194 }

```

**author** Set the author on the title page.

```

195 \setbeamertemplate{author}{
196 \vspace*{2em}
197 \insertauthor%
198 \par%
199 \vspace*{0.25em}
200 }

```

**date** Set the date on the title page.

```

201 \setbeamertemplate{date}{
202 \insertdate%

```

```

203 \par%
204 }

```

**institute** Set the institute on the title page.

```

205 \setbeamertemplate{institute}{
206 \vspace*{3mm}
207 \insertinstitute%
208 \par%
209 }

```

#### 8.2.4 Section page

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

```

210 \defbeamertemplate{section page}{simple}{
211 \begin{center}
212 \usebeamercolor[fg]{section title}
213 \usebeamerfont{section title}
214 \insertsectionhead\par
215 \ifx\insertsubsectionhead\@empty\else
216 \usebeamercolor[fg]{subsection title}
217 \usebeamerfont{subsection title}
218 \insertsubsectionhead
219 \fi
220 \end{center}
221 }
222 \defbeamertemplate{section page}{progressbar}{
223 \centering
224 \begin{minipage}{22em}
225 \raggedright
226 \usebeamercolor[fg]{section title}
227 \usebeamerfont{section title}
228 \insertsectionhead\[-1ex]
229 \usebeamertemplate*{progress bar in section page}
230 \par
231 \ifx\insertsubsectionhead\@empty\else%
232 \usebeamercolor[fg]{subsection title}%
233 \usebeamerfont{subsection title}%
234 \insertsubsectionhead
235 \fi

```

```

236 \end{minipage}
237 \par
238 \vspace{\baselineskip}
239 }
240 \newcommand{\neo@disablesectionpage}{
241   \AtBeginSection{
242     % intentionally empty
243   }
244 }
245 \newcommand{\neo@enablesectionpage}{
246   \AtBeginSection{
247     \ifbeamer@inframe
248       \sectionpage
249     \else
250       \frame[plain,c,noframenumbering]{\sectionpage}
251     \fi
252   }
253 }

```

subsection page    Template for the subsection title slide that can optionally be added to at the beginning of each subsection.

```

254 \setbeamertemplate{subsection page}{%
255   \usebeamertemplate*{section page}
256 }
257 \newcommand{\neo@disablesubsectionpage}{
258   \AtBeginSubsection{
259     % intentionally empty
260   }
261 }
262 \newcommand{\neo@enablesubsectionpage}{
263   \AtBeginSubsection{
264     \ifbeamer@inframe
265       \subsectionpage
266     \else
267       \frame[plain,c,noframenumbering]{\subsectionpage}
268     \fi
269   }
270 }

```

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.

```

271 \newlength{\neo@progressonsectionpage}
272 \newlength{\neo@progressonsectionpage@linewidth}
273 \setlength{\neo@progressonsectionpage@linewidth}{0.4pt}
274 \setbeamertemplate{progress bar in section page}{
275   \setlength{\neo@progressonsectionpage}{%
276     \textwidth * \ratio{\insertframenumbers pt}{\inserttotalframenumbers pt}}%
277   }%
278   \tikzexternaldisable%
279   \begin{tikzpicture}
280     \fill[bg] (0,0) rectangle (\textwidth, \neo@progressonsectionpage@linewidth);
281     \fill[fg] (0,0) rectangle (\neo@progressonsectionpage, \neo@progressonsectionpage@linewidth);
282   \end{tikzpicture}%
283   \tikzexternalenable%
284 }

```

The above code assumes that `\insertframenumbers` is less than or equal to `\inserttotalframenumbers`. However, this is not true on the first compile; in the absence of an `.aux` file, `\inserttotalframenumbers` defaults to 1. This behaviour could cause fatal errors for long presentations, as `\neo@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 `\inserttotalframenumbers`; presentations with over 4000 slides will still break on first compile, but users in that situation likely have deeper problems to solve.

```

285 \def\inserttotalframenumbers{100}

```

### 8.2.5 Block environments

block The three different block environments differ only in their colours. Rather  
block alerted than repeat the essentially the same template three times, we use the auxil-  
block example iary macro `\neo@block` to define all three templates.

```

286 \newlength{\neo@blocksep}
287 \newlength{\neo@blockadjust}
288 \setlength{\neo@blocksep}{0.75ex}
289 \setlength{\neo@blockadjust}{0.25ex}

```



```

290 \providecommand{\neo@strut}{%
291   \vphantom{ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz}}%
292 }
293 \newcommand{\neo@block}[1]{
294   \par\vskip\medskipamount%
295   \setlength{\parskip}{0pt}

```

If a background color is defined for the block title or body, we need to add a little bit of padding to the corresponding box. Ideally, this would be accomplished by setting `colsep=0.75ex`, which is intended to add “color separation space” only when the box has a colored background. Unfortunately, `colsep` also adds this separation if the background color is inherited, even if the inherited color is actually empty. (The technical reason for this boils down to the fact that the `\ifx` directive does not expand macros.)

To achieve the correct spacing for `alertblocks` and `exampleblocks` as well as for normal blocks, we have to begin the `beamercolorbox` differently based on whether `block title` has an empty background.

If the `block title` background is empty, or the user has explicitly removed the background from (e.g.) `block title alerted`, we just need to set a `rightskip` for a nice ragged-right block title.

```

296 \ifbeamercoloreempty[bg]{block title#1}{%
297   \begin{beamercolorbox}[rightskip=0pt plus 4em]{block title#1}}{%
298   \ifbeamercoloreempty[bg]{block title}{%
299     \begin{beamercolorbox}[rightskip=0pt plus 4em]{block title#1}%
300   }%
301 %   \end{macrocode}
302 %
303 %   Otherwise, if the |block title| has a background, we set the padding based
304 %   on |\neo@blockskip|. However, we have to visually compensate for
305 %   the |\neo@strut| added to the block title (see below) by
306 %   subtracting |\neo@blockadjust| from the top and bottom padding.
307 %
308 %   \begin{macrocode}
309 {%
310   \begin{beamercolorbox}[
311     sep=\dimexpr\neo@blocksep-\neo@blockadjust\relax,
312     leftskip=\neo@blockadjust,
313     rightskip=\dimexpr\neo@blockadjust plus 4em\relax

```

```

314   ]{block title#1}%
315   }}%
316 %   \end{macrocode}
317 %
318 %   We can now set the contents of the |block title|. The zero-width but
319 %   positive-height box |\neo@strut| ensures that the block title box
320 %   has a consistent height, even if it lacks punctuation, ascenders, or
321 %   descenders.
322 %
323 %   \begin{macrocode}
324       \usebeamerfont*{block title#1}%
325       \neo@strut%
326       \insertblocktitle%
327       \neo@strut%
328 \end{beamercolorbox}%
329 %   \end{macrocode}
330 %
331 %   Next, we typeset the |block body|. This the code is similar to, but simpler
332 %   than, the |block title| code since we don't need to adjust for any struts.
333 %
334 %   \begin{macrocode}
335 \nointerlineskip%
336 \ifbeamercolorempy[bg]{block body#1}{%
337   \begin{beamercolorbox}[vmode]{block body#1}}{
338   \ifbeamercolorempy[bg]{block body}{%
339     \begin{beamercolorbox}[vmode]{block body#1}%
340   }{%
341     \begin{beamercolorbox}[sep=\neo@blocksep, vmode]{block body#1}%
342     \vspace{-\neo@parskip}
343   }}%
344       \usebeamerfont{block body#1}%
345       \setlength{\parskip}{\neo@parskip}%
346 }

```

This concludes the auxiliary macro `\neo@block`. Finally, we define the block beamer templates using this macro.

