Trigon
A Modern, elegant and versatile theme for Beamer

Thomas Lambert
trigon@thl.ovh

v0.6.2 — 2021/11/10

Contents

1 Introduction 2
2 Getting Started 2
  2.1 Installing from CTAN ........................................ 2
  2.2 Installing from Gitlab ...................................... 3
  2.3 Overleaf ..................................................... 3
  2.4 A Minimal Example ......................................... 4
  2.5 Dependencies ................................................. 4
3 Customization 5
  3.1 Package options ............................................. 5
  3.2 Color Customization ......................................... 7
4 Tips & Tricks 8
  4.1 Backup Slides ............................................... 8
5 Known Issues 8
  5.1 Title formats ............................................... 8
6 License 9
7 Implementation 9
  7.1 Trigon parent theme ....................................... 9
1 Introduction

Trigon found its origin and inspiration in the graphical guidelines resulting from the visual identity overhaul of the University of Liège (Belgium). Although directly inspired from these guidelines, Trigon was stripped out of any mention or specificities related the University and its faculties. This makes the Trigon theme perfectly suitable for many different contexts.

The final product provides a modern, elegant and versatile theme with a high degree of customization.

Trigon’s codebase lives on Gitlab. The implementation of this theme is strongly inspired from the excellent Metropolis theme by Matthias Vogelgesang, from which it borrows numerous options.

Any feature request, issue report or merge requests are welcome.

2 Getting Started

2.1 Installing from CTAN

The latest stable version of Trigon is available on CTAN and should now be part of the usual \TeX{} distributions (\TeX{} Live, Mac\TeX{}, Mik\TeX{}), under the name beamertheme-trigon. It means that if your distribution is kept up-to-date, the package should normally be already installed on your system. If this is not the case, consider updating the packages of your \TeX{} distribution.

For \TeX{} Live and Mac\TeX{} users, this usually means running

\texttt{tlmgr update --all}

, or if administrative privileges are required

\texttt{sudo tlmgr update --all}
For MikTeX users, please refer to the official MikTeX documentation.

2.2 Installing from Gitlab

If you want to use the cutting-edge development version of Trigon, you can install it manually by following these steps:

Download the source from Trigon repository using `git clone` or as a zip archive of the latest development version.

Compile the style files by running `make sty` inside the downloaded directory. (Or run `pdflatex` directly on `source/trigontheme.ins`.)

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

Use the theme by declaring `\usetheme{trigon}` in the preamble of your document.

Trigon 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 Trigon.
- `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.

2.3 Overleaf

Since October 2021, TeXLive 2011 is available on Overleaf. Trigon is therefore automatically supported for new documents. If you want to switch older documents to use the Trigon theme, you need to select TeXLive 2021 or higher as your TeX Live version (this may cause issues with other packages used in your presentation).
In addition to that, a template project was created and is publicly available in the Overleaf Gallery to bootstrap your presentation.

2.4 A Minimal Example

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

```latex
\documentclass{beamer}
\usetheme{trigon} % Use trigon theme
\title{A minimal example}
\subtitle{A subtitle example}
\date{\today}
\author{Thomas Lambert}
\institute{Your university or company}
\begin{document}
  \maketitle
  \section{First Section}
  \begin{frame}{First Frame}
    Hello, world!
  \end{frame}
\end{document}
```

2.5 Dependencies

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

- `tikz`
- `pgfplots`

The theme works best with the open source Source Sans Pro font from Adobe. Therefore, the package `sourcesanspro` is loaded by default with the theme. However, if the user prefers to select an other font, the theme option `nosourcefonts` can be used to prevent the font package from being loaded.
3 Customization

3.1 Package options

Trigon 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 Trigon in the preamble:

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

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

\trigonset{option1=newvalue1, option2=newvalue2, ...}

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

<table>
<thead>
<tr>
<th>option key</th>
<th>list of possible values</th>
<th>default</th>
<th>A short description of the option.</th>
</tr>
</thead>
<tbody>
<tr>
<td>background</td>
<td>light,dark</td>
<td>light</td>
<td>Provides the option to have a dark background and light foreground instead of the reverse.</td>
</tr>
<tr>
<td>titleformat</td>
<td>regular, smallcaps, allsmallcaps, allcaps</td>
<td>regular</td>
<td>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 5.1 for known issues with these options.</td>
</tr>
<tr>
<td>usesourcefonts</td>
<td>true,false</td>
<td>true</td>
<td>Chooses if the sourcesanspro font package should be loaded true or not false.</td>
</tr>
</tbody>
</table>

As Trigon implementation is heavily inspired from the excellent Metropolis theme by Matthias Vogelgesang, many of Metropolis options are also available in Trigon.

3.1.1 Main theme
3.1.2 Inner theme

sectionpage none, simple

Adds a slide at the start of each section (simple). The none option disables the section page.

titlestyle plain, style1, style2

Changes the layout of the title page.

sectionstyle plain, style1, style2, style3

Changes the layout of the section page. style3 is similar to plain but with the right triangle in grey instead of theme color.

slidestyle blank, style1, cyber

Changes the background of the regular frames.
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).

