

# Beamer by Example

Andrew Mertz and William Slough

Email cfaem@eiu.edu, cfwas@eiu.edu  
Address Department of Mathematics and Computer Science  
Eastern Illinois University  
600 Lincoln Avenue  
Charleston, IL 61920-3099

**Abstract** There are a variety of L<sup>A</sup>T<sub>E</sub>X classes which can be used to produce “overhead slides” for presentations. One of these, beamer, provides flexible and powerful environments which can be used to create slides and PDF-based documents suitable for presentations. Although the class is extensively documented, first-time users may prefer learning about this class using a collection of graduated examples. The examples presented here cover a wide spectrum of use, from the simplest static slides to those with dynamic effects.

## 1 Introduction

L<sup>A</sup>T<sub>E</sub>X users in search of a way to produce “overhead slides” for use with an LCD projector have many choices today—perhaps too many! For example, Michael Wiedmann has a web site [7] that lists more than a dozen different tools, all L<sup>A</sup>T<sub>E</sub>X-based, capable of producing PDF output.

For first-time users, it can be difficult to decide which of these many approaches best matches their needs. In our experience, we have found the beamer class [6] to be easy to use, flexible, and well documented.

The user’s guide for beamer is, of course, the ultimate authority for its use. However, at 203 pages, a potential user might be frightened off before having a chance to experience its capabilities. Our intention here is to provide a sampling of beamer’s capabilities by displaying a variety of examples.

## 2 A first example

A beamer document consists of a sequence of overhead slides, known as *frames*. In the simplest case, a frame is the electronic equivalent of an overhead transparency.

Figure 1 gives a complete example of a beamer presentation, stored in a file named `talk.tex`. In the preamble, familiar  $\text{\LaTeX}$  commands appear. The body of the document specifies a title page is to appear, followed by two frames. Each frame environment specifies the desired frame title and the contents to appear on that frame.

Processing the `talk.tex` source with `pdf $\text{\LaTeX}$`  yields `talk.pdf`, a PDF file suitable for presentations. Figure 2 shows the resulting output.

Using this simple example as a template, a newcomer to beamer can produce a wide variety of presentations—in effect by learning about just one new environment, the `frame`.

## 3 Frame content

A frame can be subdivided into the following basic components, many of which are optional:

- Head line and foot line
- Left and right sidebars
- Navigation bars
- Logo
- Frame title
- Background
- Content

In our examples, we use relatively few of these components, choosing to emphasize the content over the more decorative elements which are possible. Our first example does include a navigation bar located at the lower right hand corner of the frame, which is present by default.

```
\documentclass{beamer}

\title{A Tiny Example}
\author{Andrew Mertz and William Slough}
\date{June 15, 2005}

\begin{document}

\maketitle

\begin{frame}
\frametitle{First Slide}
Contents of the first slide
\end{frame}

\begin{frame}
\frametitle{Second Slide}
Contents of the second slide
\end{frame}

\end{document}
```

Figure 1: Contents of our initial talk.tex.