```

347 \setbeamertemplate{block begin}{\neo@block{}}
348 \setbeamertemplate{block alerted begin}{\neo@block{ alerted}}
349 \setbeamertemplate{block example begin}{\neo@block{ example}}

```

```

350 \setbeamertemplate{block end}{\end{beamercolorbox}\vspace*{0.2ex}}
351 \setbeamertemplate{block alerted end}{\end{beamercolorbox}\vspace*{0.2ex}}
352 \setbeamertemplate{block example end}{\end{beamercolorbox}\vspace*{0.2ex}}

```

### 8.2.6 Lists and floats

```

353 \setbeamertemplate{itemize items}{\raise1pt\hbox{\vrule width 0.8ex height 0.8ex}}
354 \setbeamertemplate{itemize subitem}{\raise1pt\hbox{\vrule width 0.5ex height 0.5ex}}
355 \setbeamertemplate{itemize subsubitem}{\raise.5ex\hbox{\vrule width 1ex height 0.2ex}}
356 \defbeamertemplate{description item}{align left}{\insertdescriptionitem\hfill}
357 \setbeamertemplate{caption label separator}{: }
358 \setbeamertemplate{caption}[numbered]

```

### 8.2.7 Footnotes

```

359 \setbeamertemplate{footnote}{%
360   \parindent 0em\noindent%
361   \raggedright
362   \usebeamercolor{footnote}\hbox to 0.8em{\hfil\insertfootnotemark}\hangindent=0.8em
363 }

```

### 8.2.8 Text and spacing settings

```

364 \newlength{\neo@parskip}
365 \setlength{\neo@parskip}{0.5em}
366 \setlength{\parskip}{\neo@parskip}
367 \linespread{1.15}

```

By default, Beamer frames offer the `c` option to *almost* vertically center the text, but the placement is a little too high. To fix this, we redefine the `c` option to equalize `\beamer@frametopskip` and `\beamer@framebottomskip`. This solution was suggested by Enrico Gregorio in an answer to [this Stack Exchange question](#).

```

368 \define@key{beamerframe}{c}[true]{% centered
369   \beamer@frametopskip=0pt plus 1fill\relax%
370   \beamer@framebottomskip=0pt plus 1fill\relax%
371   \beamer@frametopskipautobreak=0pt plus .4\paperheight\relax%
372   \beamer@framebottomskipautobreak=0pt plus .6\paperheight\relax%
373   \def\beamer@initfirstlineunskip{}}%
374 }

```

## 8.2.9 Standout frames

**NEO** offers a custom frame format with large, centered text and an inverted background. To use it, add the key `standout` to the frame: `\begin{frame}[standout] ... \end{frame}`

`standout` Optional arguments to Beamer's frames are implemented using `\define@key` from the `keyval` package, which will execute code when the defined option is called. For the `standout` option, we begin a group, change the colors and fonts, use a plain slide, and set a alignment.

```
375 \providebool{neo@standout}
376 \define@key{beamerframe}{standout}[true]{%
377   \booltrue{neo@standout}
378   \begingroup
379     \setkeys{beamerframe}{c,plain}
380     \ifbeamercoloreempty{bg}{palette primary}{
381       \setbeamercolor{background canvas}{
382         use=palette primary,
383         bg=-palette primary.fg
384       }
385     }{
386       \setbeamercolor{background canvas}{
387         use=palette primary,
388         bg=palette primary.bg
389       }
390     }
391     \setbeamercolor{local structure}{
392       fg=palette primary.fg
393     }
394     \usebeamercolor[fg]{palette primary}
395     \makeatletter
396     \def\beamer@framenotesbegin{% at beginning of slide
397       \usebeamercolor[fg]{palette primary}
398       \gdef\beamer@noteitems{}%
399       \gdef\beamer@notes{}%
400     }
401     \makeatother
402 }
```

Then we just have to close the group after the `standout` slide is finished in order to restore the colours and fonts for the rest of the presentation.

Unfortunately, we cannot use or this (see <http://tex.stackexchange.com/questions/226319/>). Instead, we prepend the `\endgroup` to `\beamer@reseteeecodes`, which is run exactly once at the end of each slide.

```

403 \pretocmd{\beamer@reseteeecodes}{%
404   \ifbool{neo@standout}{
405     \endgroup
406     \boolfalse{neo@standout}
407   }{}
408 }{}{}

```

We set the fonts and the alignment on the inner content, in such a way that the speaker's note layout isn't affected by the custom formatting.

```

409 \AtBeginEnvironment{beamer@frameslide}{
410   \makeatletter
411   \usebeamercolor[fg]{normal text}
412   \gdef\beamer@noteitems{}%
413   \gdef\beamer@notes{}%
414   \makeatother
415   \ifbool{neo@standout}{
416     \centering
417     \usebeamerfont{standout}
418   }{}
419 }

```

### 8.2.10 Process package options

```

420 \neo@inner@setdefaults
421 \ProcessPgfPackageOptions{/neo/inner}

```

## 8.3 NEO 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.

### 8.3.1 Package dependencies

```

422 \RequirePackage{etoolbox}
423 \RequirePackage{calc}

```

```

424 \RequirePackage{pgfpages}
425 \RequirePackage{pgfopts}

```

### 8.3.2 Options

**icon** Adds an icon to the frametitle on each slide.

```

426 \pgfkeys{
427   /neo/outer/frametitle icon/.cd,
428   .is choice,
429   none/.code=\setbeamertemplate{frametitle icon}[none],
430   i4/.code=\setbeamertemplate{frametitle icon}[i4],
431   fau/.code=\setbeamertemplate{frametitle icon}[fau],
432 }

```

**numbering** Adds slide numbers to the bottom right of each slide.