3.1.4 Color theme

**block**

transparent, fill .......................... fill

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

**headingcolor**

default, theme .......................... default

Controls whether the color of all headings (title page, frame title, etc.) should be in black (default) or in a slightly darker shade of the theme color *theme*.

**textgcolor**

default, theme .......................... default

Controls whether the color of the regular text elements should be in black (default) or in a slightly darker shade of the theme color *theme*.

3.1.5 Font theme

**titleformat**

title

regular, smallcaps, allsmallcaps, allcaps .......................... regular

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

3.2 Color Customization

The included Trigon 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 a few main colors:

- **tPrim** The primary theme color, used for main triangle elements, and for headings and text if the appropriate options are selected.

- **tSec** The secondary theme color (lighter shade of the primary color), used for some triangle elements and example text.
• **tAccent** The accent color, used mostly for alerted text.

• **tGrey** A grey color, used for background elements (triangles, blocks, etc.).

• **tTxt** The default text color.

• **tBg** The default background color.

• **tGreyBg** A tint of grey to use for the triangle elements.

• **tTheme** A tint of the *tPrim* color to use for some triangle elements.

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

\[
\texttt{\colorlet{tPrim}\{ ... \}} \\
\texttt{\colorlet{tSec}\{ ... \}} \\
... 
\]

in your preamble.

### 4 Tips & Tricks

#### 4.1 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 \texttt{appendixnumberbeamer} package in your preamble and call \texttt{\appendix} before your backup slides.

Trigon will automatically turn off slide numbering for slides in the appendix.

### 5 Known Issues

#### 5.1 Title formats

Be aware that not every font supports small caps, so the \texttt{smallcaps} or \texttt{allsmallcaps} options may not work if you use a font other than \texttt{sourcesanspro}.

The title format options \texttt{allsmallcaps} and \texttt{allcaps} are quite nice from an aesthetic point of view, but their use of \texttt{\MakeLowercase} and \texttt{\MakeUppercase} can cause unexpected problems. For example:
Some commands, like \, do not work inside \texttt{\textbackslash MakeLowercase} and \texttt{\textbackslash MakeUppercase}. (See Metropolis \#125)

Only alphabetic characters are affected by \texttt{\textbackslash MakeLowercase}, so numerals and punctuation remain at full height. This can spoil some of the aesthetic benefits of \texttt{allsmallcaps}. (See Metropolis \#33)

\texttt{\textbackslash MakeLowercase} and \texttt{\textbackslash MakeUppercase} apply to math mode and \texttt{\textbackslash 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, \texttt{\textbackslash mathbb} and \texttt{\textbackslash mathcal} letters will be replaced by other math glyphs. (See Metropolis \#153)

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

6 License

Trigon is licensed under the terms of the Creative Commons Attribution-ShareAlike 4.0 license.

7 Implementation

7.1 Trigon parent theme

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

7.1.1 Package dependencies

1 \texttt{\textbackslash RequirePackage{pgfkeys}}
2 \texttt{\textbackslash RequirePackage{tikz}}

7.1.2 Options

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

3 \texttt{\textbackslash pgfkeys{/trigon/.cd},}
.search also={
  /trigon/inner,
  /trigon/outer,
  /trigon/color,
  /trigon/font
}
}

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

```
\pgfkeys{
  /trigon/titleformat plain/.cd,
  .is choice,
  regular/.code={% 
    \let\trigon@plaintitleformat\@empty%
    \setbeamerfont{standout}{shape=\normalfont}%  
  },
  smallcaps/.code={% 
    \let\trigon@plaintitleformat\@empty%
    \setbeamerfont{standout}{shape=\scshape}%  
  },
  allsmallcaps/.code={%  
    \let\trigon@plaintitleformat\MakeLowercase%
    \setbeamerfont{standout}{shape=\scshape}%  
    \PackageWarning{beamerthemetrigon}{%  
      Be aware that titleformat plain=allsmallcaps can lead to problems%  
  }
},
  allcaps/.code={%  
    \let\trigon@plaintitleformat\MakeUppercase%
    \setbeamerfont{standout}{shape=\normalfont}%  
    \PackageWarning{beamerthemetrigon}{%  
      Be aware that titleformat plain=allcaps can lead to problems%  
  }
},
}
```

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

```
\pgfkeys{
```
usesourcefonts  Select whereas the Source Sans Pro font is being loaded automatically or not.

