Org Beamer quick reference card

Fabrice Niessen

June 10, 2014

Plan

Description

Welcome to Org Beamer reference card. It contains the reference documentation that describes how to write presentations using:

- Org mode 8 and
- the LaTeX Beamer class.

Those free tools allow you to easily produce high quality PDF files which are going to look on *every* computer exactly the way they looked on *your* computer.

Objectives

As they are created like any other Org mode document, preparing presentations with Org Beamer is very different from preparing them with WYSWYG programs such as PowerPoint, Impress or Keynote.
The obvious advantage of this approach is that you don't have to know LaTeX in order to create Beamer presentations.

Requirements

- A working LaTeX installation is required for exporting to PDF. If it is not yet installed on your system, install TeX Live (for example).
- You must define a `beamer` class in `org-latex-export-classes`:

```
(eval-after-load "ox-latex"

  ;; update the list of LaTeX classes and associated header (encoding, etc.)
  ;; and structure
  '(add-to-list 'org-latex-classes
                `("beamer"
                  ,(concat "\\documentclass[presentation]{beamer}\n"
                           "[DEFAULT-PACKAGES]"
                           "[PACKAGES]"
                           "[EXTRA]\n")
                  ("\\section{%s}" . "\\section*{%s}")
                  ("\\subsection{%s}" . "\\subsection*{%s}")
                  ("\\subsubsection{%s}" . "\\subsubsection*{%s}"))))
```

Creating a title page I

A title page is automatically inserted into the first frame. By default, it will arrange the following elements on the title page:

- the document title

```
#+TITLE: Document title
```

  (file name, if none specified)

- the author(s)'s name

```
#+AUTHOR: Author
```

  (Emacs Lisp variable `user-full-name`, if none specified)

- a date

```
#+DATE: 2014-06-11
```

  (LaTeX macro `\today`, if none specified)

Creating a title page II

The author's email can be included with:

```
#+AUTHOR:     \href{mailto:email@example.com}{Author}
#+AUTHOR:     \texorpdfstring{Author\newline\url{email@example.com}}{Author} % DOES
#+BEAMER_HEADER: \author{\texorpdfstring{Author\newline\url{email@example.com}}{Aut
```

Other elements:

- the document subtitle,
- their affiliation (institute), and
- a title graphic

can be included with the following commands:

```
#+BEAMER_HEADER: \subtitle{Document subtitle}
#+BEAMER_HEADER: \institute[INST]{Institute\\\url{http://www.institute.edu}}
#+BEAMER_HEADER: \titlegraphic{\includegraphics[height=1.5cm]{InstLogo}}
```

XXX Why do I have to use :eval no (in Org blocks)?
The inner theme dictates how the title page is rendered.

```
#+BEAMER_HEADER: \logo{\includegraphics[height=.9cm]{InstLogo}}
```

Global structure

Org mode presentations contain headings at different levels. By default,

- Headings at the first outline level will become titles of the different slides (called frames in Beamer),
- Deeper levels will be used as structural environments, and
- The table of contents frame is blank.
  You can remove it by setting the `toc` option from the `#+OPTIONS:` keyword to `nil`:

```
#+OPTIONS: toc:nil
```

  (default: `toc:t`)

Creating a simple frame

```
* Introduction

** A title
   #+BEAMER: \framesubtitle{A subtitle}

Some content.
```

The subtitle does not have an Org syntax because it's specific to the Beamer back-end only.

Creating a table of contents

- If you set the `H` option from the `#+OPTIONS:` keyword such as:

```
#+OPTIONS: H:2
```

  then:

  - First-level headlines become sections listed in the table of contents (created by default), and
  - Second-level headlines become the frames.

- If you set the `H` option such as:

```
#+OPTIONS: H:3
```

  then:

  - First- and second-level headlines become sections and subsections listed in the table of contents, and
  - Third-level headlines become the frames.

In many themes, sections (and subsections, when `H:3`) appear in the sidebar or headline.

