

Documentation for MarginOutline.sty ver. 1.3

Byron Ahn*

October 2017

1 Introduction

- ♦ This package provides a new means of easily creating indented outlines
- ♦ Outside of this package, common methods of creating outlines in LaTeX employ itemize or enumerate-type environments.
 - ▶ In fact, other packages for creating outlines use them too (though perhaps behind the scenes).
- ♦ Because of this, there are some inherent limitations to what can/cannot and must/must not be done.
 - ▶ Such outlines are limited in the depth of nesting, and there are potentially undesirable interactions with other packages (e.g. numbered-example packages such as `linguex` and `expex`)
 - ▶ Also, such outlines require constant beginning or ending of itemize/enumerate/outline environments
 - This makes it clunky to skip around in adding/removing levels of indentation
 - It also makes it clunky to employ in beamer presentations
- ♦ This package defines outlining commands that are not subject to these restrictions.

2 Package Basics

- ♦ There are six options for this package:
 - `constantsize`: forces constant font size
 - `constantexindent`: forces all linguex/expex numbered-examples to be uniformly indented
 - `allbullets`: even outline items at a depth of 1 have bullets (by default, only outline items at a depth of 2 or deeper have bullets)

⁰To get my email address, switch the words before and after the period in `ahn.byron@gmail.com`.

- **varvertical**: items have a variable vertical spacing (when two items of the same depth occur in a row, the spacing is made smaller)
 - **beamerpause**: when in the beamer class is loaded and this option is invoked, pauses are inserted between items.
 - **beamerpauseatstart**: when in the beamer class is loaded and this option is invoked, pauses are inserted before (and not just in between) all items.
- ♦ There are several commands provided as well; the most useful are described below
 - ♦ Outlining basics (section 3):
 - ▶ **\startoutline**
 - Sets some spacing variables appropriately. (This is *always optional*, and you do not need it to start an outline)
 - ▶ **\stopoutline[<length>]**
 - Sets some spacing variables appropriately. (This is *always optional*, and you do not need it to end an outline)
 - When a length argument is specified by the optional argument, the indentation for numbered examples *after* the outline is set to that length.
 - When unspecified (no optional argument given), the value is 0pt.
 - ▶ **\Outline[<symbol>]{<num>}**
 - For creating outline bullet items, at depth **num**.
 - When the optional symbol argument is present, the symbol used for that bullet item is the one given as the argument
 - ▶ **\OutlinePause[<symbol>]{<num>}**
 - Same as **\Outline**, except a pause is added between items in beamer presentations.
 - ▶ **\OutlineMaybePause[<symbol>]{<num>}**
 - Same as **\OutlinePause**, except a pause is added between items if and only if (i) the document is a beamer presentation and (ii) **MarginOutline** is loaded with the **beamerpause** option.
 - ▶ **\0[<symbol>], \1[<symbol>], \2[<symbol>], \3[<symbol>], \4[<symbol>], \5[<symbol>], \6[<symbol>], \7[<symbol>], \8[<symbol>], \9[<symbol>]**
 - These are shortcut commands, which **MarginOutline** attempts to define
 - ◊ However, if any of them is already defined when **MarginOutline** is loaded, **MarginOutline** will not override them.
 - **\1** is equivalent to **\OutlineMaybePause{1}**, **\2** is equivalent to **\OutlineMaybePause{2}**, etc.


