Moloch

A clean and simple Beamer theme

Johan Larsson Matthias Vogelgesang

1 December 2025

Table of contents

1 Introduction

Beamer is a great 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 moloch is to provide a simple, modern Beamer theme suitable for anyone to use. 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.

moloch's codebase is maintained at https://github.com/jolars/moloch. If you have any issues, find mistakes in the manual or want to help make the theme even better, please get in touch there.

moloch is a fork of the popular Metropolis theme by Matthias Vogelgesang. The motivation for the fork was to fix some longstanding bugs in Metropolis and also simplify the codebase to make it easier to maintain and less fragile to changes in the underlying Beamer code.

2 Getting Started

2.1 Installing from CTAN

For most users, we recommend installing moloch from CTAN. If you keep your TeX distribution up-to-date, chances are good that moloch is already installed. If it is not, you need to update your packages. If your distribution is TeX Live (or MacTeX on OS X), the following command updates all packages.

```
tlmgr update --all
```

If this results in an error, you may need to run it with administrative privileges:

```
sudo tlmgr update --all
```

MacTeX on OS X also provides a graphical interface for `tlmgr` called TeX Live Utility.

For any other distribution please refer to its documentation on how to update your packages.

2.2 Installing from Source

If you want to use the development version of moloch, you can install it manually. You only need a recent LaTeX distribution which includes **l3build**. Then simply follow the steps below.

Download the source with a `git clone` of https://github.com/jolars/moloch

Install the package by running

```
l3build install
```

inside the downloaded directory.

2.3 A Minimal Example

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

```
\documentclass{beamer}
\usetheme{moloch}

\title{A Minimal Example}
```

```
\date{\today}
\author{Johan Larsson}
\institute{Some University}

\begin{document}
    \maketitle

    \section{First Section}

    \begin{frame}
        \frametitle{First Frame}

        Hello, world!

    \end{frame}
\end{document}
```

2.4 Dependencies

moloch depends on the `beamer` class and the following standard packages:

- tikz
- pgfopts
- etoolbox
- calc

2.5 Pandoc

To use this theme with Pandoc-based presentations, you can run the following command in your terminal:

```
pandoc -t beamer -V theme:moloch -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 Moloch in the preamble:

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

Options can be changed at any time—even mid-presentation—with the `molochset()` macro.

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

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

> **Options:** [list of possible values]{style="color: #2a7f62"} | **Default:** 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. Note that these commands do not affect math and numbers, so may not work as you expect if your titles contain these.

Options: regular, smallcaps, allsmallcaps, allcaps | **Default:** regular
Changes the format of "standout" frames (see `titleformat`, above).

3.1.2 Inner Theme

Options: none, simple, progressbar | **Default:** 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.

Options: none, simple, progressbar | **Default:** 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.

Options: none, hide, show | **Default:** none
This option decides whether or not to count standout pages as frames if frame counting. Option `none` (the default) means that the standout frames are not counted. `hide` means that they are counted but that there won't be any footer showing a frame number. `show` means that they are counted and that the frame number count is shown in the same fashion as for regular frames.

3.1.3 Outer Theme

Options: none, counter, fraction | **Default:** (none specified)
This option is deprecated and will be removed in a future version. Please use Beamer's `page number in head/foot` template instead. 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`).

Options: none, head, frametitle, foot | **Default:** 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

Options: transparent, fill | **Default:** transparent
Optionally adds a light grey background to block environments like `theorem` and `example`.

Options: dark, light | **Default:** light
Provides the option to have a dark background and light foreground instead of the reverse.

3.1.5 Font Theme

Options: regular, smallcaps, allsmallcaps, allcaps | **Default:** regular
Individually controls the format of titles, subtitles, section titles, and frame titles (see `titleformat`, above).

3.2 Color Customization

The included moloch 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 moloch 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}{ ... }
```

3.2.1 Themes

For low-light situations moloch it might be helpful to use the `moloch-highcontrast` color theme. It is enabled like any other color theme:

```
\usecolortheme{moloch-highcontrast}
```

There is also a theme based on the https://github.com/chriskempson/tomorrow-theme, which you can enable like this:

```
\usecolortheme{moloch-tomorrow}
```

3.3 Commands

3.3.1 Standout Frames

The moloch 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 Known Issues

4.1 Title Formats

Be aware that not every font supports small caps, so the `smallcaps` or `allsmallcaps` options may not work for all fonts. In particular, the Computer Modern sans-serif typeface, which is used by default when moloch is compiled with pdfLaTeX, does not have a small-caps variant.

Note that title format options `allsmallcaps` and `allcaps` do not affect the sizes of numerals, punctuation, and math symbol, and are probably best avoided if your titles contain these characters.

4.2 Interactions with Other Color Themes

moloch 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 moloch subpackages individually so the moloch color theme is never loaded. This will prevent conflicts between the moloch color theme and your preferred theme.