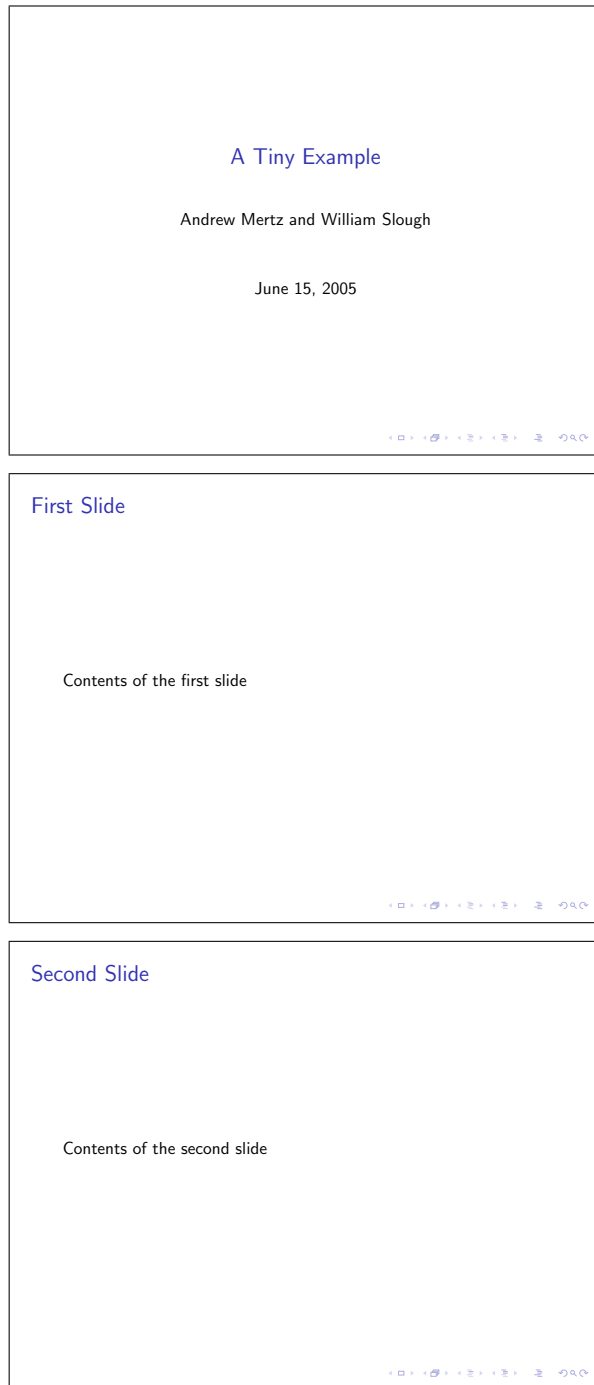


Figure 2: The three output frames of talk.pdf.

As far as frame content is concerned, most of the  $\LaTeX$  environments and commands work within beamer documents in the expected ways. So, for example, enumerated and itemized lists, mathematics, and tables can all be expressed in ways familiar to  $\LaTeX$  users. The only thing different is that these commands must appear within beamer's frame environment. To illustrate, Figures 3 and 4 show how a table can be centered on a frame.

Frames containing verbatim text require special attention. Such frames must be marked as *fragile*. This is accomplished as follows:

```
\begin{frame}[fragile]
% ... frame contents ...
\end{frame}
```

To include graphics within a frame, facilities of the `graphicx` package may be used. Several points are worth noting related to graphics. To begin with, beamer automatically loads the `graphicx` package, so no explicit `\usepackage` statement is needed. For properly sizing graphics within a frame, it helps to know that beamer formats its output to a size of 128 millimeters by 96 millimeters, or 5.04 inches by 3.78 inches. The native graphics formats supported by `pdf $\LaTeX$`  are JPEG, PNG, PDF, and MetaPost output.

Figures 5 and 6 illustrates how the graphics file named `TeX-Zone.pdf` can be placed within a frame.

## 4 Frames with color

Emphasis in presentations may be obtained by changes in color, in addition to the more traditional font changes. Since beamer automatically loads the `xcolor` package [3], colors can be specified using the syntax of `xcolor`. In particular, the “named” color model can be combined with a percentage using the `xcolor ! specifier`. For example,

```
{\color{BlueViolet!30} A B C}
```

will typeset the text “A B C” using the `BlueViolet` color, at 30% intensity.

A particularly effective use of color can be applied to a tabular environment, via the `rowcolors` command from `xcolor`. In this command, a starting row number

```

\begin{frame}
\frametitle{Famous Composers}
\begin{center}
\begin{tabular}{|l|c|}\hline
J.\ S.\ Bach    & 1685--1750 \\
W.\ A.\ Mozart & 1756--1791 \\
L.\ Beethoven  & 1770--1827 \\
F.\ Chopin     & 1810--1849 \\
R.\ Schumann   & 1810--1856 \\
B.\ Bart\'{o}k & 1881--1945 \\
\hline
\end{tabular}
\end{center}
\end{frame}

```

Figure 3: Source for a frame with a centered table.

The image shows a Beamer slide with a title bar at the top left that reads "Famous Composers". In the center of the slide is a table with two columns: the first column lists the names of composers, and the second column lists their birth and death years. The table is enclosed in a thin black border. At the bottom right of the slide, there is a small navigation bar with several icons.