\newif\if@trigon@usesourcefonts
\pgfkeys{/trigon/.cd,
usesourcefonts/.is if=@trigon@usesourcefonts,
usesourcefonts=true,
}

7.1.3 Component sub-packages

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

\useinnertheme{trigon}
\useoutertheme{trigon}
\usecolortheme{trigon}
\usefonttheme{trigon}

7.1.4 Custom commands

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

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

\newcommand{\trigonset}[1]{\pgfkeys{/trigon/.cd,#1}}

7.1.5 Process package options

\ProcessOptionsBeamer
\ProcessPgfOptions{/trigon}
\if@trigon@usesourcefonts%
\RequirePackage[default,t1,semibold]{sourcesanspro}
7.2 Trigon inner theme

A \texttt{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;
- main background elements;
- itemize, enumerate, and description environments;
- block environments including theorems and proofs;
- figures and tables; and
- footnotes and plain text.

7.2.1 Package dependencies

\begin{verbatim}
\RequirePackage{pgfplots}
\RequirePackage{tikz}
\usetikzlibrary{calc,3d}
\end{verbatim}

7.2.2 Definitions

We define the command \texttt{\logobig} to specify the logo that needs to be applied displayed on the title frame.

\begin{verbatim}
\def\logobig{}
\newcommand\biglogo[1]{
  \def\logobig{#1}
}
\end{verbatim}

7.2.3 Options

\texttt{sectionpage} controls whereas a slide with the section title should be inserted at the beginning of the section or not.

\begin{verbatim}
\pgfkeys{
  /trigon/inner/sectionpage/.cd,
  .is choice,
  none/.code=\trigon@disablesectionpage,
  simple/.code=\trigon@enablesectionpage
}\end{verbatim}
slidestyle Controls the layout that should be used for regular slides.

```latex
\def\slidestyle{}
\pgfkeys{/trigon/inner/slidestyle/.cd,
   .is choice,
   blank/.code=\def\slidestyle{blank}
\setbeamertemplate{background}[\slidestyle],
   style1/.code=\def\slidestyle{smallcornertriangle}
\setbeamertemplate{background}[\slidestyle],
   cyber/.code=\def\slidestyle{cyberbottom}
\setbeamertemplate{background}[\slidestyle]
}
```

titlestyle Controls the layout that should be used for the title page.

```latex
\def\titlestyle{}
\pgfkeys{/trigon/inner/titlestyle/.cd,
   .is choice,
   plain/.code=\def\titlestyle{bottomtrianglescolor},
   style1/.code=\def\titlestyle{titlesmallimage},
   style2/.code=\def\titlestyle{titlebigimage}
}
```

sectionstyle Controls the layout that should be used for the title page.

```latex
\def\sectionstyle{}
\pgfkeys{/trigon/inner/sectionstyle/.cd,
   .is choice,
   plain/.code=\def\sectionstyle{bottomtrianglescolor},
   style1/.code=\def\sectionstyle{bigtriangles},
   style2/.code=\def\sectionstyle{lefttriangles},
   style3/.code=\def\sectionstyle{bottomtrianglesbw}
}
```

\trigon@inner@setdefaults Sets default values for the inner theme options.
7.2.4 Backgrounds

First of all, we define a few commands in order easily create the triangles used as the main design elements of the theme. All these are defined with respect to the \paperwidth and \paperheight of the document, so that it fits the different aspect ratio possible.

\leftTriangle  Defines a triangle with the base on the left side (pointing right).

\leftTriangle\[2\]{
  (#1,#2)
  -- (0,{#2 + tan(30)*#1} )
  -- (0,{#2 - tan(30)*#1} )
  -- cycle;
}

\rightTriangle  Defines a triangle with the base on the right side (pointing left).

\rightTriangle\[2\]{
  (#1,#2)
  -- (\the\paperwidth,{#2 + tan(30)*(#1-\the\paperwidth-#1)} )
  -- (\the\paperwidth,{#2 - tan(30)*(#1-\the\paperwidth-#1)} )
  -- cycle;
}

\topTriangle  Defines a triangle with the base on the bottom side (pointing top).

\topTriangle\[2\]{
  (#1,#2)
  -- ({{#1 - tan(60)*#2},0)
  -- ({{#1 + tan(60)*#2},0)
  -- cycle;
}
\leftColorTriangle \text{ Defines a colored triangle with the base on the left side (pointing right).}

\begin{verbatim}
\newcommand{\leftColorTriangle}[3]{
  \path[fill=#3]
  \leftTriangle{#1}{#2}
}
\end{verbatim}

\rightColorTriangle \text{ Defines a colored triangle with the base on the right side (pointing left).}

\begin{verbatim}
\newcommand{\rightColorTriangle}[3]{
  \path[fill=#3]
  \rightTriangle{#1}{#2}
}
\end{verbatim}

\topColorTriangle \text{ Defines a colored triangle with the base on the bottom side (pointing top).}

\begin{verbatim}
\newcommand{\topColorTriangle}[3]{
  \path[fill=#3]
  \topTriangle{#1}{#2}
}
\end{verbatim}

We then define the different backgrounds choices.

