

# Org-Mode Beamer Example

Derek Feichtinger

*[2015-07-12 Sun]*

# Outline

- 1 Introduction
- 2 A collection of example pages
- 3 Animations by overlays
- 4 Multiple Columns
- 5 Conclusions

- 1 Introduction
- 2 A collection of example pages
- 3 Animations by overlays
- 4 Multiple Columns
- 5 Conclusions

# Introduction

# Instructions

Look at the **Org source** file to learn about available options. I also added many comments explaining the usage, there.

- generating presentation notes.
- inserting a table of contents with the current section highlighted at the beginning of each section.
- configuring transparency of yet uncovered overlay elements.

# Org mode version information

Emacs version:

GNU Emacs 24.5.1 (x86\_64-unknown-linux-gnu, GTK+ Version 3.10.8)  
of 2015-05-04 on dflt1w

org version: 8.2.10

# Sources and Links

- I started this example based on [the Worg hosted example by Eric S. Fraga](#)
- Basic L<sup>A</sup>T<sub>E</sub>X Beamer links
  - [An introduction to Beamer \(German\)](#)
  - great [beamer reference card](#) by Fabrice Niessen on GitHub.
  - nice link for choosing a theme: [beamer theme matrix](#)
  - [nice example of beamer features \(pure Latex\)](#)

# A simple slide

This slide consists of some text with a number of bullet points:

- the first, very **important**, point!
- the previous point shows the use of the special markup which translates to the Beamer specific *alert* command for highlighting text.

The above list could be numbered or any other type of list and may include sub-lists.



# A more complex slide

This slide illustrates the use of Beamer blocks. The following text, with its own headline, is displayed in a block:

## Theorem (Org mode increases productivity)

- *org mode means not having to remember  $\text{\LaTeX}$  commands.*
- *it is based on ascii text which is inherently portable.*
- *Emacs!*



# Tables

The size of the table font can be chosen by giving a #+LATEX: `\small` command (or `\tiny` or `\footnotesize`)

WNs	Processors	Cores/node	HS06/node	total cores	total HS06
20	2*Xeon X5560	8	118	160	2360
11	2*E5-2670 2.60GHz	16	263	176	2893
4	2*AMD 6272 2.40GHz	32	241	128	964
35				464	6217

- 1 Introduction
- 2 A collection of example pages
- 3 Animations by overlays
- 4 Multiple Columns
- 5 Conclusions

# block environments

## a block

```
\begin{block}{A block}  
...  
\end{block}
```

## an alert block

```
\begin{alertblock}{An alert block}  
...  
\end{alertblock}
```

## an example block

```
\begin{exampleblock}{An alert block}  
...  
\end{exampleblock}
```

## some more blocks

The beamercolorbox does not seem to work

```
\begin{beamercolorbox}[shadow=true, rounded=true]{eecks}  
...  
\end{beamercolorbox}
```

A `fullframe` is a frame with an ignored slide title. `frametitle` is set to the empty string

- A headline with an `ignoreheading` environment will only have its contents displayed in the output. The heading text itself is ignored, and no heading bar is shown.

- Contents are not inserted in any `frame` environment.

- `ignoreheading` is useful as a structural element in order to again place normal text after a previous element (like a block or a column environment).

# structureenv environment

- For highlighting text.
- To help the audience see the structure of your presentation.
- **TODO:** Currently I see no effect on the text style when using structureenv
- might need to use ignoreheading (like here) in order to then insert some more normal text after the structureenv.



## Definition (definition)

Contents of the definition

# proof environment and revealing line by line

proof.

- Suppose  $p$  were the largest prime number.
- Let  $q$  be the product of the first  $p$  numbers.
- Then  $q + 1$  is not divisible by any of them.
- But  $q + 1$  is greater than 1, thus divisible by some prime number not in the first  $p$  numbers. □

# proof environment and revealing line by line

proof.

- Suppose  $p$  were the largest prime number.
- Let  $q$  be the product of the first  $p$  numbers.
- Then  $q + 1$  is not divisible by any of them.
- But  $q + 1$  is greater than 1, thus divisible by some prime number not in the first  $p$  numbers. □

# proof environment and revealing line by line

proof.

- Suppose  $p$  were the largest prime number.
- Let  $q$  be the product of the first  $p$  numbers.
- Then  $q + 1$  is not divisible by any of them.
- But  $q + 1$  is greater than  $1$ , thus divisible by some prime number not in the first  $p$  numbers. □

# proof environment and revealing line by line

proof.

- Suppose  $p$  were the largest prime number.
- Let  $q$  be the product of the first  $p$  numbers.
- Then  $q + 1$  is not divisible by any of them.
- But  $q + 1$  is greater than  $1$ , thus divisible by some prime number not in the first  $p$  numbers. □

- 1 Introduction
- 2 A collection of example pages
- 3 Animations by overlays
- 4 Multiple Columns
- 5 Conclusions

# Highlighting text

The double @@ can be used to enclose active code. Here we use it to specify beamer code that will highlight text by specifying an overlay.

A useful feature

# Highlighting text

The double @@ can be used to enclose active code. Here we use it to specify beamer code that will highlight text by specifying an overlay.

A **useful** feature



# Lists

For the first list we use an `#+ATTR_BEAMER: :overlay +/-` specification. It acts like `\begin{itemize}[<+>]`. So, it will cause the list items to appear one after the other.