```

433 \pgfkeys{
434   /neo/outer/numbering/.cd,
435   .is choice,
436   none/.code=\setbeamertemplate{frame numbering}[none],
437   counter/.code=\setbeamertemplate{frame numbering}[counter],
438   fraction/.code=\setbeamertemplate{frame numbering}[fraction],
439 }

```

**notes** Show notes in presentation

```

440 \pgfkeys{
441   /neo/outer/notes/.cd,
442   .is choice,
443   none/.code=\pgfkeysalso{notes=hide},
444   hide/.code=\setbeameroption{hide notes},
445   show/.code={\setbeamertemplate{note page}[print]\setbeameroption{show notes}},
446   only/.code={\setbeamertemplate{note page}[print]\setbeameroption{show only notes}},
447   preview-left/.code={\setbeamertemplate{note page}[default]\setbeameroption{show preview-left}},
448   preview-right/.code={\setbeamertemplate{note page}[default]\setbeameroption{show preview-right}},
449   preview-top/.code={\setbeamertemplate{note page}[default]\setbeameroption{show preview-top}},
450   preview-bottom/.code={\setbeamertemplate{note page}[default]\setbeameroption{show preview-bottom}},
451   preview-left-big/.code={\setbeamertemplate{note page}[preview-big]\setbeameroption{show preview-left-big}},
452   preview-right-big/.code={\setbeamertemplate{note page}[preview-big]\setbeameroption{show preview-right-big}},
453   preview-top-big/.code={\setbeamertemplate{note page}[preview-big]\setbeameroption{show preview-top-big}},
454   preview-bottom-big/.code={\setbeamertemplate{note page}[preview-big]\setbeameroption{show preview-bottom-big}},

```

```

455 left/.code={\setbeamertemplate{note page}[print]\setbeameroption{show notes on
456 right/.code={\setbeamertemplate{note page}[print]\setbeameroption{show notes on
457 top/.code={\setbeamertemplate{note page}[print]\setbeameroption{show notes on s
458 bottom/.code={\setbeamertemplate{note page}[print]\setbeameroption{show notes on
459 }

```

**footer** Adds additional presentation information to the footer

```

460 \pgfkeys{
461 /neo/outer/footer/.cd,
462 .is choice,
463 none/.code=\setbeamertemplate{frame footer}[none],
464 author/.code=\setbeamertemplate{frame footer}[author],
465 author title/.code=\setbeamertemplate{frame footer}[author title],
466 title/.code=\setbeamertemplate{frame footer}[title],
467 title section/.code=\setbeamertemplate{frame footer}[title section],
468 }

```

**footer style** Footer background color

```

469 \providebool{neo@standoutfooter}
470 \pgfkeys{
471 /neo/outer/footer style/.cd,
472 .is choice,
473 plain/.code={\boolfalse{neo@standoutfooter}\setbeamertemplate{footline}[plain]}
474 standout/.code={\booltrue{neo@standoutfooter}\setbeamertemplate{footline}[stand
475 }

```

**progressbar** Adds a progress bar to the top, bottom, or frametitle of each slide. In case this is used in combination with footer style, it must be set afterwards (e.g. to have the progressbar on top of the footer styled with standout).

```

476 \pgfkeys{
477 /neo/outer/progressbar/.cd,
478 .is choice,
479 none/.code={%
480 \setbeamertemplate{headline}[plain]
481 \setbeamertemplate{frametitle}[plain]
482 },
483 head/.code={\pgfkeys{/neo/outer/progressbar=none}
484 \addtobeamertemplate{headline}{}{}%

```

```

485     \usebeamertemplate*{progress bar in head/footer}
486   }
487 },
488 frametitle/.code={\pgfkeys{/neo/outer/progressbar=none}
489   \addtobeamertemplate{frametitle}{}{}%
490   \usebeamertemplate*{progress bar in head/footer}
491   }
492 },
493 foot/.code={\pgfkeys{/neo/outer/progressbar=none}
494   \ifbool{neo@standoutfooter}{}%
495   \addtobeamertemplate{footline}{\usebeamertemplate*{progress bar in head/footer}}{}%
496   }%
497   \addtobeamertemplate{footline}{}{\usebeamertemplate*{progress bar in head/footer}}
498   }
499 },
500 }

```

`\neo@outer@setdefaults` Sets default values for outer theme options.

```

501 \newcommand{\neo@outer@setdefaults}{
502   \pgfkeys{/neo/outer/.cd,
503     frametitle icon=none,
504     footer=none,
505     footer style=plain,
506     numbering=counter,
507     progressbar=none,
508   }
509 }

```

### 8.3.3 Head and footline

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

```

510 \setbeamertemplate{navigation symbols}{}

```

`frametitle icon` Templates for the icon on the right of the frame title.

```

511 \defbeamertemplate{frametitle icon}{none}{}
512 \defbeamertemplate{frametitle icon}{i4}{ \hfill\raisebox{-.25\height}{\includegraphics[width=1cm]{i4.png}}}
513 \defbeamertemplate{frametitle icon}{fau}{ \hfill\raisebox{-.25\height}{\includegraphics[width=1cm]{fau.png}}}

```



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

```

514 \defbeamertemplate{frame footer}{none}{}
515 \defbeamertemplate{frame footer}{author}{
516   \insertshortauthor%
517 }
518 \defbeamertemplate{frame footer}{author title}{
519   \insertshortauthor%
520   \hfill%
521   \insertshorttitle%
522   \hfill%
523 }
524 \defbeamertemplate{frame footer}{title}{
525   \insertshorttitle%
526 }
527 \defbeamertemplate{frame footer}{title section}{
528   \insertshorttitle%
529   \hfill%
530   \insertsection%
531   \hfill%
532 }
533 \defbeamertemplate{frame footer}{custom}[1]{ #1 }

```

Add strut to ensure that frame numbers don't jump

```

534 \newcommand{\neo@framenumberingstrut}{\vphantom{0123456789}}
535 \defbeamertemplate{frame numbering}{none}{}
536 \defbeamertemplate{frame numbering}{counter}{\neo@framenumberingstrut\insertframenum}
537 \defbeamertemplate{frame numbering}{fraction}{
538   \neo@framenumberingstrut\insertframenum/\inserttotalframenum
539 }

```

headline    Templates for the head- and footline at the top and bottom of each frame.

footline

```

540 \defbeamertemplate{headline}{plain}{}
541 \defbeamertemplate{footline}{plain}{%
542   \begin{beamercolorbox}[wd=\textwidth, sep=1.3ex]{footline}%
543     \usebeamerfont{page number in head/foot}%
544     \usebeamertemplate*{frame footer}
545     \hfill%

```

```

546 \parbox{.1\framewidth}{\hfill\usebeamertemplate*{frame numbering}}
547 \end{beamercolorbox}%
548 }
549 \defbeamertemplate{footline}{standout}{%
550 \begin{beamercolorbox}[wd=\textwidth, sep=1.3ex]{palette primary}%
551 \usebeamerfont{page number in head/foot}%
552 \usebeamertemplate*{frame footer}
553 \hfill%
554 \parbox{.1\framewidth}{\hfill\usebeamertemplate*{frame numbering}}
555 \end{beamercolorbox}%
556 }

```

### 8.3.4 Frametitle

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

```

557 \newlength{\neo@frametitle@padding}
558 \setlength{\neo@frametitle@padding}{2.2ex}
559 \newcommand{\neo@frametitlestrut@start}{
560 \rule{0pt}{\neo@frametitle@padding +%
561 \totalheightof{%
562 \ifcsdef{neo@frametitleformat}{\neo@frametitleformat X}{X}%
563 }%
564 }%
565 }
566 \newcommand{\neo@frametitlestrut@end}{
567 \rule[-\neo@frametitle@padding]{0pt}{\neo@frametitle@padding}
568 }
569 \defbeamertemplate{frametitle}{plain}{%
570 \nointerlineskip%
571 \begin{beamercolorbox}[%
572 wd=\paperwidth,%
573 sep=0pt,%
574 leftskip=\neo@frametitle@padding,%
575 rightskip=\neo@frametitle@padding,%
576 ]{frametitle}%
577 \neo@frametitlestrut@start%
578 \insertframetitle%
579 \usebeamertemplate*{frametitle icon}%

```

```

580 \nolinebreak%
581 \neo@frametitlestrut@end%
582 \end{beamercolorbox}%
583 }
584 \setbeamertemplate{frametitle continuation}{%
585 \usebeamerfont{frametitle}
586 {\normalfont (\insertcontinuationcount)}
587 }

```

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.