\textbf{blank} \text{ Blank slide background.}

\begin{verbatim}
\defbeamertemplate{background}{blank}{}
\end{verbatim}

\textbf{smallcornertriangle} \text{ Background with a small triangle in bottom left corner (used for normal frames when the appropriate option is selected).}

\begin{verbatim}
\defbeamertemplate{background}{smallcornertriangle}{
  \begin{tikzpicture}[blend group=multiply]
  \useasboundingbox (0,0) rectangle(\the\paperwidth,\the\paperheight);
  \leftColorTriangle{0.62\paperwidth}{0}{tGrey!30!tBg}
  \end{tikzpicture}
}
\end{verbatim}

\textbf{cyberbottom} \text{ Background with "mesh" effect on the bottom left corner (used for normal frames when the appropriate option is selected).}

defbeamertemplate{background}{cyberbottom}{
\tikzstyle{cyberVertex}=[scale=1,draw=tGreyBg,circle,fill=tBg]
\tikzstyle{cyberVertex2}=[scale=0.5,draw=tGreyBg,circle,fill=tGreyBg]
\tikzstyle{cyberEdge} = [draw,color=tGreyBg]
\begin{tikzpicture}%[blend group=multiply]
\useasboundingbox (0,0) rectangle(\the\paperwidth,\the\paperheight);
\foreach \pos/name in {
  {(-0.05\paperwidth,0.75\paperheight)/0},
  {(0.02\paperwidth,0.53\paperheight)/1},
  {(-0.04\paperwidth,0.38\paperheight)/2},
  {(0.08\paperwidth,0.40\paperheight)/3},
  {(0.052\paperwidth,0.21\paperheight)/5},
  {(0.24\paperwidth,0.26\paperheight)/6},
  {(-0.08\paperwidth,-0.06\paperheight)/7},
  {(0.12\paperwidth,0.07\paperheight)/8},
  {(0.2\paperwidth,0.-0.08\paperheight)/9},
  {(0.38\paperwidth,0.06\paperheight)/10},
  {(0.56\paperwidth,-0.08\paperheight)/11}}
\node[cyberVertex] (name) at \pos {};
\foreach \pos/name in {
  {(-0.05\paperwidth,0.75\paperheight)/0},
  {(0.02\paperwidth,0.53\paperheight)/1},
  {(-0.04\paperwidth,0.38\paperheight)/2},
  {(0.08\paperwidth,0.40\paperheight)/3},
  {(0.052\paperwidth,0.21\paperheight)/5},
  {(0.24\paperwidth,0.26\paperheight)/6},
  {(-0.08\paperwidth,-0.06\paperheight)/7},
  {(0.12\paperwidth,0.07\paperheight)/8},
  {(0.2\paperwidth,0.-0.08\paperheight)/9},
  {(0.38\paperwidth,0.06\paperheight)/10},
  {(0.56\paperwidth,-0.08\paperheight)/11}}
\node[cyberVertex2] (name_2) at \pos {};
\foreach \source/ \dest in
  {0/1,1/2,1/3,2/3,2/5,3/5,3/6,5/6,5/7,7/8,8/3,8/9,9/10,10/6,10/11}
\path[cyberEdge] (\source) -- (\dest);
\end{tikzpicture}
}

\textbf{titlebigimage} Background for title page with a big image in the top left corner.
Background for the titlepage with a small image in the top left corner.

titlesmallimage Background for the titlepage with a small image in the top left corner.
**bottomtrianglescolor** Background for title/section pages with two colored triangles in the bottom.

\begin{tikzpicture}[blend group=multiply]
\useasboundingbox (0,0) rectangle(\the\paperwidth,\the\paperheight);
\leftColorTriangle{0.62\paperwidth}{0}{tTheme}
\rightColorTriangle{0.38\paperwidth}{0}{tSec}
\end{tikzpicture}

**bottomtrianglesbw** Background for title/section pages with two black and white triangles in the bottom.

\begin{tikzpicture}[blend group=multiply]
\useasboundingbox (0,0) rectangle(\the\paperwidth,\the\paperheight);
\leftColorTriangle{0.62\paperwidth}{0}{tTheme}
\rightColorTriangle{0.38\paperwidth}{0}{tGrey!30!tBg}
\end{tikzpicture}

**bigtriangles** Background for section pages with a big triangle on the left.

\begin{tikzpicture}[blend group=multiply]
\useasboundingbox (0,0) rectangle(\the\paperwidth,\the\paperheight);
\leftColorTriangle{0.80\paperwidth}{0.62\paperheight}{tTheme}
\rightColorTriangle{0.68\paperwidth}{0.58\paperheight}{tGrey!30!tBg}
\end{tikzpicture}

**lefttriangles** Background for section pages with small triangles on the left.

\begin{tikzpicture}[blend group=multiply]
\useasboundingbox (0,0) rectangle(\the\paperwidth,\the\paperheight);
\leftColorTriangle{0.52\paperwidth}{\paperheight}{tTheme}
\leftColorTriangle{0.775\paperwidth}{0}{tGrey!30!tBg}
\end{tikzpicture}
Before anything, the regular frame background is selected.