J. S. Bach	1685–1750
W. A. Mozart	1756–1791
L. Beethoven	1770–1827
F. Chopin	1810–1849
R. Schumann	1810–1856
B. Bartók	1881–1945

Figure 4: Output produced by Figure 3.

```

\begin{frame}
\frametitle{Including Graphics with Beamer}
\begin{center}
\includegraphics[height=3.0in]{TeX-Zone}
\end{center}
\end{frame}

```

Figure 5: Source for a frame with included graphics.

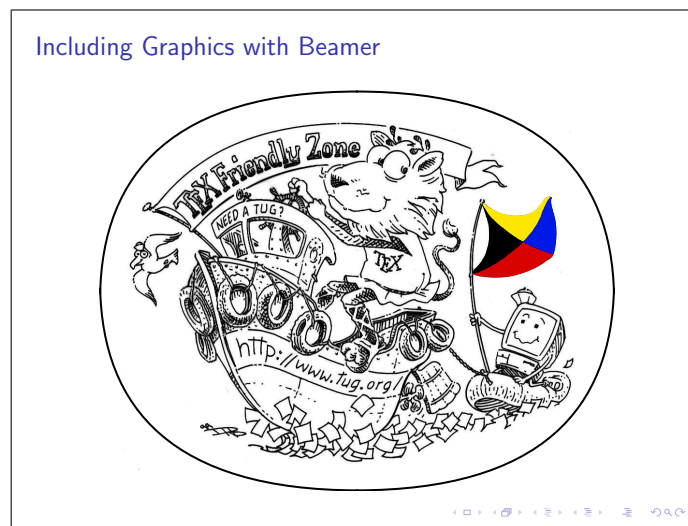


Figure 6: Output produced by Figure 5.

and two colors are specified. These two colors are used to alternately shade the rows of a table, beginning with the given row. For example,

```
\rowcolors{1}{RoyalBlue!20}{RoyalBlue!5}
```

requests two shades of RoyalBlue are to be applied to a table, beginning with its first row. This command should immediately precede the tabular environment, as shown in Figures 7 and 8.

To use color specifications like these within a beamer document, some additional options must be given within the documentclass. Ordinarily, these color capabilities would be obtained with an appropriate `\usepackage` command.

```

\begin{frame}
\frametitle{Famous Composers}
\begin{center}
\rowcolors{1}{RoyalBlue!20}{RoyalBlue!5}
\begin{tabular}{|l|c|}\hline
J.\ S.\ Bach & 1685--1750 \\
W.\ A.\ Mozart & 1756--1791 \\
L.\ Beethoven & 1770--1827 \\
F.\ Chopin & 1810--1849 \\
R.\ Schumann & 1810--1856 \\
B.\ Bart\`{o}k & 1881--1945 \\
\hline
\end{tabular}
\end{center}
\end{frame}

```

Figure 7: Source for a frame containing a table with alternating colors.

Famous Composers

J. S. Bach	1685–1750
W. A. Mozart	1756–1791
L. Beethoven	1770–1827
F. Chopin	1810–1849
R. Schumann	1810–1856
B. Bartók	1881–1945

Figure 8: Output produced by Figure 7.



However, as mentioned earlier, beamer automatically loads `xcolor` (among others), thus making the usual command, `\usepackage{xcolor}`, both unnecessary and illegal. What this means is that options we wish to specify for these packages must be given in a different way. This can be accomplished with beamer options. For example,

```
\documentclass[xcolor=pdfTeX,dvipsnames,table]{beamer}
```

specifies three different options to be used with the `xcolor` package. The first option, `pdfTeX`, provides information about the correct color driver to use. The option `dvipsnames` allows a set of predefined color names, such as `RoyalBlue`, to be used. (These named colors are sometimes referred to as the “Crayola” colors.) Finally, the `table` option informs `xcolor` that the `colortbl` package needs to be loaded. It is this last option that defines the `\rowcolors` command used in Figure 7.

## 5 Frames with two columns

Since frames have a landscape orientation, it can be helpful to be able to subdivide a frame into two columns. The `columns` environment of beamer is designed to meet this need.