For example, overriding the color theme as follows may not work as expected because ‘`loads the [moloch]{.sans-serif} 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{moloch}
\usecolortheme{seahorse}
```

The correct colors are chosen if the moloch outer, inner, and font themes are loaded seperately:

```
\useoutertheme{moloch}
\useinnertheme{moloch}
\usefonttheme{moloch}
\usecolortheme{seahorse}   % or your preferred color theme
```

Please note that moloch may not use all the colors defined in your favourite Beamer color theme. In particular, moloch 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.

4.3 Notes on Second Screen

If you use the `[show notes on second screen]` option built in to Beamer and compile with , 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 itself. You can work around it either by compiling with or by adding the following code to your preamble to reset the text color on each slide.

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

4.4 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:
```

4.5 Standout Frames with Pandoc

With Pandoc versions prior to 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`.

4.6 License

moloch 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.

4.7 Implementation

Part I

Implementation

A Main Theme

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

A.0.1 Package Dependencies

```
\RequirePackage{pgfopts}
```

A.0.2 Options

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

```
\pgfkeys{/moloch/.cd,
  .search also={
      /moloch/inner,
      /moloch/outer,
      /moloch/color,
      /moloch/font,
    }
}
```

titleformat plain Controls the formatting of the text on standout "plain" frames.

```
\pgfkeys{
  /moloch/titleformat plain/.cd,
  .is choice,
  regular/.code={%
      \let\moloch@plaintitleformat\@empty%
      \setbeamerfont{standout}{shape=\normalfont}%
    },
  smallcaps/.code={%
      \let\moloch@plaintitleformat\@empty%
      \setbeamerfont{standout}{shape=\scshape}%
    },
  allsmallcaps/.code={%
      \let\moloch@plaintitleformat\MakeLowercase%
```

```
      \setbeamerfont{standout}{shape=\scshape}%
    },
  allcaps/.code={%
      \let\moloch@plaintitleformat\MakeUppercase%
      \setbeamerfont{standout}{shape=\normalfont}%
    },
}
```

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

```
\pgfkeys{
  /moloch/titleformat/.code=\pgfkeysalso{
    font/titleformat title=#1,
    font/titleformat subtitle=#1,
    font/titleformat section=#1,
    font/titleformat frame=#1,
    titleformat plain=#1,
  }
}
```

Set default values for options.

```
\newcommand{\moloch@setdefaults}{
  \pgfkeys{/moloch/.cd,
    titleformat plain=regular,
  }
}
```

A.0.3 Component Sub-Packages

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

```
\useinnertheme{moloch}
\useoutertheme{moloch}
\usecolortheme{moloch}
\usefonttheme{moloch}
```

A.0.4 Custom Commands

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

Allows the user to change options midway through a presentation.

```
\newcommand{\molochset}[1]{\pgfkeys{/moloch/.cd,#1}}
```

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

A.0.5 Process Package Options

```
\moloch@setdefaults
\ProcessPgfOptions{/moloch}
```

B 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.

B.0.1 Package Dependencies

```
\RequirePackage{keyval}
\RequirePackage{calc}
\RequirePackage{pgfopts}
\RequirePackage{tikz}
```

B.0.2 Memoization and Tikz Externalization

See the documentation for the correspondign section under the outer theme for more information on the following lines.

```
\providecommand{\tikzexternalenable}{}
\providecommand{\tikzexternaldisable}{}
\providecommand{\mmzUnmemoizable}{}
```

B.0.3 Options

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

```
\pgfkeys{
  /moloch/inner/sectionpage/.cd,
  .is choice,
  none/.code=\moloch@disablesectionpage,
  simple/.code={%
```

```
      \moloch@enablesectionpage%
      \setbeamertemplate{section page}[simple]%
    },
  progressbar/.code={%
      \moloch@enablesectionpage%
      \setbeamertemplate{section page}[progressbar]%
    },
}
```

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

```
\pgfkeys{
  /moloch/inner/subsectionpage/.cd,
  .is choice,
  none/.code=\moloch@disablesubsectionpage,
  simple/.code={%
      \moloch@enablesubsectionpage%
      \setbeamertemplate{section page}[simple]%
    },
  progressbar/.code={%
      \moloch@enablesubsectionpage%
      \setbeamertemplate{section page}[progressbar]%
    },
}
```

standoutnumbering Whether or not to number standout pages. Option `none` means that standout pages are not numbered (do not count as frames). `hide` means that they do count as frames, but that the footer with the number is not shown. Option `show` means that they both count as frames and that the footer with a frame count is shown.

```
\providebool{moloch@enableStandoutFooter}
\providebool{moloch@enableStandoutNumbering}
\pgfkeys{
  /moloch/inner/standoutnumbering/.cd,
  .is choice,
  none/.code={
      \boolfalse{moloch@enableStandoutNumbering}
      \boolfalse{moloch@enableStandoutFooter}
    },
  show/.code={
      \booltrue{moloch@enableStandoutNumbering}
      \booltrue{moloch@enableStandoutFooter}
    },
  hide/.code={
      \booltrue{moloch@enableStandoutNumbering}
```

```
      \boolfalse{moloch@enableStandoutFooter}
    }
}
```

titleseparator linewidth Set the width of the line separating the title from the author.

```
\newlength{\moloch@titleseparator@linewidth}
\pgfkeys{
  /moloch/inner/.cd,
  titleseparatorlinewidth/.code={\setlength{\moloch@titleseparator@linewidth}{#1}},
  titleseparatorlinewidth/.default=0.4pt,
}
```

titleseparator aliases Allows `titleseparator linewidth` to be used in `\usetheme`.

```
\pgfkeys{
  /moloch/inner/.cd,
  titleseparator linewidth/.code=\pgfkeysalso{titleseparatorlinewidth=#1},
}
```