- item 1
- item 2
- item 3

For the second list we classify each line by angular brackets

- item 1
- item 2
- item 3

# Lists

For the first list we use an `#+ATTR_BEAMER: :overlay +/-` specification. It acts like `\begin{itemize}[<+>]`. So, it will cause the list items to appear one after the other.

- item 1
- item 2
- item 3

For the second list we classify each line by angular brackets

- item 1
- item 2
- item 3

# Lists

For the first list we use an `#+ATTR_BEAMER: :overlay +/-` specification. It acts like `\begin{itemize}[<+>]`. So, it will cause the list items to appear one after the other.

- item 1
- item 2
- item 3

For the second list we classify each line by angular brackets

- item 1
- item 2
- item 3

# Two blocks animated by BEAMER\_act

## First Block

- this is visible from the beginning

## Second Block

- and this one is revealed afterwards by using the `BEAMER_act` keyword in the `PROPERTIES` section.

# Two blocks animated by BEAMER\_act

## First Block

- this is visible from the beginning

## Second Block

- and this one is revealed afterwards by using the BEAMER\_act keyword in the PROPERTIES section.

# Two blocks with invisible layering

## First Block

- this is visible from the beginning

# Two blocks with invisible layering

## First Block

- this is visible from the beginning

## Second Block

- and this one is revealed afterwards by using the `BEAMER_act` keyword in the `PROPERTIES` section.

# different transparency setting and default overlay

## First Block

this is visible from the beginning. Note that we specified another transparency compared to the previous slide.

## Second Block

and this is revealed later by using the `BEAMER_act` keyword in the frame's `PROPERTIES` section.

## Third Block

Same here!



# different transparency setting and default overlay

## First Block

this is visible from the beginning. Note that we specified another transparency compared to the previous slide.

## Second Block

and this is revealed later by using the `BEAMER_act` keyword in the frame's `PROPERTIES` section.

## Third Block

Same here!

# different transparency setting and default overlay

## First Block

this is visible from the beginning. Note that we specified another transparency compared to the previous slide.

## Second Block

and this is revealed later by using the `BEAMER_act` keyword in the frame's `PROPERTIES` section.

## Third Block

Same here!

# plain text between two blocks

## block 1

### The first block

A plain text paragraph. I only managed to get the right uncovering behavior by using `#+LATEX: \onslide<2->` in front of the paragraph.

## block 2

### The second block

# plain text between two blocks

## block 1

### The first block

A plain text paragraph. I only managed to get the right uncovering behavior by using `#+LATEX: \onslide<2->` in front of the paragraph.

## block 2

### The second block

# plain text between two blocks

## block 1

### The first block

A plain text paragraph. I only managed to get the right uncovering behavior by using `#+LATEX: \onslide<2->` in front of the paragraph.

## block 2

### The second block

- 1 Introduction
- 2 A collection of example pages
- 3 Animations by overlays
- 4 Multiple Columns**
- 5 Conclusions

# Blocks in two columns

## A left block

- this slide consists of two columns
- This is the first column

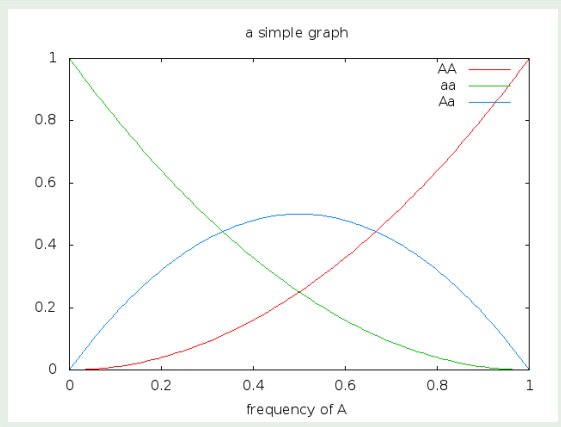
## A right block

- this is the right column

# A text section and a figure

- this slide consists of two columns
- the first (left) column has no heading and consists of text
- the second (right) column has an image and is enclosed in an **example** block

## Example (A screenshot)

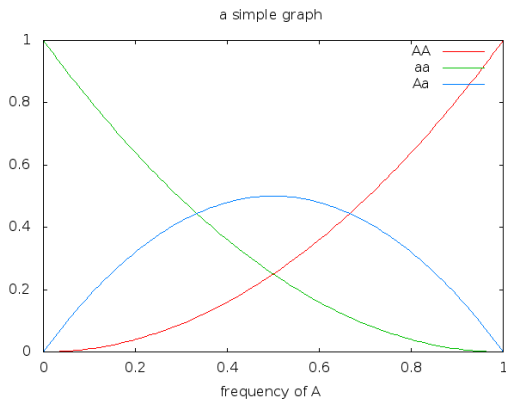




# A centered text section and a figure

- a centered text section. I found no good way for using `\vfill` or `\minipage` as referenced [here](#)

## Example (A screenshot)



## Octave code

```
A = [1 2 ; 3 4]
b = [1; 1];
x = A\b
```

## The output

```
A =

     1     2
     3     4

x =

    -1
     1
```

- 1 Introduction
- 2 A collection of example pages
- 3 Animations by overlays
- 4 Multiple Columns
- 5 Conclusions

# Summary

- org is an incredible tool for time management
- **but** it is also excellent for writing and for preparing presentations
- Beamer is a very powerful  $\text{\LaTeX}$  package for presentations
- the combination is unbeatable!

# Appendix

SOME BACKUP SLIDES. The Appendix will not be listed in the table of contents.

# Backup slide 1

Some backup info

# Backup slide 2

These details are not part of the main talk.