Figures 9 and 10 provides an example of how a frame can be subdivided into two columns. The `columns` environment allows an alignment option which specifies whether columns are to be aligned along their top line, bottom line, or centered. In this example, the `c` option causes the two columns to be aligned along their vertical centers. Within the `columns` environment two columns appear, as specified with the two `\column` commands.

## 6 Frames with dynamic effects

When using overhead projectors, speakers may choose to superimpose multiple transparencies in order to gradually build up a final image. In beamer, each component of this final image is known as an *overlay*.

Up to this point, the frames we have considered consisted of a single overlay. When the frame is displayed, everything on that frame appears at once.

```

\begin{frame}
\frametitle{Two Column Output}

\begin{columns}[c]
\column{1.5in}
Practical \TeX\ 2005\\
Practical \TeX\ 2005\\
Practical \TeX\ 2005

\column{1.5in}
\framebox{\includegraphics[width=1.5in]{p2005}}
\end{columns}

\end{frame}

```

Figure 9: Source for a double-column frame.

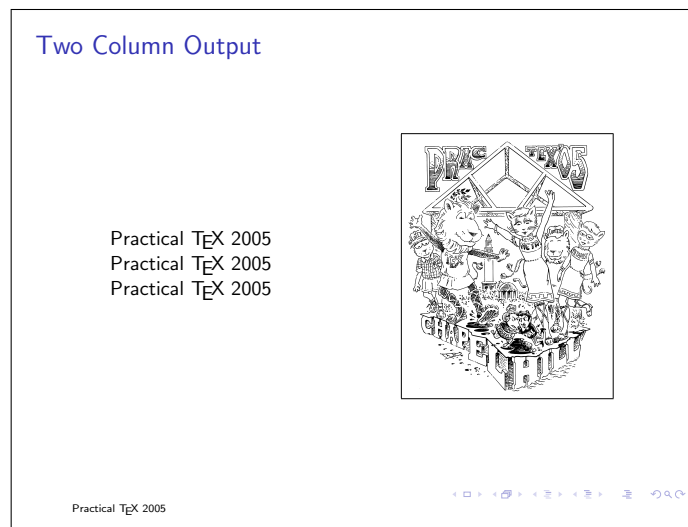


Figure 10: Output produced by Figure 9.

Alternatively, a frame can consist of a sequence of overlays, which can be used to support incremental display. Overlays can be used to “hold back” information during a presentation or to produce certain kinds of animated effects.

As a matter of taste, some people feel that a single overlay is preferable, since information is not hidden from the audience. However, we feel that there are situations where multiple overlays are appropriate, especially when used judiciously.

The beamer class provides numerous ways to specify frames with multiple overlays. We illustrate three techniques: using the `\pause` command, using overlay specifications, and including multiple graphics files

Using the `\pause` command is a simple way to produce overlays. All text from the beginning of the frame to the place where a `\pause` command appears is formatted and placed into an overlay. In this way, the example in Figure 11 creates a frame with three overlays. When the resulting PDF file is viewed, the three lines of output are incrementally revealed.

For this example, the command

```
\setbeamercovered{dynamic}
```

is in effect, so overlays not yet revealed will faintly appear. This allows the speaker to focus on the current overlay, yet not entirely hide information from the audience.

Our second example of incremental display involves overlay specifications. In beamer, every overlay within a frame is assigned a number, starting with one, reflecting the order in which they are displayed. An overlay specification describes a sequence of overlays and is denoted with angle brackets. For example, the

```
\begin{frame}
\frametitle{Overlays with {\tt pause}}
\setbeamercovered{dynamic}
Practical \TeX\ 2005\\ \pause
Practical \TeX\ 2005\\ \pause
Practical \TeX\ 2005
\end{frame}
```

Figure 11: Specifying multiple overlays with pause.

specification `<1,3,5>` indicates overlays 1, 3, and 5 and `<3->` describes all overlays beginning with the third.

Figure 12 illustrates how a game of tic-tac-toe can be displayed. In this example, there are ten overlays—one for the grid, and one for each successive play in the game.

The use of the syntax

```
\onslide<m->{text}
```

in Figure 12 indicates that the specified *text* is to appear on every overlay from *m* onwards.