@inner@setdefaults Set default values for inner theme options.

```
\newcommand{\moloch@inner@setdefaults}{
  \pgfkeys{/moloch/inner/.cd,
    sectionpage=progressbar,
    subsectionpage=none,
    standoutnumbering=none,
    titleseparator linewidth=0.4pt,
  }
}
```

B.0.4 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.

```
\setbeamertemplate{title page}{
  \null%
  \vspace{0pt plus 1.618fil}%
  \vfil%
  \ifx\inserttitlegraphic\@empty\else\usebeamertemplate*{title graphic}\fi
  \ifx\inserttitle\@empty\else\usebeamertemplate*{title}\fi
  \ifx\insertsubtitle\@empty\else\usebeamertemplate*{subtitle}\fi
  \usebeamertemplate*{title separator}
  \expandafter\ifblank\expandafter{\beamer@andstripped}{}{%
    \usebeamertemplate*{author}%
```

```
  }
  \ifx\insertinstitute\@empty\else\usebeamertemplate*{institute}\fi
  \ifx\insertdate\@empty\else\usebeamertemplate*{date}\fi
  \vspace{0pt plus 1fil}%
  \null
}
```

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 and ensure the title frame number doesn't count.

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

```
\def\maketitle{%
  \ifbeamer@inframe
    \titlepage
  \else
    \begingroup
    \renewcommand\footnoterule{}%
    \frame[plain,noframenumbering]{\titlepage}
    \endgroup
  \fi
}
\def\titlepage{%
  % Apply title-page specific footnote settings
  \renewcommand{\@makefntext}[1]{%
    {\par\usebeamercolor[fg]{thanks}\usebeamerfont{thanks}$^{\@thefnmark}$##1\medskip}%
  }

  % Process the title page
  \usebeamertemplate{title page}\@thanks
}
```

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

```
\setbeamertemplate{title graphic}{
  \inserttitlegraphic%
  \par%
  \vspace*{1em}
}
```

`title` Set the title on the title page.

```
\setbeamertemplate{title}{
  \raggedright%
```

```
  \moloch@titleformat{\inserttitle}%
  \par%
}
```

subtitle Set the subtitle on the title page.

```
\setbeamertemplate{subtitle}{
  \vspace*{0.3em}
  \raggedright%
  \moloch@subtitleformat{\insertsubtitle}%
  \par%
}
```

title separator Template to set the title separator.

```
\setbeamertemplate{title separator}{
  \tikzexternaldisable%
  \begin{tikzpicture}[baseline=(current bounding box.north)]
    \mmzUnmemoizable%
    \fill[fg] (0,0) rectangle (\textwidth, \moloch@titleseparator@linewidth);
    \useasboundingbox (0,0) rectangle (\textwidth,-\moloch@titleseparator@linewidth);
  \end{tikzpicture}%
  \tikzexternalenable%
  \par%
  \vspace*{0.8em}
}
```

author Set the author on the title page.

```
\setbeamertemplate{author}{
  \raggedright%
  \insertauthor%
  \par%
  \vspace*{0.5em}
}
```

institute Set the institute on the title page.

```
\setbeamertemplate{institute}{
  \insertinstitute%
  \par%
  \vspace*{1em}
}
```

date Set the date on the title page.

```
\setbeamertemplate{date}{
  \insertdate%
```

```
  \par%
}
```

B.0.5 Section Page

section Page

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

```
\defbeamertemplate{section page}{simple}{
  \begin{center}
    \usebeamercolor[fg]{section title}
    \usebeamerfont{section title}
    \moloch@sectiontitleformat{\insertsectionhead}\par
    \usebeamercolor[fg]{subsection title}%
    \usebeamerfont{subsection title}%
    \strut%
    \ifx\insertsubsectionhead\@empty\else%
      \insertsubsectionhead%
    \fi
  \end{center}
  \vspace{\baselineskip - 1ex + \moloch@titleseparator@linewidth}
}
\defbeamertemplate{section page}{progressbar}{
  \centering
  \begin{minipage}{0.7875\linewidth}
    \raggedright
    \usebeamercolor[fg]{section title}
    \usebeamerfont{section title}
    \moloch@sectiontitleformat{\insertsectionhead}\\[-0.5\baselineskip]
    \usebeamertemplate*{progress bar in section page}
    \par
    \usebeamercolor[fg]{subsection title}%
    \usebeamerfont{subsection title}%
    \strut%
    \ifx\insertsubsectionhead\@empty\else%
      \insertsubsectionhead%
    \fi
  \end{minipage}
  \par
}
\newcommand{\moloch@disablesectionpage}{
  \AtBeginSection{
```

```
    % intentionally empty
  }
}
\newcommand{\moloch@enablesectionpage}{
  \AtBeginSection{
    \ifbeamer@inframe
      \sectionpage
    \else
      \frame[plain,c,noframenumbering]{\sectionpage}
    \fi
  }
}
```