Create a handout

You can print your presentation in the form of handouts. When there are animations, it will remove the pauses (avoid the overlays) and just print the last "slide" of each frame.

```
#+LATEX_CLASS_OPTIONS: [handout]

#+LaTeX_HEADER: \usepackage{pgfpages}
#+LaTeX_HEADER: \mode<handout>
#+LaTeX_HEADER: {
#+LaTeX_HEADER:   ... see below ...
#+LaTeX_HEADER: }
```

- with one frame per page (page size extended to A4)

```
#+LaTeX_HEADER:   \pgfpagesuselayout{resize to}[a4paper,landscape]
```

- with two frames per page

```
#+LaTeX_HEADER:   \pgfpagesuselayout{2 on 1}[a4paper,border shrink=5mm]
```

- with four frames per page

```
#+LaTeX_HEADER:   \pgfpagesuselayout{4 on 1}[a4paper,border shrink=5mm,%
#+LaTeX_HEADER:                              landscape]
```

Draw a border around the frames in the handout

Add a rectangle around each frame:

```
#+LaTeX_HEADER: \setbeamertemplate{background canvas}{
#+LaTeX_HEADER:   \tikz \draw (current page.north west) rectangle
#+LaTeX_HEADER:                (current page.south east);
#+LaTeX_HEADER: }
```

Show speaker notes

Show reminders about what to say during each part of your presentation.
Your laptop monitor and your projector should have the same resolution.
http://freakazoid.teamblind.de/2011/03/30/latex-presentations-with-notes-on-windows-7/

Print handout with speaker notes

xxx

```
#+LaTeX_HEADER: \usepackage{handoutWithNotes}
#+LaTeX_HEADER: \pgfpagesuselayout{3 on 1 with notes}[a4paper,border shrink=5mm]
```

Print as article

Present a bibliography

LaTeX class

```
#+LATEX_CLASS: beamer
#+LATEX_CLASS_OPTIONS:
```

Common options:

- 8pt, 9pt, 10pt, 11pt, 12pt, 14pt, 17pt, 20pt
- draft: no graphics, footlines,...
- handout: no overlays

,#+LaTeX_CLASS_options: [bigger,allowframebreaks]

LaTeX preamble

Append any line of code in the LaTeX preamble with:

```
#+LaTeX_HEADER:       \usepackage{...}
#+LaTeX_HEADER_EXTRA: \usepackage{...}
#+BEAMER_HEADER:      \institute[short name]{Institute's name}
```

It will go (in that order) in the `[EXTRA]` placeholder of the header associated to the `beamer` LaTeX class (see `org-latex-classes`).

Affiliated keywords

The Beamer back-end reads both

- `#+ATTR_LATEX:` and
- `#+ATTR_BEAMER:`

affiliated keywords.

Appearance of the presentation

```
#+BEAMER_THEME: Boadilla
```

is equivalent (for Boadilla) to:

```
#+BEAMER_COLOR_THEME: dolphin
#+BEAMER_FONT_THEME:  default
#+BEAMER_INNER_THEME: [shadow]rounded
#+BEAMER_OUTER_THEME: infolines
```

Beamer back-end (for Org export engine)

Type:

```
M-x load-library RET ox-beamer RET
```

to load the Beamer back-end library, and to obtain extra commands in the LaTeX export menu:

`C-c C-e l B` Export as LaTeX buffer (Beamer).

`C-c C-e l b` Export as LaTeX file (Beamer).

`C-c C-e l P` Export as PDF file (Beamer).

`C-c C-e l O` Export as PDF file and open (Beamer).

Structure editing support

Type:

```
M-x org-beamer-mode RET
```

to load the minor mode `org-beamer-mode` easing the edition of the document structure (through the key binding `C-c C-b`, which offers fast selection of a Beamer environment).
You can also turn it on with:

```
#+STARTUP: beamer
```

in your document.

Column view

For a column view of options and configurations for the individual frames

```
#+COLUMNS: %45ITEM %10BEAMER_env(Env) %10BEAMER_act(Act) %4BEAMER_col(Col) %8BEAMER_
#+COLUMNS: %20ITEM %13BEAMER_env(Env) %6BEAMER_envargs(Args) %4BEAMER_col(Col) %7BE
```

Environment specification (BEAMER_env property)

XXX Put = around BEAMER_env in title. . .

- This becomes visible through the `B_frame` tag (visual aid only).

frame

- Headlines become frames when their level is equal to `org-beamer-frame-level` (or `H` value in the `OPTIONS` line).
- Though, if a headline in the current tree has a `BEAMER_env` property set to either `frame` or `fullframe`, its level overrides the variable, giving you some flexibility in deciding what is and what isn't a frame.
  This works in both "directions": to add or to remove sectioning levels above the current headline (which becomes a frame)!

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

Blocks

Environment specification (BEAMER_env property)

XXX Use ~ or = in title
Use a different block type for the current "block" environment (default: `block`).