```

588 \newlength{\neo@progressinheadfoot}
589 \newlength{\neo@progressinheadfoot@linewidth}
590 \setlength{\neo@progressinheadfoot@linewidth}{0.8pt}
591 \setbeamertemplate{progress bar in head/foot}{
592 \nointerlineskip
593 \setlength{\neo@progressinheadfoot}{%
594 \paperwidth * \ratio{\insertframenum pt}{\inserttotalframenum pt}%
595 }%
596 \begin{beamercolorbox}[wd=\paperwidth]{progress bar in head/foot}
597 \tikzexternaldisable%
598 \begin{tikzpicture}
599 \fill[bg] (0,0) rectangle (\paperwidth, \neo@progressinheadfoot@linewidth);
600 \fill[fg] (0,0) rectangle (\neo@progressinheadfoot, \neo@progressinheadfoot@linewidth);
601 \end{tikzpicture}%
602 \tikzexternalenable%
603 \end{beamercolorbox}
604 }

```

custom notes    Templates for note pages

```

605 \defbeamertemplate{note page}{preview-big}
606 {%
607 {%
608 \scriptsize
609 \usebeamerfont{note title}\usebeamercolor[fg]{note title}%
610 \ifbeamercolorempy[bg]{note title}{}{}%
611 \insertvrule{.45\paperheight}{note title.bg}%
612 \vskip-.45\paperheight%

```

```

613     \nointerlineskip%
614 }%
615 \vbox{
616     \hfill\insertslideintonotes{0.45}\hskip-\Gm@rmargin\hskip0pt%
617     \vskip-0.45\paperheight%
618     \nointerlineskip
619     \begin{pgfpicture}{0cm}{0cm}{0cm}{0cm}
620         \begin{pgflowlevelscope}{\pgftransformrotate{90}}
621             {\pgftransformshift{\pgfpoint{-2cm}{0.2cm}}}%
622             \pgftext[base,left]{\usebeamerfont{note date}\usebeamerfont{note date}}
623         \end{pgflowlevelscope}
624     \end{pgfpicture}}
625 \nointerlineskip
626 \vbox to .45\paperheight{\vskip0.5em
627     \hbox{\insertshorttitle[width=8cm]}}%
628     \setbox\beamer@tempbox=\hbox{\insertsection}%
629     \hbox{\ifdim\wd\beamer@tempbox>1pt{\hskip4pt\raise3pt\hbox{\vrule
630         width0.4pt height7pt\vrule width 9pt
631         height0.4pt}}\hskip1pt\hbox{\begin{minipage}[t]{7.5cm}\def\breakhere{}}\
632     }%
633     \setbox\beamer@tempbox=\hbox{\insertsubsection}%
634     \hbox{\ifdim\wd\beamer@tempbox>1pt{\hskip17.4pt\raise3pt\hbox{\vrule
635         width0.4pt height7pt\vrule width 9pt
636         height0.4pt}}\hskip1pt\hbox{\begin{minipage}[t]{7.5cm}\def\breakhere{}}\
637     }%
638     \setbox\beamer@tempbox=\hbox{\insertshortframetitle}%
639     \hbox{\ifdim\wd\beamer@tempbox>1pt{\hskip30.8pt\raise3pt\hbox{\vrule
640         width0.4pt height7pt\vrule width 9pt
641         height0.4pt}}\hskip1pt\hbox{\insertshortframetitle[width=7cm]}}\fi%
642     }%
643     \vfil}%
644 }%
645 \ifbeamercolorempy[bg]{note page}{\fi}%
646     \nointerlineskip%
647     \insertvrule{.55\paperheight}{note page.bg}%
648     \vskip-.55\paperheight%
649 }%
650 \vskip.25em
651 \nointerlineskip
652 \insertnote

```

```

653 }
654 \defbeamertemplate{note page}{print}
655 {%
656   {%
657     \nointerlineskip%
658     \begin{beamercolorbox}[%
659       wd=\paperwidth,%
660       sep=0pt,%
661       leftskip=\neo@frametitle@padding,%
662       rightskip=\neo@frametitle@padding,%
663     ]{note title}%
664     \usebeamerfont{frametitle}%
665     \neo@frametitlestrut@start%
666     \insertframetitle%
667     \usebeamertemplate*{frametitle icon}%
668     \nolinebreak%
669     \neo@frametitlestrut@end%
670   \end{beamercolorbox}%
671   }%
672   \insertnote%
673   \vfill%
674   \begin{beamercolorbox}[wd=\paperwidth, sep=3ex]{footline}%
675     \usebeamerfont{page number in head/foot}%
676     \usebeamertemplate*{frame footer}
677     \hfill%
678     \usebeamertemplate*{frame numbering}
679   \end{beamercolorbox}%
680   \vskip4pt%
681 }

```

appendix Removes page numbering and per-slide progress bars when `\appendix` is called. This makes it easier to include additional “backup slides” at the end of the presentation, especially in conjunction with the package `appendixnumberbeamer`.

```

682 \AtBeginDocument{%
683   \apptocmd{\appendix}{%
684     \pgfkeys{%
685       /neo/outer/.cd,
686       numbering=none,
687       progressbar=none}

```

```

688     }{}{}
689 }

```

### 8.3.5 Process package options

```

690 \neo@outer@setdefaults
691 \ProcessPgfPackageOptions{/neo/outer}

```

## 8.4 NEO font theme

A beamer font theme sets the style of the font used in the document.

### 8.4.1 Package dependencies

```

692 \RequirePackage{etoolbox}
693 \RequirePackage{ifxetex}
694 \RequirePackage{ifluatex}
695 \RequirePackage{pgfopts}

```

### 8.4.2 Load Fira fonts

If the presentation is compiled with Xe $\LaTeX$  or Lua $\LaTeX$ , the fontspec package is loaded and we search for the Fira fonts.