subsection page

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

```
\setbeamertemplate{subsection page}{%
  \usebeamertemplate*{section page}
}
\newcommand{\moloch@disablesubsectionpage}{
  \AtBeginSubsection{
    % intentionally empty
  }
}
\newcommand{\moloch@enablesubsectionpage}{
  \AtBeginSubsection{
    \ifbeamer@inframe
      \subsectionpage
    \else
      \frame[plain,c,noframenumbering]{\subsectionpage}
    \fi
  }
}
```

progress bar in section page

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

```
\setbeamertemplate{progress bar in section page}{
  \pgfmathsetlength{\moloch@progressonsectionpage}{
    \textwidth * min(1,\insertframenumber/\inserttotalframenumber)
  }%
  \tikzexternaldisable%
```

```
  \begin{tikzpicture}[baseline=(current bounding box.north)]
    \mmzUnmemoizable%
    \fill[bg]
    (0,0)
    rectangle
    (\textwidth, \moloch@progressonsectionpage@linewidth);
    \fill[fg]
    (0,0)
    rectangle
    (\moloch@progressonsectionpage,
    \moloch@progressonsectionpage@linewidth);
    \useasboundingbox (0,0) rectangle (\textwidth,-\moloch@progressonsectionpage@linewidth);
  \end{tikzpicture}%
  \tikzexternalenable%
}
```

The code above 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 `\moloch@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.

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

B.0.6 Lists and Floats

Moloch uses custom symbols for the `itemize` environment. The symbols are defined as below, using `pgf` commands to draw the shapes. This is slightly different than what beamer does by default, which is to use font glyphs from the math font. But we want to avoid this reliance on the math font, which may have somewhat surprising side effects.

By default, we use a filled circle for the first-level `itemize` items, a filled square for the second level, and a filled triangle for the third level. Since the triangle tapers to a point, we add a slight overhang to it so that it visually aligns better with the other symbols. We do the same for the circle, but to a lower extent.

```
\newcommand{\mitemover}[2]{\makebox[0pt][r]{#1}\kern-#2}
\newcommand{\mitem}[1]{\mitemover{#1}{0pt}}

\newcommand{\overhangSquare}{0pt}
\newcommand{\overhangCircle}{0.05ex}
\newcommand{\overhangTriangle}{0.25ex}
```

```
\newcommand{\moloch@circle}{
  \begin{pgfpicture}
    \pgfsetbaseline{-0.7ex}
    \pgfpathcircle{\pgfpoint{0}{0}}{0.16em}
    \pgfusepath{fill}
  \end{pgfpicture}%
}
%
\newcommand{\moloch@square}{
  \begin{pgfpicture}
    \pgfsetbaseline{-0.7ex}
    \pgfpathrectangle{\pgfpoint{-0.165em}{-0.165em}}{\pgfpoint{0.31em}{0.31em}}
    \pgfusepath{fill}
  \end{pgfpicture}%
}
%
\newcommand{\moloch@triangle}{
  \begin{pgfpicture}
    \pgfsetbaseline{-0.7ex}
    \pgfpathmoveto{\pgfpoint{0.21em}{0}}       % right vertex (tip)
    \pgfpathlineto{\pgfpoint{-0.21em}{0.21em}}  % top left
    \pgfpathlineto{\pgfpoint{-0.21em}{-0.21em}} % bottom left
    \pgfpathclose
    \pgfusepath{fill}
  \end{pgfpicture}%
}
```

Next, we set the itemize templates to use these symbols.

```
% \setbeamertemplate{itemize item}[circle]
\setbeamertemplate{itemize item}{\mitemover{\moloch@circle}{\overhangCircle}}
\setbeamertemplate{itemize subitem}{\mitemover{\moloch@square}{\overhangSquare}}
\setbeamertemplate{itemize subsubitem}{\mitemover{\moloch@triangle}{\overhangTriangle}}
\setbeamertemplate{caption label separator}{: }
\setbeamertemplate{caption}[numbered]
```

B.0.7 Footnotes

```
\setbeamertemplate{footnote}{%
  \parindent 0em\noindent\raggedright \usebeamercolor{footnote}\hbox to
  0.8em{\hfil\insertfootnotemark}\insertfootnotetext%%%
  \par%
```

```
}
```

B.0.8 Text and Spacing Settings

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.

```
\define@key{beamerframe}{c}[true]{% centered
  \beamer@frametopskip=0pt plus 1fil\relax%
  \beamer@framebottomskip=0pt plus 1fil\relax%
  \beamer@frametopskipautobreak=0pt plus .4\paperheight\relax%
  \beamer@framebottomskipautobreak=0pt plus .6\paperheight\relax%
  \def\beamer@initfirstlineunskip{}%
}
```