\setbeamertemplate{background}{\slidestyle}

Finally, we need to set the backgrounds properly before creating the different frames, create the special frames (title, section) and restore the regular background afterwards.

\titleframe  Sets the titleframe.

\sectionframe  Sets the titleframe.

7.2.5 Layout

The following macros define the proper position of the various elements of the frame, so the fit nicely with the different background selected.
titlebigimage  Place the text on the title frame with a big image on the top left side.

270 \defbeamertemplate{title page}{titlebigimage}
271 {
272 \vskip135pt
273 \begin{flushright}
274 \begin{minipage}[t][c]{0.5\textwidth}
275 \centering
276 \usebeamerfont{title}\usebeamercolor[fg]{title}
277 \inserttitle\par
278 \end{minipage}
279 \ifx \insertsubtitle\@empty
280 \else
281 \vskip15pt
282 \begin{minipage}[t][c]{0.5\textwidth}
283 \centering
284 \usebeamerfont{subtitle}\insertsubtitle\par
285 \end{minipage}
286 \fi
287 \vskip15pt
288 \begin{minipage}[t][c]{0.5\textwidth}
289 \centering
290 \usebeamerfont{author}\insertauthor
291 \vskip5pt
292 \usebeamerfont{date}\insertdate
293 \end{minipage}
294 \end{flushright}
295 \vfill
296 }

titlesmallimage  Place the text on the title frame with a small image on the top left side.

297 \defbeamertemplate{title page}{titlesmallimage}
298 {
299 \vskip140pt
300 \begin{beamercolorbox}[wd=\textwidth,sep=4pt]{title page header}
301 \usebeamerfont{title}\usebeamercolor{fg}{title}\inserttitle\par
302 \end{beamercolorbox}\%
303 \ifx \insertsubtitle\@empty
304 \else
305 \vskip8pt
306 \else
307 \vskip8pt
308 \end{beamercolorbox}\%
bottomtrianglescolor  Place the text on the title frame with two triangles on the bottom.

\defbeamertemplate{title page}{bottomtrianglescolor}{
\vskip0pt
\begin{center}
\begin{minipage}[t][\textwidth]
\centering
\ifx\logbig@empty \else
\includegraphics[height=0.15\textheight]{\logbig}\fi
\end{minipage}
\vskip15pt
\begin{minipage}[t][c]{\textwidth}
\centering
\usebeamerfont{title}\usebeamercolor[fg]{title}\inserttitle\par
\end{minipage}
\ifx \insertsubtitle@empty
\else
\vskip15pt
\begin{minipage}[t][c]{\textwidth}
\centering
\usebeamerfont{subtitle}\insertsubtitle\par
\end{minipage}
\fi
\end{center}
\vfill
}\end{beamercolorbox}
\begin{minipage}{0.5\textwidth} 
\centering 
\usebeamerfont{author}\insertauthor\\[0.3cm]\usebeamerfont{date}\insertdate
\end{minipage} 
\end{center} 
\vfill 
\bottomtrianglescolor
\defbeamertemplate{section page}{bottomtrianglescolor} 
{ 
\vskip40pt 
\begin{center} 
\begin{minipage}{\textwidth} 
\centering 
\usebeamerfont{section title}\usebeamercolor[fg]{title} \insertsectionhead\par 
\end{minipage} 
\end{center} 
\vfill 
\bottomtrianglesbw
\defbeamertemplate{section page}{bottomtrianglesbw} 
{ 
\vskip40pt 
\begin{center} 
\begin{minipage}{\textwidth} 
\centering 
\usebeamerfont{section title}\usebeamercolor[fg]{title} \insertsectionhead\par 
\end{minipage} 
\end{center} 
\vfill 
}
The three different block environments differ only in their colours. Rather than repeat the essentially the same template three times, we use the auxiliary macro \trigon@block to define all three templates.
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 the `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.

```latex
\ifbeamercolorempty[bg]{block title#1}{% 
  \begin{beamercolorbox}[rightskip=0pt plus 4em]{block title#1}% 
\ifbeamercolorempty[bg]{block title}{% 
  \begin{beamercolorbox}[rightskip=0pt plus 4em]{block title#1}%
% Otherwise, if the `block title` has a background, we set the padding based on `\trigon@blockskip`. However, we have to visually compensate for the `\trigon@strut` added to the block title (see below) by subtracting `\trigon@blockadjust` from the top and bottom padding.

```latex
\begin{beamercolorbox}{
  sep=dimexpr\trigon@blocksep-\trigon@blockadjust\relax,
  leftskip=\trigon@blockadjust,
  rightskip=dimexpr\trigon@blockadjust plus 4em\relax
}{block title#1}%

We can now set the contents of the `block title`. The zero-width but positive-height box `\trigon@strut` ensures that the block title box has a consistent height,
even if it lacks punctuation, ascenders, or descenders.