```

696 \ifboolexpr{bool {xetex} or bool {luatex}}{
697   \@ifpackageloaded{fontspec}{
698     \PassOptionsToPackage{no-math}{fontspec}
699   }{
700     \RequirePackage[no-math]{fontspec}
701   }

702   \IfFileExists{FiraSans-Regular.otf}{
703     \defaultfontfeatures{
704       Scale      = 1.0,
705       Extension = .otf
706     }
707   }{
708     \PackageWarning{beamerthemeneo}{%
709       FiraSans font not found in path, therefore using system fonts. %
710       Make sure you have the fonts installed.%
711     }
712   }

```

```

713 \setmonofont
714   [ Numbers = {Monospaced,Lining},
715     UprightFont   = *-Regular ,
716     ItalicFont    = *-Regular ,
717     BoldFont      = *-Medium ,
718     BoldItalicFont = *-Medium ,
719   ]
720   {FiraMono}
721 \newcommand{\neo@fontsave}{
722   \let\firaneofamily\sfddefault
723   \renewcommand*\familydefault{\firaneofamily}
724 }
725 \newcommand{\neo@fontlight}{
726   \setsansfont[
727     Numbers = {OldStyle, Monospaced},
728     UprightFont   = *-Light ,
729     ItalicFont    = *-LightItalic ,
730     BoldFont      = *-Regular ,
731     BoldItalicFont = *-RegularItalic ,
732   ]{FiraSans}
733   \neo@fontsave
734 }
735 \newcommand{\neo@fontbook}{
736   \setsansfont[
737     Numbers = {OldStyle, Monospaced},
738     UprightFont   = *-Book ,
739     ItalicFont    = *-BookItalic ,
740     BoldFont      = *-Medium ,
741     BoldItalicFont = *-MediumItalic ,
742   ]{FiraSans}
743   \neo@fontsave
744 }
745 \newcommand{\neo@fontregular}{
746   \setsansfont[
747     Numbers = {OldStyle, Monospaced},
748     UprightFont   = *-Regular ,
749     ItalicFont    = *-RegularItalic ,
750     BoldFont      = *-SemiBold ,
751     BoldItalicFont = *-SemiBoldItalic ,
752   ]{FiraSans}

```

```

753   \neo@fontsave
754 }
755 \AtBeginEnvironment{tabular}{%
756   \addfontfeature{Numbers={Monospaced}}}%
757 }
758 }{%
759   \RequirePackage[utf8]{inputenc}
760   \IfFileExists{FiraSans.sty}{
761     \RequirePackage[T1]{fontenc}
762     \RequirePackage[sfdefault]{FiraSans}
763     \RequirePackage[nomap,lining]{FiraMono}
764     \def\bfseries@tt{mb}
765     \newcommand{\neo@fontsave}{
766       \edef\familydefault{\sfdefault}
767       \edef\seriesdefault{\mdseries@sf}
768     }
769     \newcommand{\neo@fontlight}{
770       \def\mdseries@sf{l}
771       \def\bfseries@sf{m}
772       \neo@fontsave
773     }
774     \newcommand{\neo@fontbook}{
775       \def\bfseries@sf{mb}
776       \neo@fontsave
777     }
778     \newcommand{\neo@fontregular}{
779       \def\mdseries@sf{m}
780       \def\bfseries@sf{sb}
781       \neo@fontsave
782     }
783   }{
784     \PackageWarning{beamerthemeneo}{%
785       You need to install the Fira Fonts package or compile with XeLaTeX or %
786       LuaLaTeX to use the included Fira fonts%
787     }
788   }
789 }

```

This concludes the portion of the code which is only run when compiled with Xe<sub>Λ</sub>T<sub>Ε</sub>X or Lua<sub>Λ</sub>T<sub>Ε</sub>X. The remainder of this package applies regardless of the com-



piling engine.

### 8.4.3 General font definitions

```
790 \setbeamerfont{title}{size=\Large,%  
791           series=\bfseries}  
792 \setbeamerfont{author}{size=\small}  
793 \setbeamerfont{date}{size=\small}  
794 \setbeamerfont{section title}{size=\Large,%  
795           series=\bfseries}  
796 \setbeamerfont{block title}{size=\normalsize,%  
797           series=\bfseries}  
798 \setbeamerfont{block title alerted}{size=\normalsize,%  
799           series=\bfseries}  
800 \setbeamerfont*{subtitle}{size=\large}  
801 \setbeamerfont{frametitle}{size=\large,%  
802           series=\bfseries}  
803 \setbeamerfont{caption}{size=\small}  
804 \setbeamerfont{caption name}{series=\bfseries}  
805 \setbeamerfont{description item}{series=\bfseries}  
806 \setbeamerfont{page number in head/foot}{size=\scriptsize}  
807 \setbeamerfont{bibliography entry author}{size=\normalsize,%  
808           series=\normalfont}  
809 \setbeamerfont{bibliography entry title}{size=\normalsize,%  
810           series=\bfseries}  
811 \setbeamerfont{bibliography entry location}{size=\normalsize,%  
812           series=\normalfont}  
813 \setbeamerfont{bibliography entry note}{size=\small,%  
814           series=\normalfont}  
815 \setbeamerfont{standout}{size=\Large,%  
816           series=\bfseries}
```

### 8.4.4 Font style options

`titleformat title` Controls the overall font style.

```
817 \pgfkeys{  
818   /neo/font/style/.cd,  
819   .is choice,  
820   light/.code={\neo@fontlight},  
821   book/.code={\neo@fontbook},
```

```

822     regular/.code={\neo@fontregular},
823 }

```

#### 8.4.5 Title format options

`titleformat title` Controls the format of the title.

```

824 \pgfkeys{
825   /neo/font/titleformat title/.cd,
826   .is choice,
827   regular/.code={%
828     \let\neo@titleformat\@empty%
829     \setbeamerfont{title}{shape=\normalfont}%
830   },
831   smallcaps/.code={%
832     \let\neo@titleformat\@empty%
833     \setbeamerfont{title}{shape=\scshape}%
834   },
835   allsmallcaps/.code={%
836     \let\neo@titleformat\lowercase%
837     \setbeamerfont{title}{shape=\scshape}%
838     \PackageWarning{beamerthemeneo}{%
839       Be aware that titleformat title=allsmallcaps can lead to problems%
840     }
841   },
842   allcaps/.code={%
843     \let\neo@titleformat\uppercase%
844     \setbeamerfont{title}{shape=\normalfont}%
845     \PackageWarning{beamerthemeneo}{%
846       Be aware that titleformat title=allcaps can lead to problems%
847     }
848   },
849 }

```

`titleformat subtitle` Control the format of the subtitle.

```

850 \pgfkeys{
851   /neo/font/titleformat subtitle/.cd,
852   .is choice,
853   regular/.code={%
854     \let\neo@subtitleformat\@empty%

```

```

855     \setbeamerfont{subtitle}{shape=\normalfont}%
856 },
857 smallcaps/.code={%
858     \let\neo@subtitleformat\@empty%
859     \setbeamerfont{subtitle}{shape=\scshape}%
860 },
861 allsmallcaps/.code={%
862     \let\neo@subtitleformat\lowercase%
863     \setbeamerfont{subtitle}{shape=\scshape}%
864     \PackageWarning{beamerthemeneo}{%
865         Be aware that titleformat subtitle=allsmallcaps can lead to problems%
866     }
867 },
868 allcaps/.code={%
869     \let\neo@subtitleformat\uppercase%
870     \setbeamerfont{subtitle}{shape=\normalfont}%
871     \PackageWarning{beamerthemeneo}{%
872         Be aware that titleformat subtitle=allcaps can lead to problems%
873     }
874 },
875 }

```

`titleformat section` Controls the format of the section title.