B.0.9 Standout Frames

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 set frame options.

```
\providebool{moloch@standout}
\define@key{beamerframe}{standout}[true]{%
  \booltrue{moloch@standout}
  \begingroup
  \setkeys{beamerframe}{c}
  \ifbool{moloch@enableStandoutNumbering}{}{%
    \setkeys{beamerframe}{noframenumbering}}
  \ifbeamercolorempty[bg]{palette primary}{
    \setbeamercolor{background canvas}{
      use=palette primary,
      bg=-palette primary.fg
    }
  }{
    \setbeamercolor{background canvas}{
      use=palette primary,
```

```
      bg=palette primary.bg
    }
  }
  \setbeamercolor{local structure}{
    fg=palette primary.fg
  }
  \usebeamercolor[fg]{palette primary}
  \setbeamercolor{page number in head/foot}{
    use=palette primary,
    fg=palette primary.fg
  }
  \ifbool{moloch@enableStandoutFooter}{}{\setbeamertemplate{footline}{}}
}
```

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 for this (see http://tex.stackexchange.com/questions/226319/). Instead, we prepend the `\endgroup` to `\beamer@reseteecodes`, which is run exactly once at the end of each slide.

```
\pretocmd{\beamer@reseteecodes}{%
  \ifbool{moloch@standout}{
    \endgroup
    \boolfalse{moloch@standout}
  }{}
}{}{}
```

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.

```
\AtBeginEnvironment{beamer@frameslide}{
  \ifbool{moloch@standout}{
    \centering
    \usebeamerfont{standout}
  }{}
}
```

B.0.10 Process Package Options

```
\moloch@inner@setdefaults
\ProcessPgfPackageOptions{/moloch/inner}
```

C 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.

C.0.1 Package dependencies

```
\RequirePackage{calc}
\RequirePackage{pgfopts}
```

C.0.2 Memoization and Tikz Externalization

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. We do the same for the "mmzUnmemoizable" command from the memoize package, in order to avoid memoization of the progress bars.

```
\providecommand{\tikzexternalenable}{}
\providecommand{\tikzexternaldisable}{}
\providecommand{\mmzUnmemoizable}{}
```

C.0.3 Options

`progressbar` Adds a progress bar to the top, bottom, or frametitle of each slide.

```
\pgfkeys{
  /moloch/outer/progressbar/.cd,
  .is choice,
  none/.code={%
      \setbeamertemplate{headline}[plain]
      \setbeamertemplate{frametitle}[plain]
      \setbeamertemplate{footline}[plain]
    },
  head/.code={\pgfkeys{/moloch/outer/progressbar=none}
      \addtobeamertemplate{headline}{}{%
        \usebeamertemplate*{progress bar in head/foot}
```

```
      }
    },
  frametitle/.code={\pgfkeys{/moloch/outer/progressbar=none}
      \addtobeamertemplate{frametitle}{}{%
        \usebeamertemplate*{progress bar in head/foot}
      }
    },
  foot/.code={\pgfkeys{/moloch/outer/progressbar=none}
      \addtobeamertemplate{footline}{}{%
        \usebeamertemplate*{progress bar in head/foot}%
      }
    },
}
```

progressbar linewidth Sets the linewidth of the progress bar for sectionpages and frames.

```
\newlength{\moloch@progressonsectionpage}
\newlength{\moloch@progressonsectionpage@linewidth}
\newlength{\moloch@progressinheadfoot}
\newlength{\moloch@progressinheadfoot@linewidth}
\pgfkeys{
  /moloch/outer/.cd,
  progressbarlinewidth/.code={
      \setlength{\moloch@progressonsectionpage@linewidth}{#1}
      \setlength{\moloch@progressinheadfoot@linewidth}{#1}
    },
  progressbarlinewidth/.default=0.4pt,
}
```

progressbar aliases Allows `progressbar linewidth` to be used in `\molochset`.

```
\pgfkeys{
  /moloch/outer/.cd,
  progressbar linewidth/.code=\pgfkeysalso{progressbarlinewidth=#1},
}
```

frametitle margin Sets the margins of the frame title.

```
\pgfkeys{
  /moloch/outer/.cd,
  frametitlemarginleft/.code=\renewcommand{\moloch@frametitle@margin@left}{#1},
  frametitlemarginright/.code=\renewcommand{\moloch@frametitle@margin@right}{#1},
  frametitlemargintop/.code=\renewcommand{\moloch@frametitle@margin@top}{#1},
  frametitlemarginbottom/.code=\renewcommand{\moloch@frametitle@margin@bottom}{#1},
}
```

C.0.4 Deprecated Options

These options are deprecated and will be removed in a future version.