Next, we typeset the block body. This the code is similar to, but simpler than, the block title code since we don’t need to adjust for any struts.

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

7.2.7 Lists and floats

7.2.8 Text and spacing settings
7.2.9 Margins

The margins are defined as

\setbeamersize{text margin left=.05\paperwidth, text margin right=.05\paperwidth}

7.2.10 Process package options

\trigon@inner@setdefaults
\ProcessPgfPackageOptions{/trigon/inner}

7.3 Trigon outer theme

A beamer outer theme dictates the style of the frame elements traditionally set outside the “body” of each slide, such as the frame title, the logo, the footer, etc.

7.3.1 Package dependencies

\RequirePackage{pgfopts}

7.3.2 Definitions

We define the command \logsmall to specify the logo that needs to be applied displayed on all normal frames.

\def\logsmall{}
\newcommand\smalllogo[1]{
  \def\logsmall{#1}
}

7.3.3 Options

\textbf{numbering} Adds slide number to the bottom of each regular frame

\pgfkeys{
  /trigon/outer/numbering/.cd,
  .is choice,
  none/.code=\setbeamertemplate{frame numbering}[none],
  counter/.code=\setbeamertemplate{frame numbering}[counter],
  fraction/.code=\setbeamertemplate{frame numbering}[fraction]
}
\trigon@outer@setdefaults \hspace{1cm} Sets default values for the outer theme options.

\begin{verbatim}
464 \newcommand{\trigon@outer@setdefaults}{
465 \pgfkeys{/trigon/outer/.cd,
466   numbering=counter
467 } }
468 \end{verbatim}

7.3.4 Frame title

frametitle \hspace{1cm} Template for the frame title

\begin{verbatim}
469 \defbeamertemplate{frametitle}{trigon} { 
470 \vspace{-1pt}
471 \begin{beamercolorbox}[wd=\paperwidth, leftskip=0.05\paperwidth]{frametitle}
472 \vskip.4cm
473 \ifx\insertframesubtitle\@empty%
474 \else%
475 \fi
476 \vskip.3cm
477 \end{beamercolorbox}
478 }
479 \end{verbatim}

7.3.5 Frame header

headline \hspace{1cm} Templates for the frame header

\begin{verbatim}
483 \defbeamertemplate{headline}{none}{}
484 \defbeamertemplate{headline}{logo}{
485 \hfill\includegraphics[width=20pt]{\logsmall}\hspace{20pt}
486 \ifx\logsmall\@empty\else
487 \fi
488 }
489 \end{verbatim}

7.3.6 Frame footer

Define various template for frame footer and manage frame numbering.
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.

\AtBeginDocument{%
\apptocmd{\appendix}{%\pgfkeys{% /trigon/outer/.cd, numbering=none} }{%}{}{}}

7.3.7 Process package options

\trigon@outer@setdefaults
\ProcessPgfPackageOptions{/trigon/outer}
7.4 Trigon font theme

A \texttt{beamer} font theme sets the style of the font used in the document.

7.4.1 Package dependencies

\begin{verbatim}
\RequirePackage{pgfopts}
\end{verbatim}

7.4.2 Title format options

\texttt{titleformat title} Controls the format of the title.

\begin{verbatim}
\pgfkeys{
  /trigon/font/titleformat title/.cd,
  .is choice,
  regular/.code={%
    \let\trigon@titleformat\@empty%
    \setbeamerfont{title}{shape=\normalfont}%
  },
  smallcaps/.code={%
    \let\trigon@titleformat\@empty%
    \setbeamerfont{title}{shape=\scshape}%
  },
  allsmallcaps/.code={%
    \let\trigon@titleformat\lowercase%
    \setbeamerfont{title}{shape=\scshape}%
    \PackageWarning{beamerthemetrigon}{%Be aware that titleformat title=allsmallcaps can lead to problems%
  }%
  },
  allcaps/.code={%
    \let\trigon@titleformat\uppercase%
    \setbeamerfont{title}{shape=\normalfont}%
    \PackageWarning{beamerthemetrigon}{%Be aware that titleformat title=allcaps can lead to problems%
  }%
  }
}
\end{verbatim}

\texttt{titleformat subtitle} Controls the format of the subtitle.

\begin{verbatim}
\pgfkeys{
\end{verbatim}

29
Controls the format of the section title.

\pgfkeys{
  /trigon/font/titleformat section/.cd,
  .is choice,
  regular/.code={%
    \let\trigon@sectiontitleformat\@empty%
    \setbeamerfont{section title}{shape=\normalfont}%
  },
  smallcaps/.code={%
    \let\trigon@sectiontitleformat\@empty%
    \setbeamerfont{section title}{shape=\scshape}%
  },
  allsmallcaps/.code={%
    \let\trigon@sectiontitleformat\lowercase%
    \setbeamerfont{section title}{shape=\scshape}%
    \PackageWarning{beamerthemetrigon}{%
      Be aware that titleformat subtitle=allsmallcaps can lead to problems%
    }
  },
  allcaps/.code={%
    \let\trigon@sectiontitleformat\uppercase%
    \setbeamerfont{section title}{shape=\normalfont}%
    \PackageWarning{beamerthemetrigon}{%
      Be aware that titleformat subtitle=allcaps can lead to problems%
    }
  }
}
\let\trigon@sectiontitleformat\MakeLowercase%
\setbeamerfont{section title}{shape=\scshape}%
\PackageWarning{beamerthemetrigon}{% Be aware that titleformat section=allsmallcaps can lead to problems%
}
}