Initially, only the grid should appear, so no X or O appears on the first overlay. In this example, the first play is an X in the upper right-hand corner, so this X appears on overlay 2 and every successive overlay. O counters by playing in the center, so this O appears on overlay 3 and every successive overlay. The rest of the example follows in a similar way. In this situation, we do want to hide from view overlays not yet revealed, so the command

```
\setbeamercovered{invisible}
```

is appropriate. It is worth noting that beamer makes the overlay specification feature available to other L<sup>A</sup>T<sub>E</sub>X commands, such as `\item`.

For our third example of incremental effects, we use the facilities of the package `xmpmulti` and a drawing tool which supports multiple layers. We use `xfig`, but many other choices are possible.

Revisiting the game of tic-tac-toe, a grid can be drawn on layer 0, followed by the first move on layer 1, the second move on layer 2, and so forth. After all layers of the drawing are complete, each layer is exported to a PDF file with a suffix which matches the layer number. For example, layer 0 is exported to `game-0.pdf`, layer 1 is exported to `game-1.pdf` and so forth. The `\multiinclude` command of the `xmpmulti` package, illustrated in Figure 13, causes the graphics files to appear as overlays.

In beamer, overlays are numbered beginning with 1, whereas `xmpmulti` considers the first overlay to begin with 0.

```

\begin{frame}
\frametitle{Tic-Tac-Toe via {\tt tabular}}
\setbeamercovered{invisible}
{\Huge
\begin{center}
\begin{tabular}{c|c|c}
\onslide<9->{0} & \onslide<8->{X} & \onslide<2->{X} \\ \hline
\onslide<6->{X} & \onslide<3->{0} & \onslide<5->{0} \\ \hline
\onslide<10->{X} & \onslide<7->{0} & \onslide<4->{X}
\end{tabular}
\end{center}
}
\end{frame}

```

Figure 12: Using overlay specifications.

```

\begin{frame}
\frametitle{Tic-Tac-Toe via Graphics Files}
\setbeamercovered{invisible}
\begin{center}
\multiinclude[format=pdf,width=3in]{game}
\end{center}
\end{frame}

```

Figure 13: Specifying multiple overlays with graphics files. The files to be included are named `game-0.pdf`, `game-1.pdf`, `...`, `game-9.pdf`.

## 7 Ornamental aspects

It is possible to “dress up” beamer presentations in a variety of ways. In Section 3, we noted that a frame includes a number of components, one of which is a foot line. Authors often employ a running foot line with their name, conference title, date, or some similar identification. In beamer, this can be accomplished by including a command in the preamble, similar to the following:

```
\setbeamertemplate{footline}[text line]{A Sample Talk}
```

As another example, one could use the Microsoft Comic Sans font, which can be made available with the `comicsans` package [5]. Once this font has been established within the  $\text{\TeX}$  system, adding the following two lines to the preamble of the beamer document will make it the default font:

```
\usepackage{comicsans}  
\renewcommand{\sfdefault}{comic}
```

A frame with the Comic Sans font is shown in Figure 14. Alternately, one might choose to utilize the Arev Sans font [2] which is well suited for computer projectors.

Another possibility is to choose from among the many different beamer *themes*. As an example, adding the following lines to the preamble of our beamer presentation gives the title frame shown in Figure 15.

```
\usepackage{beamerthemesplit}  
\usetheme{Berkeley}  
\usecolortheme{dolphin}
```

In addition to the visual appeal of themes, additional navigation tools are incorporated in the frames based on the  $\text{\LaTeX}$  sections and subsections present in the beamer presentation.