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

```
\pgfkeys{
  /moloch/outer/numbering/.cd,
  .is choice,
  none/.code={%
      \PackageWarning{moloch}{The ``numbering'' option is deprecated.
        Use beamer's ``page number in head/foot'' template instead}%
      \setbeamertemplate{page number in head/foot}[default]
    },
  counter/.code={%
      \PackageWarning{moloch}{The ``numbering'' option is deprecated.
        Use beamer's ``page number in head/foot'' template instead}%
      \setbeamertemplate{page number in head/foot}[framenumber]
    },
  fraction/.code={%
      \PackageWarning{moloch}{The ``numbering'' option is deprecated.
        Use beamer's ``page number in head/foot'' template instead}%
      \setbeamertemplate{page number in head/foot}[totalframenumber]
    },
}
```

C.0.5 Slide Numbering

Moloch defaults to numbering frames. To modify this, simply copy this line to your preamble and replace `framenumber`.

```
\setbeamertemplate{page number in head/foot}[framenumber]
```

C.0.6 Head and footline

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

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

headline

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

```
\defbeamertemplate{headline}{plain}{}
\defbeamertemplate{footline}{plain}{%
```

```
  \begin{beamercolorbox}[
      leftskip=4pt,%
      rightskip=5pt,%
      wd=\textwidth,%
    ]{footline}%
    \usebeamercolor[fg]{page number in head/foot}%
    \usebeamerfont{page number in head/foot}%
    \usebeamertemplate*{frame footer}%
    \hfill%
    \usebeamertemplate*{page number in head/foot}\vskip4pt%
  \end{beamercolorbox}%
}
```

C.0.7 Frametitle

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

```
\newcommand{\moloch@frametitlestrut@start}{%
  \rule{0pt}{\moloch@frametitle@margin@top + \ht\strutbox}%
}%

\newcommand{\moloch@frametitlestrut@end}{%
  \rule[-\moloch@frametitle@margin@bottom]{0pt}{\moloch@frametitle@margin@bottom}%
}

\newcommand{\moloch@frametitle@margin@left}{1.6ex}
\newcommand{\moloch@frametitle@margin@right}{1.6ex}
\newcommand{\moloch@frametitle@margin@top}{1.4ex}
\newcommand{\moloch@frametitle@margin@bottom}{1.4ex}

\defbeamertemplate{frametitle}{plain}{%
  \nointerlineskip%
  \begin{beamercolorbox}[%
      wd=\paperwidth,%
      leftskip=\moloch@frametitle@margin@left,%
      rightskip=\the\glueexpr \moloch@frametitle@margin@right plus 1fil\relax,%
    ]{frametitle}%
    \usebeamerfont{frametitle}%
    \moloch@frametitlestrut@start%
    \moloch@frametitleformat{\insertframetitle}%
    {%
      \ifx\insertframesubtitle\@empty%
        \else%
```

```
        {%
          \par%
          \usebeamerfont{framesubtitle}%
          \vspace{-0.8ex}%
          \usebeamercolor[fg]{framesubtitle}%
          \insertframesubtitle%
        }%
      \fi
    }%
    \moloch@frametitlestrut@end%
  \end{beamercolorbox}%
}
\setbeamertemplate{frametitle continuation}{%
  \romannumeral\insertcontinuationcount}
```

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`.

```
\setbeamertemplate{progress bar in head/foot}{
  \nointerlineskip%
  \pgfmathsetlength{\moloch@progressinheadfoot}{%
    \paperwidth * min(1,\insertframenumber/\inserttotalframenumber)%
  }%
  \begin{beamercolorbox}[wd=\paperwidth]{progress bar in head/foot}
    \tikzexternaldisable%
    \begin{tikzpicture}
      \mmzUnmemoizable%
      \fill[bg]
      (0,0)
      rectangle
      (\paperwidth, \moloch@progressinheadfoot@linewidth);
      \fill[fg]
      (0,0)
      rectangle
      (\moloch@progressinheadfoot, \moloch@progressinheadfoot@linewidth);
    \end{tikzpicture}
    \tikzexternalenable%
  \end{beamercolorbox}
}
```

C.0.8 Process package options

```
\moloch@outer@setdefaults
\ProcessPgfPackageOptions{/moloch/outer}
```

D Font Theme

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

D.0.1 Package dependencies

```
\RequirePackage{etoolbox}
\RequirePackage{pgfopts}
```

D.0.2 General font definitions

```
\setbeamerfont{title}{size=\Large, series=\bfseries}
\setbeamerfont{author}{size=\small}
\setbeamerfont{date}{size=\small}
\setbeamerfont{section title}{size=\Large, series=\bfseries}
\setbeamerfont{block title}{size=\normalsize, series=\bfseries}
\setbeamerfont{block title alerted}{size=\normalsize, series=\bfseries}
\setbeamerfont*{subtitle}{size=\large}
\setbeamerfont{frametitle}{size=\large, series=\bfseries}
\setbeamerfont{framesubtitle}{size=\small}
\setbeamerfont{caption}{size=\small}
\setbeamerfont{caption name}{series=\bfseries}
\setbeamerfont{description item}{series=\bfseries}
\setbeamerfont{standout}{size=\Large, series=\bfseries}
```