- ♦ Bullet symbols (section 4):
 - ▶ `\thefirstbullet`, `\thesecondbullet`, `\thethirdbullet`, `\thefourthbullet`,
`\thefifthbullet`, `\thesixthbullet`, `\theseventhbullet`, `\theighthbullet`
 - These commands define the symbols used for bulleting
 - ▶ `\drarrow`, `\pointright`, `\cnbullet{<num>}`, `\cnbullet*{<num>}`
 - These commands provide shortcuts to handy symbols
- ♦ Controlling Spacing (section 5):
 - ▶ `\bulletspacingfactor{<number>}`
 - Changes the vertical spacing between bullet points. Give it a number value, e.g. 1.5, to multiply the spacing between lines by 1.5.
 - ▶ `\bulletindentfactor{<number>}`
 - Changes the horizontal spacing of bullet point indentation. Give it a number value, e.g. 1.5, to multiply all the indentations by 1.5.
 - ▶ `\exgotodepth{<number>}`
 - When used before an example (either `\ex.` or `\ex \xe`), the example's is indented to the same level of indentation as the number provided.
 - ▶ `\exaddedindent{<length>}`
 - Changes how much numbered examples are indented from the (relevant) left margin. Give it a length value, e.g. 12pt, to make an indent of 12pt.
- ♦ Finally, the package defines two new environments
 - ▶ `\begin{CenterBox}[<color>][<width>]{<header>}`
`<body>`
`\end{CenterBox}`
 - Creates a centered, framed box that is centered in the middle of your outline. (See section 7.)
 - ▶ `\begin{frameoutline}[<optional paramters>]`
`\end{frameoutline}`
 - Mimics a beamer frame, with some added formatting tools to make these outline commands work better. (See section 6.) The optional parameters are the same as those provided to the `frame` environment, as defined by beamer.
 - NOTE: *This only works when used in a beamer document*

3 Outline Basics

- ◆ Unlike environments, where `\begin{}/\end{}` commands are required, there is no requirement here.
 - ▶ This structuring was motivated for making the creation of beamer slides easier
 - In other outline packages, `\begin/\end` commands are required in every frame that has an outline; this can be cumbersome.
 - ▶ The `\startoutline/\stopoutline` commands may be used to begin/end the outline
 - Their use is recommended, to ensure proper spacing adjustments take place.
 - But using them is not required (and so forgetting them will not crash L^AT_EX).
 - And there is no problem of requiring a `\stopoutline` that is local to each `\startoutline` (unlike `\begin/\end` commands).
- ◆ Creating outline items is simple; `\Outline{1}` (shortcut: `\1`) creates an outline item that is aligned with the left margin of the document's text area.
 - ▶ `\Outline{2}` (`\2`) creates an outline item that is indented one level to the right, `\Outline{3}` (`\3`) indents two levels to the right, etc. etc.
 - ▶ There is no theoretical limit on the depth of nesting
 - ▶ But beyond `\Outline{9}`, bullets are undefined
 - Thus `\Outline{10}` produces no bullet
 - ◇ Call it with a defined bullet (e.g. `\Outline[--]{10}`) or define the bullet (see section 4)
 - ▶ (However, as for the shortcut commands, L^AT_EX does not allow for commands with a numeral in any position but the first. E.g., it would treat `\10` as `\1` followed by 0)
- ◆ Skipping around in indentation can be done mindlessly
 - ▶ You can have a `\1` immediately following a `\8`, or a `\4` immediately following a `\2`, and the indentation and bulleting is done as you would expect
- ◆ New **sections**, **subsections** and **subsubsections** can be introduced in the middle of an outline
 - ▶ The section header will be sized according to the document's default (or according to whatever other package is loaded and relevant)
 - ▶ The header will also be aligned with the page's left margin. (This is achieved through the `titlesec` package.)