\pgfkeys{ /trigon/font/titleformat frame/.cd,
 .is choice,
 regular/.code={% \let\trigon@frametitleformat\@empty%
 \setbeamerfont{frametitle}{shape=\normalfont}%
 },
 smallcaps/.code={%
 \let\trigon@frametitleformat\@empty%
 \setbeamerfont{frametitle}{shape=\scshape}%
 },
 allsmallcaps/.code={%
 \let\trigon@frametitleformat\MakeLowercase%
 \setbeamerfont{frametitle}{shape=\scshape}%
 \PackageWarning{beamerthemetrigon}{% Be aware that titleformat frame=allsmallcaps can lead to problems%
 }
 },
 allcaps/.code={%
 \let\trigon@frametitleformat\MakeUppercase%
 \setbeamerfont{frametitle}{shape=\normalfont}%
 \PackageWarning{beamerthemetrigon}{% Be aware that titleformat frame=allcaps can lead to problems%
 }
 }
}

**titleformat frame** Controls the format of the frame title.
titleformat aliases  Allows titleformat title et al. to be used in the \usetheme declaration, where \LaTeX automatically removes all spaces.

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

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

\newcommand{\trigon@font@setdefaults}{
  \pgfkeys{/trigon/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\trigon@titleformat#1{#1}
\def\trigon@subtitleformat#1{#1}
\def\trigon@sectiontitleformat#1{#1}
\def\trigon@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.

\patchcmd{\beamer@title}{}
\patchcmd{\beamer@subtitle}{}
\patchcmd{\beamer@section}{}
\patchcmd{\beamer@frametitle}{}
Similarly, to make the \MakeLowercase and \MakeUppercase macros work in the frame title we have to patch \beamersubsection.

\patchcmd{\beamersubsection}{}
\patchcmd{\insertsubsectionhead}{\hyperlink{Navigation\the\c@page}{#1}}
\patchcmd{\insertsubsectionhead}{\hyperlink{Navigation\the\c@page}{% trigon@sectiontitleformat[#1]}}{% \@tempswatrue}{\patchcmd{\insertsubsectionhead}{\noexpand\hyperlink{Navigation\the\c@page}{% trigon@sectiontitleformat[#1]}}}% \if@tempswa\else\PackageError{beamerfontthemetrigon}{Patching section title failed}\@ehc\fi

\patchcmd{\beamersubsection}{}
\patchcmd{\insertsubsectionhead}{\noexpand\hyperlink{Navigation\the\c@page}{#1}}{% \noexpand\trigon@sectiontitleformat[#1]}}% \if@tempswa\else\PackageError{beamerfontthemetrigon}{Patching frame title failed}\@ehc\fi

7.4.3 General font definitions
\setbeamerfont{author}{size=\normalsize}
\setbeamerfont{title}{size=\LARGE,series=\bfseries}
### 7.4.4 Process package options

```latex
\setbeamerfont{section title}{size=\LARGE,series=\mdseries}
\setbeamerfont{date} {size=\small}
\setbeamerfont*{subtitle} {size=\Large}
\setbeamerfont{frametitle} {size=\LARGE}
\setbeamerfont{framesubtitle}{size=\large}
\setbeamerfont{alerted text} {size=\normalsize,series=\bfseries}
\setbeamerfont{block title}{size=\normalsize,series=\bfseries}
\setbeamerfont{block title alerted}{size=\normalsize,series=\bfseries}
\setbeamerfont{section in toc}{size=\Large}
\setbeamerfont{subsection in toc}{size=\large}
\setbeamerfont{page number in head/foot}{size=\scriptsize}
\setbeamerfont{description item}{series=\bfseries}
\setbeamerfont{caption}{size=\small}
\setbeamerfont{caption name}{series=\bfseries}
```

#### 7.5 Trigon color theme

A beamer color theme sets the colors used for the different elements of the document.

##### 7.5.1 Package dependencies

```latex
\RequirePackage{pgfopts}
```

##### 7.5.2 Options

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

```latex
\pgfkeys{
  /trigon/color/background/.cd,
  .is choice,
  dark/.code=\trigon@colors@dark,
  light/.code=\trigon@colors@light
}
```
headingcolor  Select the color to use for all headings (title, section, frame, etc.).