D.0.3 Title format options

`titleformat title` Controls the format of the title.

```
\pgfkeys{
  /moloch/font/titleformat title/.cd,
  .is choice,
  regular/.code={%
      \let\moloch@titleformat\@empty%
      \setbeamerfont{title}{shape=\normalfont}%
    },
  smallcaps/.code={%
```

```
      \let\moloch@titleformat\@empty%
      \setbeamerfont{title}{shape=\scshape}%
    },
  allsmallcaps/.code={%
      \let\moloch@titleformat\lowercase%
      \setbeamerfont{title}{shape=\scshape}%
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat title=allsmallcaps can
        lead to problems}
    },
  allcaps/.code={%
      \let\moloch@titleformat\uppercase%
      \setbeamerfont{title}{shape=\normalfont}
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat title=allcaps can lead to problems%
      }
    },
}
```

titleformat subtitle Control the format of the subtitle.

```
\pgfkeys{
  /moloch/font/titleformat subtitle/.cd,
  .is choice,
  regular/.code={%
      \let\moloch@subtitleformat\@empty%
      \setbeamerfont{subtitle}{shape=\normalfont}%
    },
  smallcaps/.code={%
      \let\moloch@subtitleformat\@empty%
      \setbeamerfont{subtitle}{shape=\scshape}%
    },
  allsmallcaps/.code={%
      \let\moloch@subtitleformat\MakeLowercase%
      \setbeamerfont{subtitle}{shape=\scshape}%
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat subtitle=allsmallcaps
        can lead to problems}
    },
  allcaps/.code={%
      \let\moloch@subtitleformat\MakeUppercase%
      \setbeamerfont{subtitle}{shape=\normalfont}%
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat subtitle=allcaps can
```

```
        lead to problems}
    },
}
```

titleformat section Controls the format of the section title.

```
\pgfkeys{
  /moloch/font/titleformat section/.cd,
  .is choice,
  regular/.code={%
      \let\moloch@sectiontitleformat\@empty%
      \setbeamerfont{section title}{shape=\normalfont}%
    },
  smallcaps/.code={%
      \let\moloch@sectiontitleformat\@empty%
      \setbeamerfont{section title}{shape=\scshape}%
    },
  allsmallcaps/.code={%
      \let\moloch@sectiontitleformat\MakeLowercase%
      \setbeamerfont{section title}{shape=\scshape}%
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat section=allsmallcaps
        can lead to problems}
    },
  allcaps/.code={%
      \let\moloch@sectiontitleformat\MakeUppercase%
      \setbeamerfont{section title}{shape=\normalfont}%
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat section=allcaps
        can lead to problems}
    },
}
```

frametitleformat Control the format of the frame title.

```
\pgfkeys{
  /moloch/font/titleformat frame/.cd,
  .is choice,
  regular/.code={%
      \let\moloch@frametitleformat\@empty%
      \setbeamerfont{frametitle}{shape=\normalfont}%
    },
  smallcaps/.code={%
      \let\moloch@frametitleformat\@empty%
      \setbeamerfont{frametitle}{shape=\scshape}%
```

```
    },
  allsmallcaps/.code={%
      \let\moloch@frametitleformat\MakeLowercase%
      \setbeamerfont{frametitle}{shape=\scshape}%
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat frame=allsmallcaps
        can lead to problems}
    },
  allcaps/.code={%
      \let\moloch@frametitleformat\MakeUppercase%
      \setbeamerfont{frametitle}{shape=\normalfont}
      \PackageNote{beamerthememoloch}{%
        Be aware that titleformat frame=allcaps can lead to problems%
      }
    },
}
```

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

```
\pgfkeys{
  /moloch/font/.cd,
  titleformattitle/.code=\pgfkeysalso{titleformat title=#1},
  titleformatsubtitle/.code=\pgfkeysalso{titleformat subtitle=#1},
  titleformatsection/.code=\pgfkeysalso{titleformat section=#1},
  titleformatframe/.code=\pgfkeysalso{titleformat frame=#1},
}
```

@font@setdefaults Sets default values for font theme options.

```
\newcommand{\moloch@font@setdefaults}{
  \pgfkeys{/moloch/font/.cd,
    titleformat title=regular,
    titleformat subtitle=regular,
    titleformat section=regular,
    titleformat frame=regular,
  }
}
```

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

```
\def\moloch@titleformat#1{#1}
\def\moloch@subtitleformat#1{#1}
\def\moloch@sectiontitleformat#1{#1}
\def\moloch@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.

D.0.4 Process package options

```
\moloch@font@setdefaults
\ProcessPgfPackageOptions{/moloch/font}
```

E Color Theme

E.0.1 Package Dependencies

```
\RequirePackage{pgfopts}
```

E.0.2 Options

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

```
\pgfkeys{
  /moloch/color/block/.cd,
  .is choice,
  transparent/.code=\moloch@block@transparent,
  fill/.code=\moloch@block@fill,
}
```

colors Provides the option to have a dark background and light foreground instead of the reverse.