4 Bulleting Symbols

- ♦ The symbols used for bulleting can be redefined easily.
 - ▶ Currently, the bullet commands are defined as the following:

```
\newcommand{\thefirstbullet}{\ensuremath{\text{\texttt{\textbackslash thisblackdiamond}}}}
\newcommand{\thesecondbullet}{\ensuremath{\text{\texttt{\textbackslash thissmblacktriangleright}}}}
\newcommand{\thethirdbullet}{\ensuremath{\text{\texttt{\textbackslash thissqbullet}}}}
\newcommand{\thefourthbullet}{\ensuremath{\text{\texttt{\textbackslash diamond}}}}
\newcommand{\thefifthbullet}{\ensuremath{\text{\texttt{\textbackslash thissmalltriangleright}}}}
\newcommand{\thesixthbullet}{\ensuremath{\text{\texttt{\textbackslash thissquare}}}}
\newcommand{\theseventhbullet}{\ensuremath{\text{\texttt{\textbackslash thisvarstar}}}}
\newcommand{\theeighthbullet}{\ensuremath{\text{\texttt{\textbackslash circ}}}}
\newcommand{\theninthbullet}{\ensuremath{\text{\texttt{\textbackslash bullet}}}}
```
 - ▶ You can redefine these commands using `\renewcommand`, in the preamble, as normal.
- ♦ Defining new bullets for deeper levels uses similar logic
 - ▶ You'll need to define a symbol and edit the `\@bulletinserter` command.
 - ▶ The logic of this ought to be transparent.
- ♦ As with `\item` in `itemize` environments, you can define a different symbol for the bullet by specifying an optional argument in `[]s` after the outline level command; e.g. `\Outline[**]{3}` or `\3[**]`.
 - ▶ For convenience, there are some additional symbols provided by this package for this kind of custom bulleting:
 - ↳ `mathabx`'s `\drsh` (\hookrightarrow) is provided as `\drarrow`
 - ↳ `pifont`'s `\ding{43}` () is provided as `\pointright`
 - ↳ `pifont`'s circle numbers 1-10 ($\textcircled{1}$ - $\textcircled{10}$) are provided as `\cnbullet{<1-10>}`, and its filled circle numbers 1-10 ($\textcircled{\bullet}$ - $\textcircled{\bullet 10}$) are provided as `\cnbullet*{<1-10>}`

5 Controlling Spacing

- ♦ You can control the spacing between items, using `\bulletspacingfactor`.
 - ▶ For example, if you would like items to be twice as far apart, invoke the command `\bulletspacingfactor{2}`.
 - ▶ This command will only make all items that follow it twice as far apart.
 - ▶ It will also only affect local items; i.e. in a context like:

```
\1 { \2 \bulletspacingfactor{2} \3 \3 } \2
```

`\bulletspacingfactor{2}` will *only* be affect the spacing between the first `\2` and the first `\3`, and between the first `\3` and the second `\3` (because of the `{}`s)

- ♦ Similarly, you can control the spacing between the left margin and the bullet, using `\bulletindentfactor`
 - ▶ Just the same, invoking the command `\bulletindentfactor{2}` doubles the spacing
 - ▶ Like `\bulletspacingfactor`, this only affects items following it locally
- ♦ Finally, this package is made to be compatible with either of the following two example-numbering packages: `linguex` and `expex`.
 - ▶ By default, numbered examples are indented to be left-aligned with the text of the outline bullet in which it occurs.
 - e.g., an example after `\Outline{2} <text>` will be left-aligned with `<text>`
 - ▶ However, for some it may be more aesthetically pleasing if the numbered examples were left-aligned with the page's left margin.
 - To achieve this, load the package with `constantexindent` option.
 - ▶ You may want to set the numbered examples to be more/less indented than the default alignment (whether you use `constantexindent` or not)
 - At a *global* level, you can change the indentation with `\exaddedindent{<length>}`.
 - ◊ If you loaded the `constantexindent` option, the length you specify will indent examples from the page's left margin.
 - ◊ If not, the length you specify will indent examples from the text of the outline bullet containing it.
 - At a *local* level, you can change the indentation with `\exgotodepth{<number>}`.
 - ◊ Use this before the example whose indentation you want to change
 - ◊ e.g., `\exgotodepth{2}` will make it so that the example is left-aligned with the text of `\Outline{2}` (whether you use `constantexindent` or not)
 - ◊ (If `\exaddedindent{<length>}` has been used previously, the `<length>` will still be added)

6 A Note for Use with Beamer

- ♦ `\OutlineMaybePause` and its derivatives (`\1`, `\2`, `\3`, ...) rely on counters that check what the last outline command was
 - ▶ If `MarginOutline` detects that this is the first outline command...
 - ...and the `beamerpause` option is loaded, but `beamerpauseatstart` is not, no pause will be inserted
 - ...and the `beamerpauseatstart` option is loaded, a pause will be inserted