```

876 \pgfkeys{
877   /neo/font/titleformat section/.cd,
878   .is choice,
879   regular/.code={%
880       \let\neo@sectiontitleformat\@empty%
881       \setbeamerfont{section title}{shape=\normalfont}%
882   },
883   smallcaps/.code={%
884       \let\neo@sectiontitleformat\@empty%
885       \setbeamerfont{section title}{shape=\scshape}%
886   },
887   allsmallcaps/.code={%
888       \let\neo@sectiontitleformat\MakeLowercase%
889       \setbeamerfont{section title}{shape=\scshape}%
890       \PackageWarning{beamerthemeneo}{%
891         Be aware that titleformat section=allsmallcaps can lead to problems%

```

```

892     }
893   },
894   allcaps/.code={%
895     \let\neo@sectiontitleformat\MakeUppercase%
896     \setbeamerfont{section title}{shape=\normalfont}%
897     \PackageWarning{beamerthemeneo}{%
898       Be aware that titleformat section=allcaps can lead to problems%
899     }
900   },
901 }

```

`frametitleformat` Control the format of the frame title.

```

902 \pgfkeys{
903   /neo/font/titleformat frame/.cd,
904   .is choice,
905   regular/.code={%
906     \let\neo@frametitleformat\@empty%
907     \setbeamerfont{frametitle}{shape=\normalfont}%
908   },
909   smallcaps/.code={%
910     \let\neo@frametitleformat\@empty%
911     \setbeamerfont{frametitle}{shape=\scshape}%
912   },
913   allsmallcaps/.code={%
914     \let\neo@frametitleformat\MakeLowercase%
915     \setbeamerfont{frametitle}{shape=\scshape}%
916     \PackageWarning{beamerthemeneo}{%
917       Be aware that titleformat frame=allsmallcaps can lead to problems%
918     }
919   },
920   allcaps/.code={%
921     \let\neo@frametitleformat\MakeUppercase%
922     \setbeamerfont{frametitle}{shape=\normalfont}%
923     \PackageWarning{beamerthemeneo}{%
924       Be aware that titleformat frame=allcaps can lead to problems%
925     }
926   },
927 }

```

`titleformat aliases` Allows `titleformat title` et al. to be used in the `\usetheme` declaration, where  $\TeX$  automatically removes all spaces.

```
928 \pgfkeys{
929   /neo/font/.cd,
930   titleformattitle/.code=\pgfkeysalso{titleformat title=#1},
931   titleformatsubtitle/.code=\pgfkeysalso{titleformat subtitle=#1},
932   titleformatsection/.code=\pgfkeysalso{titleformat section=#1},
933   titleformatframe/.code=\pgfkeysalso{titleformat frame=#1},
934 }
```

`\neo@font@setdefaults` Sets default values for font theme options.

```
935 \newcommand{\neo@font@setdefaults}{
936   \pgfkeys{/neo/font/.cd,
937     style=book,
938     titleformat title=regular,
939     titleformat subtitle=regular,
940     titleformat section=regular,
941     titleformat frame=regular,
942   }
943 }
```

We first define hooks to change the case format of the titles.

```
944 \def\neo@titleformat#1{#1}
945 \def\neo@subtitleformat#1{#1}
946 \def\neo@sectiontitleformat#1{#1}
947 \def\neo@frametitleformat#1{#1}
```

To make the uppercase and lowercase macros work in the title, subtitle, etc., we have to patch the appropriate beamer commands that set their values. This solution was suggested by Enrico Gregorio in an answer to [this StackExchange question](#).

```
948 \patchcmd{\beamer@title}%
949   {\def\inserttitle{#2}}%
950   {\def\inserttitle{\neo@titleformat{#2}}}%
951   {}%
952   {\PackageError{beamerfontthemeneo}{Patching title failed}\@ehc}
953 \patchcmd{\beamer@subtitle}%
```

```

954 {\def\insertsubtitle{#2}}%
955 {\def\insertsubtitle{\neo@subtitleformat{#2}}}%
956 {}%
957 {\PackageError{beamerfontthemeneo}{Patching subtitle failed}\@ehc}
958 \patchcmd{\sectionentry}
959 {\def\insertsectionhead{#2}}
960 {\def\insertsectionhead{\neo@sectiontitleformat{#2}}}
961 {}
962 {\PackageError{beamerfontthemeneo}{Patching section title failed}\@ehc}
963 \@tempswafalse
964 \patchcmd{\beamer@section}
965 {\edef\insertsectionhead{\noexpand\hyperlink{Navigation\the\c@page}}{\unexpanded{#
966 {\edef\insertsectionhead{\noexpand\hyperlink{Navigation\the\c@page}}{%
967 \noexpand\neo@sectiontitleformat{\unexpanded{#1}}}}}
968 {\@tempswatrue}
969 {}
970 \patchcmd{\beamer@section}
971 {\def\insertsectionhead{\hyperlink{Navigation\the\c@page}}{#1}}
972 {\def\insertsectionhead{\hyperlink{Navigation\the\c@page}}{%
973 \neo@sectiontitleformat{#1}}}}
974 {\@tempswatrue}
975 {}
976 \patchcmd{\beamer@section}
977 {\edef\insertsectionhead{\noexpand\hyperlink{Navigation\the\c@page}}{\unexpanded{#
978 {\edef\insertsectionhead{\noexpand\hyperlink{Navigation\the\c@page}}{%
979 \noexpand\neo@sectiontitleformat{\unexpanded{#1}}}}}
980 {\@tempswatrue}
981 {}
982 \patchcmd{\beamer@section}
983 {\protected@edef\insertsectionhead{\noexpand\hyperlink{Navigation\the\c@page}}{#1}
984 {\protected@edef\insertsectionhead{\noexpand\hyperlink{Navigation\the\c@page}}{%
985 \noexpand\neo@sectiontitleformat{#1}}}}
986 {\@tempswatrue}
987 {}
988 \if@tempswa\else
989 \PackageError{beamerfontthemeneo}{Patching section title failed}\@ehc
990 \fi
991 \@tempswafalse
992 \patchcmd{\beamer@subsection}
993 {\edef\insertsubsectionhead{\noexpand\hyperlink{Navigation\the\c@page}}{\unexpanded

```

```

994 {\edef\insertsubsectionhead{\noexpand\hyperlink{Navigation\the\c@page}{%
995   \noexpand\neo@sectiontitleformat{\unexpanded{#1}}}}
996 {\@tempswattrue}
997 {}
998 \patchcmd{\beamer@section}
999 {\def\insertsubsectionhead{\hyperlink{Navigation\the\c@page}{#1}}}
1000 {\def\insertsubsectionhead{\hyperlink{Navigation\the\c@page}{%
1001   \neo@sectiontitleformat{#1}}}}
1002 {\@tempswattrue}
1003 {}
1004 \patchcmd{\beamer@section}
1005 {\protected@edef\insertsubsectionhead{\noexpand\hyperlink{Navigation\the\c@page}{%
1006   \protected@edef\insertsubsectionhead{\noexpand\hyperlink{Navigation\the\c@page}{%
1007     \noexpand\neo@sectiontitleformat{#1}}}}
1008   {\@tempswattrue}
1009   {}
1010   \if@tempswa\else
1011     \PackageError{beamerfontthemeneo}{Patching section title failed}\@ehc
1012   \fi

```

Similarly, to make the \MakeLowercase and \MakeUppercase macros work in the frame title we have to patch \beamer@@frametitle.