```
\pgfkeys{
  /moloch/color/background/.cd,
  .is choice,
  dark/.code=\moloch@colors@dark,
  light/.code=\moloch@colors@light,
}
```

@color@setdefaults Sets default values for color theme options.

```
\newcommand{\moloch@color@setdefaults}{
  \pgfkeys{/moloch/color/.cd,
    background=light,
  }
}
```

E.0.3 Base Colors

```
\definecolor{mDarkBrown}{HTML}{604c38}
\definecolor{mDarkTeal}{HTML}{23373b}
```

```
\definecolor{mLightBrown}{HTML}{EB811B}
\definecolor{mLightGreen}{RGB}{0,128,128}
```

E.0.4 Base Styles

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

```
\setbeamercolor{alerted text}{%
  fg=mLightBrown
}
\setbeamercolor{example text}{%
  fg=mLightGreen
}
```

E.0.5 Hooks for Color Themes

Moloch color themes can register light and dark color schemes using the commands below. The registered colors will be stored in the macros `\moloch@define@light@colors` and `\moloch@define@dark@colors` respectively. These macros are invoked when the `background=light` or `background=dark` options are selected.

```
\newcommand{\moloch@define@light@colors}{}
\newcommand{\moloch@define@dark@colors}{}

\newcommand{\moloch@register@light@colors}[1]{%
  \renewcommand{\moloch@define@light@colors}{#1}%
}

\newcommand{\moloch@register@dark@colors}[1]{%
  \renewcommand{\moloch@define@dark@colors}{#1}%
}

\newcommand{\moloch@colors@dark}{
  \moloch@define@dark@colors
  \usebeamercolor[fg]{normal text}
}
\newcommand{\moloch@colors@light}{
  \moloch@define@light@colors
  \usebeamercolor[fg]{normal text}
}

\moloch@register@light@colors{%
```

```
  \setbeamercolor{normal text}{%
    fg=mDarkTeal,
    bg=black!2
  }%
}

\moloch@register@dark@colors{%
  \setbeamercolor{normal text}{%
    fg=black!2,
    bg=mDarkTeal
  }%
}
```

E.0.6 Derived Colors

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

```
\setbeamercolor{titlelike}{use=normal text, parent=normal text}
\setbeamercolor{author}{use=normal text, parent=normal text}
\setbeamercolor{date}{use=normal text, parent=normal text}
\setbeamercolor{institute}{%
  use=normal text, fg=normal text.fg!80!normal text.bg}
\setbeamercolor{structure}{use=normal text, fg=normal text.fg}
\setbeamercolor{thanks}{%
  use=normal text,fg=normal text.fg!80!normal text.bg}
```

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

```
\setbeamercolor{palette primary}{%
  use=normal text,
  fg=normal text.bg,
  bg=normal text.fg
}
\setbeamercolor{frametitle}{%
  use=palette primary,
  parent=palette primary
}
```

The 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`.

```
\setbeamercolor{progress bar}{%
  use=alerted text,
  fg=alerted text.fg,
  bg=alerted text.fg!50!black!30
}
\setbeamercolor{title separator}{
  use=progress bar,
  parent=progress bar
}
\setbeamercolor{progress bar in head/foot}{%
  use=progress bar,
  parent=progress bar
}
\setbeamercolor{progress bar in section page}{
  use=progress bar,
  parent=progress bar
}
```

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.

```
\newcommand{\moloch@block@transparent}{
  \setbeamercolor{block title}{bg=}
  \setbeamercolor{block body}{bg=}
  \setbeamercolor{block title alerted}{bg=}
  \setbeamercolor{block title example}{bg=}
}
\newcommand{\moloch@block@fill}{
  \setbeamercolor{block title}{%
    bg=normal text.bg!80!fg
  }
  \setbeamercolor{block body}{%
    use=block title,
    bg=block title.bg!50!normal text.bg
  }
  \setbeamercolor{block title alerted}{%
    bg=block title.bg,
  }
  \setbeamercolor{block title example}{%
```

```
    bg=block title.bg,
  }
}
\setbeamercolor{block title}{%
  use=normal text,
  fg=normal text.fg
}
\setbeamercolor{block title alerted}{%
  use={block title, alerted text},
  fg=alerted text.fg
}
\setbeamercolor{block title example}{%
  use={block title, example text},
  fg=example text.fg
}
\setbeamercolor{block body alerted}{use=block body, parent=block body}
\setbeamercolor{block body example}{use=block body, parent=block body}
```

Footnotes

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

```
\setbeamercolor{bibliography entry author}{fg=, bg=}
\setbeamercolor{bibliography entry title}{fg=, bg=}
\setbeamercolor{bibliography entry location}{fg=, bg=}
\setbeamercolor{bibliography entry note}{fg=, bg=}
```

E.0.7 Process Package Options

```
\moloch@color@setdefaults
\ProcessPgfPackageOptions{/moloch/color}
```

```
\mode<all>
```

F Color Theme: Tomorrow

Register tomorrow-specific light and dark color schemes

```
\mode<all>
```

G Color Theme: High Contrast

Register high-contrast-specific light and dark color schemes

```
\mode<all>
```