\def\headcol{tDefaulttxt}
\pgfkeys{/trigon/color/headingcolor/.cd,
  .is choice,
  default/.code=\def\headcol{tTxt},
  theme/.code=\def\headcol{tPrim}
}

textcolor  Select the color to use for all headings (title, section, frame, etc.).

\def\txtcol{black}
\pgfkeys{/trigon/color/textcolor/.cd,
  .is choice,
  default/.code=\def\txtcol{tTxt},
  theme/.code=\def\txtcol{tPrim!50!tTxt}
}

block  Optionally removes the light grey background to block environments like theorem and example.

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

% \begin{macro}{\trigon@color@setdefaults}
% \begin{macrocode}
\newcommand{\trigon@color@setdefaults}{
\pgfkeys{/trigon/color/.cd,
  background=light,
  block=fill,
  headingcolor=default,
  textcolor=default
}
7.5.3 Base colors

\definecolor{tGreenBlue}{HTML}{00707F}
\definecolor{tGreenBlueLight}{HTML}{5FA4B0}
\definecolor{tOrange}{HTML}{F07F3C}
\definecolor{tYellowOrange}{HTML}{F8AA00}
\definecolor{tBeigePale}{HTML}{E6E6E1}
\definecolor{tBeige}{HTML}{C6C0B4}
\definecolor{tDarkBg}{HTML}{232931}
\definecolor{tDarkFg}{HTML}{EEEEEE}
\definecolor{tLightBg}{HTML}{FFFFFF}

7.5.4 Derived elements

All the elements are set using the base colors defined here above.

\newcommand{\trigon@colors@dark}{
  \colorlet{tPrim}{tGreenBlueLight}
  \colorlet{tSec}{tGreenBlue}
  \colorlet{tAccent}{tYellowOrange}
  \colorlet{tTxt}{tDarkFg}
  \colorlet{tBg}{tDarkBg}
  \colorlet{tGreyBg}{tGrey!7!tBg}
}
\newcommand{\trigon@colors@light}{
  \colorlet{tPrim}{tGreenBlue}
  \colorlet{tSec}{tGreenBlue}
  \colorlet{tAccent}{tOrange}
  \colorlet{tTxt}{black}
  \colorlet{tBg}{tLightBg}
  \colorlet{tGreyBg}{tGrey!25!tBg}
}
\colorlet{tGrey}{tBeige}
\colorlet{tTheme}{tGreenBlue}
\setbeamercolor{background canvas}{bg=tBg}
\setbeamercolor{normal text}{fg=tPrim, bg=tBg}
\setbeamercolor{example text}{fg=tPrim}
\setbeamercolor{alerted text}{fg=tAccent}
\setbeamercolor{title}{fg=\headcol, bg= , parent=normal text}
\setbeamercolor{titlelike}{use=title, parent=title}
\setbeamercolor{author}{use=normal text, bg= , parent=normal text}
Block environments such as \texttt{theorem} and \texttt{example} have a grey-tinted background color by default. The option \texttt{block=transparent} removes the background color.

\begin{verbatim}
\newcommand{\trigon@block@transparent}{
\setbeamercolor{block title}{%
  use=normal text, 
  fg=normal text.fg, 
  bg=normal text.bg!96!fg
}
\setbeamercolor{block body}{
  use={block title, normal text}, 
  bg=block title.bg!35!normal text.bg
}
}
\newcommand{\trigon@block@fill}{
\setbeamercolor{block title}{%
  use={block title, alerted text}, 
  bg=block title.bg, 
  fg=alerted text.fg
}
\setbeamercolor{block body}{
  use={block title, normal text}, 
  bg=block title.bg!50!normal text.bg
}
}
\setbeamercolor{block title alerted}{%
  use={block title, alerted text}, 
  bg=block title.bg,
  fg=alerted text.fg
}
\setbeamercolor{block title example}{%
  use={block title, example text},
\end{verbatim}
The color of other smaller elements is defined as follows

\setbeamercolor{footnote}{fg=normal text.fg!90!normal text.bg}
\setbeamercolor{footnote mark}{fg=}
\setbeamercolor{footline}{fg=normal text.fg!50!normal text.bg, parent=normal text}
\setbeamercolor{caption}{fg=normal text.fg!60!normal text.bg, parent=normal text}

\setbeamercolor{itemize item}{use=example text, parent=example text}
\setbeamercolor{itemize subitem}{use=itemize item, parent=itemize item}
\setbeamercolor{itemize subsubitem}{use=itemize item, parent=itemize item}
\setbeamercolor{enumerate item}{use=example text, parent=example text}
\setbeamercolor{enumerate subitem}{use=enumerate item, parent=enumerate item}
\setbeamercolor{enumerate subsubitem}{use=enumerate item, parent=enumerate item}
\setbeamercolor{description item}{use=example text, parent=example text}

7.5.5 Process package options
\trigon@color@setdefaults
\ProcessPgfPackageOptions{/trigon/color}