```

1013 \patchcmd{\beamer@@frametitle}
1014 {{%
1015   \gdef\insertframetitle{{#2\ifnum\beamer@autobreakcount>0\relax}}\space%
1016   \usebeamertemplate*{frametitle continuation}\fi}}%
1017 \gdef\beamer@frametitle{#2}%
1018 \gdef\beamer@shortframetitle{#1}%
1019 }}
1020 {{%
1021   \gdef\insertframetitle{{\neo@frametitleformat{#2}\ifnum%
1022     \beamer@autobreakcount>0\relax}}\space%
1023   \usebeamertemplate*{frametitle continuation}\fi}}%
1024 \gdef\beamer@frametitle{#2}%
1025 \gdef\beamer@shortframetitle{#1}%
1026 }}
1027 {}
1028 {\PackageError{beamerfontthemeneo}{Patching frame title failed}\@ehc}

```

### 8.4.6 Process package options

```
1029 \neo@font@setdefaults
1030 \ProcessPgfPackageOptions{/neo/font}
```

## 8.5 NEO color theme

### 8.5.1 Package dependencies

```
1031 \RequirePackage{pgfopts}
```

### 8.5.2 Options

**block** Optionally adds a light grey background to block environments like theorem and example.

```
1032 \pgfkeys{
1033   /neo/color/block/.cd,
1034   .is choice,
1035   transparent/.code=\neo@block@transparent,
1036   fill/.code=\neo@block@fill,
1037 }
```

**colors** Provides the option to have a dark background and light foreground instead of the reverse. White is similar to light but uses a pure white background.

```
1038 \pgfkeys{
1039   /neo/color/background/.cd,
1040   .is choice,
1041   dark/.code=\neo@colors@dark,
1042   light/.code=\neo@colors@bright,
1043   white/.code=\neo@colors@white,
1044 }
```

**\neo@color@setdefaults** Sets default values for color theme options.

```
1045 \newcommand{\neo@color@setdefaults}{
1046   \pgfkeys{/neo/color/.cd,
1047     background=light,
1048     block=transparent,
1049   }
1050 }
```



### 8.5.3 Base colors

```
1051
1052 \definecolor{nDarkGrey}{RGB}{152,164,174}
1053 \definecolor{nGrey}{RGB}{210,213,215}
1054 \definecolor{nLightGrey}{RGB}{235,236,238}
1055
1056 \definecolor{nDarkRed}{RGB}{141,20,41}
1057 \definecolor{nRed}{RGB}{201,169,147}
1058 \definecolor{nLightRed}{RGB}{237,231,222}
1059
1060 \definecolor{nDarkGreen}{RGB}{0,155,119}
1061 \definecolor{nGreen}{RGB}{170,207,189}
1062 \definecolor{nLightGreen}{RGB}{229,239,234}
1063
1064 \definecolor{nDarkBlue}{RGB}{0,56,101}
1065 \definecolor{nBlue}{RGB}{144,167,198}
1066 \definecolor{nLightBlue}{RGB}{221,229,240}
1067
1068 \definecolor{nDarkCyan}{RGB}{0,177,235}
1069 \definecolor{nCyan}{RGB}{180,214,245}
1070 \definecolor{nLightCyan}{RGB}{234,243,252}
1071
1072 \definecolor{nDarkYellow}{RGB}{201,147,19}
1073 \definecolor{nYellow}{RGB}{217,198,137}
1074 \definecolor{nLightYellow}{RGB}{243,238,223}
1075
1076 \definecolor{nBlack}{HTML}{011F32}
1077 \definecolor{nWhite}{RGB}{250,250,250}
```

### 8.5.4 Alias colors

Support the colors provided by the old i4 beamer theme.

```
1078 \colorlet{i4red}{nDarkRed}
1079 \colorlet{i4green}{nDarkGreen}
1080 \colorlet{i4blue}{nDarkBlue}
1081 \colorlet{i4cyan}{nDarkCyan}
1082 \colorlet{i4yellow}{nDarkYellow}
1083 \colorlet{i4grey}{nDarkGrey}
1084 \definecolor{darkred}{rgb}{0.8,0,0}
```

```
1085 \colorlet{beamergreen}{green!50!black}
```

### 8.5.5 Base styles

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

```
1086 \newcommand{\neo@colors@dark}{
1087   \setbeamercolor{normal text}{%
1088     fg=nWhite,
1089     bg=nBlack
1090   }
1091   \setbeamercolor{normal item}{%
1092     fg=nWhite,
1093     bg=nDarkBlue
1094   }
1095   \usebeamercolor[fg]{normal text}
1096 }
1097 \newcommand{\neo@colors@bright}{
1098   \setbeamercolor{normal text}{%
1099     fg=nBlack,
1100     bg=nWhite
1101   }
1102   \setbeamercolor{normal item}{%
1103     fg=nDarkBlue,
1104     bg=nWhite
1105   }
1106   \renewcommand{\neo@colors@light}{\neo@colors@bright}
1107 }
1108 \renewcommand{\neo@colors@light}{\neo@colors@bright}
1109 \newcommand{\neo@colors@white}{
1110   \setbeamercolor{normal text}{%
1111     fg=nBlack,
1112     bg=white
1113   }
1114   \setbeamercolor{normal item}{%
1115     fg=nDarkBlue,
1116     bg=white
1117   }
1118   \renewcommand{\neo@colors@light}{\neo@colors@white}
```

```

1119 }
1120 \setbeamercolor{alerted text}{%
1121   fg=nDarkRed
1122 }
1123 \setbeamercolor{example text}{%
1124   fg=nDarkYellow
1125 }
1126 \setbeamercolor{note title}{%
1127   fg=nDarkBlue,
1128   bg=nGrey
1129 }
1130 \setbeamercolor{note page}{%
1131   fg=nBlack,
1132   bg=nLightGrey
1133 }

```

### 8.5.6 Derived colors

The titles and structural elements (e.g. itemize bullets) are set in the same color as normal text and normal item. This would ideally be done by setting normal text and normal item 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`.

```

1134 \setbeamercolor{titlelike}{use=normal text, parent=normal text}
1135 \setbeamercolor{author}{use=normal text, parent=normal text}
1136 \setbeamercolor{date}{use=normal text, parent=normal text}
1137 \setbeamercolor{institute}{use=normal text, parent=normal text}
1138 \setbeamercolor{structure}{use=normal item, fg=normal item.fg}

```

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

```

1139 \setbeamercolor{palette primary}{%
1140   use=normal text,
1141   fg=normal text.bg,
1142   bg=nDarkBlue
1143 }
1144 \setbeamercolor{frametitle}{%
1145   use=palette primary,

```

```

1146 parent=palette primary
1147 }

```

The **NEO** 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`.

```

1148 \setbeamercolor{progress bar}{%
1149   use=normal text,
1150   fg=nDarkBlue,
1151   bg=nLightBlue
1152 }
1153 \setbeamercolor{title separator}{
1154   use=progress bar,
1155   parent=progress bar
1156 }
1157 \setbeamercolor{progress bar in head/foot}{%
1158   use=normal text.fg,
1159   fg=nBlack,
1160   parent=progress bar
1161 }
1162 \setbeamercolor{progress bar in section page}{
1163   use=progress bar,
1164   parent=progress bar
1165 }

```

Block environments such as `theorem` and `example` have no background color by default. The option `block=fill` sets a background color based on the background and foreground of `normal text`. The option `block=transparent` reverts the block environments to an empty background, which can be useful if changing colors mid-presentation.