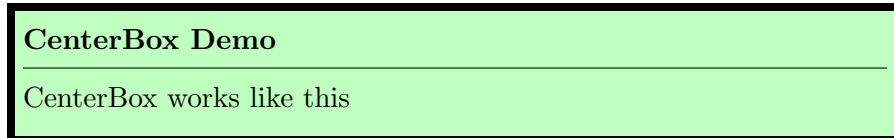
- ◆ You may want `MarginOutline` to treat some outline commands as ‘the first outline command’ for purposes of pause-insertion
 - ▶ For example, you may want the first outline command on each slide to be treated this way
 - ▶ To do this, you can insert a `\ZeroOutLeft` command at the beginning of each slide
 - Alternatively, you can define your own environment for slides, which executes `\ZeroOutLeft` every time
 - Such an environment – `frameoutline` – is provided by this package

7 An Extra Presentation Tool

- ◆ With the `CenterBox` environment, you can make boxes that stand out from the rest of your outline
 - ▶ This environment takes two optional arguments, and one mandatory argument
- ◆ To begin a `CenterBox` environment, use the command:

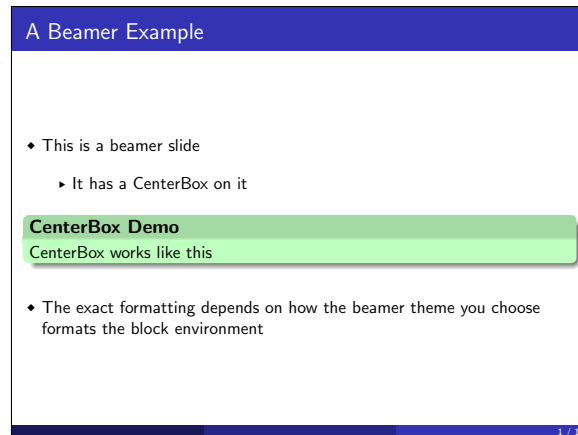

```
\begin{CenterBox}[<color>][<width>]{<header>}
```

 - ▶ The optional `color` argument specifies a background color. The default is white.
 - (In the case of beamer slides, the default is the background color of your slide.)
 - ▶ The optional `width` argument specifies the width of the border around the box. The default value is 1pt.
 - (This argument has no effect in beamer, as beamer `\CenterBoxes` are formatted according to the `beamertheme`.)
 - ▶ The `header` argument puts a header at the top of the `\CenterBox`.
- ◆ No matter how deeply embedded you are into your outline:
 - ▶ The `CenterBox` always appears in the center of the page, at a width of 75% of the line width
 - This is exemplified below:



- ◊ This was produced with the code:


```
\begin{CenterBox}[green!25][3pt]{CenterBox Demo}
CenterBox works like this
\end{CenterBox}
```
 - ▶ And here is an example of `\CenterBox` in action on a Beamer slide, produced with the same code for the `CenterBox` environment



- The full .tex for this beamer document above:

```
\documentclass[handout]{beamer}
\usetheme{Madrid}
\usepackage[allbullets]{MarginOutline}

\begin{document}

\begin{frame}{A Beamer Example}
\1 This is a beamer slide
\2 It has a CenterBox on it
\begin{CenterBox}[green!25][3pt]{CenterBox Demo}
CenterBox works like this
\end{CenterBox}
\1 The exact formatting depends on how the beamer theme you choose
formats the block environment
\end{frame}

\end{document}
```

- ♦ In addition, if you add a * after \begin{CenterBox}, the header will be suppressed:

- ▶ Here is an example:



- ▶ This was produced with the code:

```
\begin{CenterBox}*[green!25][3pt]{CenterBox Demo}
CenterBox works like this
\end{CenterBox}
```

- **NOTE:** Even though the header is suppressed in the output, you *must* include the header argument (if only as curly braces without any content: i.e. {})
- ◊ If you do not, you may get some errors