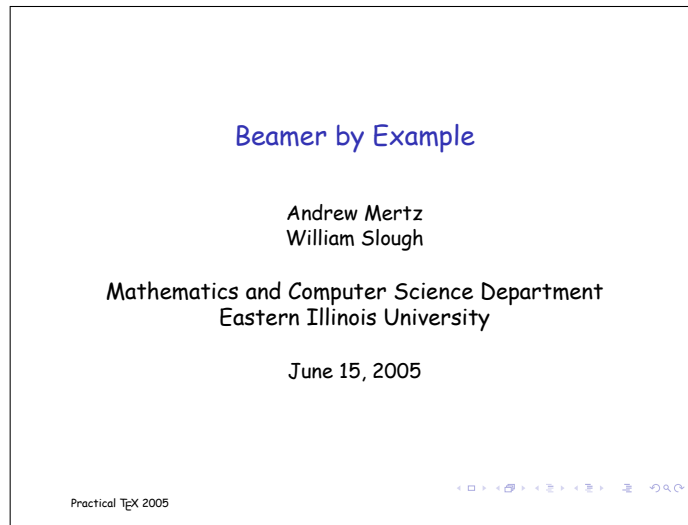


Figure 14: Title frame composed with the Microsoft Comic Sans font.

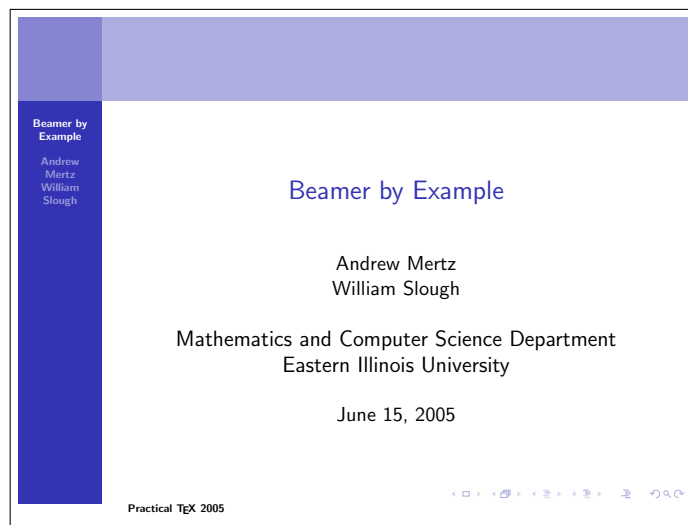


Figure 15: Title frame composed with the Berkeley theme.

## 8 Producing handouts

It is often desirable to produce a printed document which mirrors the content of the beamer presentation. There are two steps needed to accomplish this; first, create an overlay-free version of the presentation, and second, produce an N-up version of the presentation.

Removing overlays from a beamer presentation is easily done within the preamble, by adding the handout option:

```
\documentclass[handout,xcolor=pdftex,dvipsnames,table]{beamer}
```

Processing a beamer document with this option causes all overlays for a given frame to be collapsed into a single frame.

Once overlays have been removed, putting multiple frames onto a single sheet of paper is a separate problem related to PDF files. The pdfpages package [4], for example, solves this problem. For an automated approach based on the same package, pdfjam [1], a Unix shell script, can be used.

## 9 Conclusion

The contribution of many individuals in the L<sup>A</sup>T<sub>E</sub>X community have made it possible to produce overhead slides using typesetting standards of the highest quality. We are especially indebted to the work of Till Tantau and the other package designers cited earlier.

## References

- [1] David Firth. PDFjam. <http://www2.warwick.ac.uk/fac/sci/statistics/staff/academic/firth/software/pdfjam>.
- [2] Stephen G. Hartke. *Arev Sans for T<sub>E</sub>X and L<sup>A</sup>T<sub>E</sub>X*. <http://ctan.org/get?fn=/fonts/arev/doc/fonts/arev/arevdoc.pdf>.
- [3] Uwe Kern. *Extending L<sup>A</sup>T<sub>E</sub>X's color facilities: the xcolor package*. <http://www.ctan.org/tex-archive/macros/latex/contrib/xcolor>.



- [4] Andreas Matthias. *The pdfpages package*.  
<http://www.ctan.org/tex-archive/macros/latex/contrib/pdfpages/>.
- [5] Scott Pakin. *The comicsans package*.  
<http://www.ctan.org/tex-archive/macros/latex/contrib/comicsans>.
- [6] Till Tantau. *User's Guide to the Beamer Class, Version 3.01*.  
<http://latex-beamer.sourceforge.net>.
- [7] Michael Wiedmann. *Tools for Creating Screen or Online Presentations*.  
<http://www.miwi.e.org/presentations/presentations.html>.