```

1166 \newcommand{\neo@block@transparent}{
1167   \setbeamercolor{block title}{%
1168     use=normal text,
1169     fg=nDarkBlue,
1170     bg=
1171   }
1172   \setbeamercolor{block title alerted}{%
1173     use={block title, alerted text},

```

```

1174     bg=block title.bg,
1175     fg=alerted text.fg
1176 }
1177 \setbeamercolor{block title example}{%
1178     use={block title, example text},
1179     bg=block title.bg,
1180     fg=example text.fg
1181 }
1182 \setbeamercolor{block body}{
1183     bg=
1184 }
1185 \setbeamercolor{block body alerted}{
1186     use=block body,
1187     parent=block body
1188 }
1189 \setbeamercolor{block body example}{
1190     use=block body,
1191     parent=block body
1192 }
1193 }
1194 \newcommand{\neo@block@fill}{
1195     \setbeamercolor{block title}{%
1196         use=normal text,
1197         fg=nDarkBlue,
1198         bg=nGrey
1199     }
1200     \setbeamercolor{block title alerted}{%
1201         use={block title, alerted text},
1202         bg=alerted text.fg,
1203         fg=alerted text.bg
1204     }
1205     \setbeamercolor{block title example}{%
1206         use={block title, example text},
1207         bg=example text.fg,
1208         fg=example text.bg
1209     }
1210     \setbeamercolor{block body}{
1211         use={block title, normal text},
1212         bg=nLightGrey
1213     }

```

```

1214 \setbeamercolor{block body alerted}{
1215     use=block body,
1216     parent=block body,
1217     bg=nRed!50,
1218 }
1219 \setbeamercolor{block body example}{
1220     use=block body,
1221     parent=block body,
1222     bg=nYellow!50
1223 }
1224 }
1225

```

### Footnotes

```

1226 \setbeamercolor{footnote}{fg=normal text.fg!90}
1227 \setbeamercolor{footnote mark}{fg=.
```

We also reset the bibliography colors in order to pick up the surrounding colors at the time of use. This prevents us having to set the correct color in normal and standout mode.

```

1228 \setbeamercolor{bibliography entry author}{fg=, bg=}
1229 \setbeamercolor{bibliography entry title}{fg=, bg=}
1230 \setbeamercolor{bibliography entry location}{fg=, bg=}
1231 \setbeamercolor{bibliography entry note}{fg=, bg=}

```

### 8.5.7 Process package options

```

1232 \neo@color@setdefaults
1233 \ProcessPgfPackageOptions{/neo/color}
1234 \mode<all>

```

## 8.6 Tol pgfplots theme

Paul Tol's 12-color palette<sup>1</sup> is as follows:

```

1235 \definecolor{TolDarkPurple}{HTML}{332288}
1236 \definecolor{TolDarkBlue}{HTML}{6699CC}
1237 \definecolor{TolLightBlue}{HTML}{88CCEE}

```

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

```

1238 \definecolor{TolLightGreen}{HTML}{44AA99}
1239 \definecolor{TolDarkGreen}{HTML}{117733}
1240 \definecolor{TolDarkBrown}{HTML}{999933}
1241 \definecolor{TolLightBrown}{HTML}{DDCC77}
1242 \definecolor{TolDarkRed}{HTML}{661100}
1243 \definecolor{TolLightRed}{HTML}{CC6677}
1244 \definecolor{TolLightPink}{HTML}{AA4466}
1245 \definecolor{TolDarkPink}{HTML}{882255}
1246 \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.

```

1247 \pgfplotscreateplotcyclelist{mbarplot cycle}{%
1248   {draw=TolDarkBlue,    fill=TolDarkBlue!70},
1249   {draw=TolLightBrown,  fill=TolLightBrown!70},
1250   {draw=TolLightGreen,  fill=TolLightGreen!70},
1251   {draw=TolDarkPink,    fill=TolDarkPink!70},
1252   {draw=TolDarkPurple,  fill=TolDarkPurple!70},
1253   {draw=TolDarkRed,     fill=TolDarkRed!70},
1254   {draw=TolDarkBrown,   fill=TolDarkBrown!70},
1255   {draw=TolLightRed,    fill=TolLightRed!70},
1256   {draw=TolLightPink,   fill=TolLightPink!70},
1257   {draw=TolLightPurple, fill=TolLightPurple!70},
1258   {draw=TolLightBlue,   fill=TolLightBlue!70},
1259   {draw=TolDarkGreen,   fill=TolDarkGreen!70},
1260 }

```

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

```

1261 \pgfplotscreateplotcyclelist{mlineplot cycle}{%
1262   {TolDarkBlue, mark=*, mark size=1.5pt},
1263   {TolLightBrown, mark=square*, mark size=1.3pt},
1264   {TolLightGreen, mark=triangle*, mark size=1.5pt},
1265   {TolDarkBrown, mark=diamond*, mark size=1.5pt},
1266 }

```

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.

```
1267 \pgfplotsset{
1268   compat=1.9,
```

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

```
1269   mlineplot/.style={
1270     mbaseplot,
1271     xmajorgrids=true,
1272     ymajorgrids=true,
1273     major grid style={dotted},
1274     axis x line=bottom,
1275     axis y line=left,
1276     legend style={
1277       cells={anchor=west},
1278       draw=none
1279     },
1280     cycle list name=mlineplot cycle,
1281   },
```

`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`.

```
1282   mbarplot base/.style={
1283     mbaseplot,
1284     bar width=6pt,
1285     axis y line*=none,
1286   },
1287   mbarplot/.style={
1288     mbarplot base,
1289     ybar,
1290     xmajorgrids=false,
1291     ymajorgrids=true,
1292     area legend,
1293     legend image code/.code={%
1294       \draw[#1] (0cm,-0.1cm) rectangle (0.15cm,0.1cm);
1295     },
```



```

1296     cycle list name=mbarplot cycle,
1297 },
1298 horizontal mbarplot/.style={
1299     mbarplot base,
1300     xmajorgrids=true,
1301     ymajorgrids=false,
1302     xbar stacked,
1303     area legend,
1304     legend image code/.code={%
1305         \draw[#1] (0cm,-0.1cm) rectangle (0.15cm,0.1cm);
1306     },
1307     cycle list name=mbarplot cycle,
1308 },

```

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

```

1309 mbaseplot/.style={
1310     legend style={
1311         draw=none,
1312         fill=none,
1313         cells={anchor=west},
1314     },
1315     x tick label style={
1316         font=\footnotesize
1317     },
1318     y tick label style={
1319         font=\footnotesize
1320     },
1321     legend style={
1322         font=\footnotesize
1323     },
1324     major grid style={
1325         dotted,
1326     },
1327     axis x line*=bottom,
1328 },
1329 disable thousands separator/.style={
1330     /pgf/number format/.cd,
1331     1000 sep={}
1332 },

```

1333 }