`structureenv` environment

- For highlighting text.
- To help the audience see the structure of your presentation.

Paragraph Heading.

`block` environment

Answered Questions

How many primes are there?

Blocks

Environment specification (BEAMER_env property)

XXX Use ~ or = in title
Use a different block type for the current "block" environment (default: `block`).

`structureenv` environment

- For highlighting text.
- To help the audience see the structure of your presentation.

Paragraph Heading.

`block` environment

Answered Questions

How many primes are there?

Blocks

Environment specification (BEAMER_env property)

XXX Use ~ or = in title
Use a different block type for the current "block" environment (default: `block`).

`structureenv` environment

- For highlighting text.
- To help the audience see the structure of your presentation.

Paragraph Heading.

`block` environment

Answered Questions

How many primes are there?

Special cases

You can add an appendix (frames that you do not intend to show during your talk, but which might be useful to answer a question) by inserting such a level 1 headline after the last regular slide of your actual presentation:

```
* Appendix material follows                                    :B_appendix:
  :PROPERTIES:
  :BEAMER_env: appendix
  :END:

# Backup slides
```

Ignoring page number in backup slides can be achieved by setting the option `noframenumbering` on all "backup" slides.

noteNH

Note with its title ignored.

againframe

You can "continue" frames that you previously started

overprint
> What may not be easy or possible is to use the directive, which is > what I used in my previous response to you.
You can always use the only environment. https://github.com/suvayu/.emacs.d/blob/master/org-mode-config.el#L215
That said, I think overlays with only is not as smooth as with simple overlay specifications to regular environments or macros like `\includegraphics`, `\item`, etc.
As for an :overlay specification, I believe it is already supported but only for lists (ox-beamer.el:725). I would love to have that for images too!

Overlay specification (BEAMER_act property)

Set overlay specifications in current block to create dynamic effects (*multiple slides*, called *overlays*, for a single frame) = old `BEAMER_envargs` property.
Headlines support the `BEAMER_act` property:

```
* Headline
  :PROPERTIES:
  :BEAMER_act: [+-]
  :END:

  # Diff with [<+->]?

  - Item
  - Item
```

It is translated as:

- an overlay/action specification, or

Overlay specification (BEAMER_act property)

Set overlay specifications in current block to create dynamic effects (*multiple slides*, called *overlays*, for a single frame) = old `BEAMER_envargs` property.
Headlines support the `BEAMER_act` property:

```
* Headline
  :PROPERTIES:
  :BEAMER_act: [+-]
  :END:

  # Diff with [<+->]?

  - Item
  - Item
```

It is translated as:

- an overlay/action specification, or
- a default overlay specification when enclosed within square brackets.

The Queen's old armchair

- Princess Anne
- Prince Charles
- corgis

Question on ML

```
\begin{figure}
  \begin{center}
    \includegraphics<1>[width=.7\textwidth]{figure1}
    \includegraphics<2>[width=.7\textwidth]{figure2}
    \includegraphics<3->[width=.7\textwidth]{figure3}
   \end{center}
\end{figure}
```

The following works for me:

```
#+beamer: \only<1>{
[[file:figure1.png]]
#+beamer: }\only<2>{
[[file:figure2.png]]
#+beamer: }\only<3->{
[[file:figures3.png]]
#+beamer: }
```

There is the `BEAMER_act` property that can be used to apply overlay information on blocks but I don't think it's possible on individual figures. Of course, you could put each figure in a separate block. The following/attached will match what you had originally.

Option specification (BEAMER_opt property)

Insert optional arguments for the current `frame` environment using the `BEAMER_OPT` property.
XXX or block? See
http://orgmode.org/manual/Beamer-export.html.
I'd still like to see something more like a "for-dummies" explanation of passing options and arguments to LaTeX entities. I'm not saying the documentation is woefully inadequate (hardly that – Suvayu's page got me rather far, and I got stuck on a couple of details). My experience was: it never would have occurred to me on my own to use the headline text for LaTeX code, and if there was a hint anywhere in the docs to suggest that this would be the way to go, I didn't find it. That's a conceptual leap that passed me by.

- This is for frames, and for environments within a frame
- It specifies options for the current frame or block, and

Column specification (BEAMER_COL property)

Splitting a frame into multiple columns

To get multiple columns in a frame:

1. Press `C-c C-b |` (`BMCOL`) on the headlines (inside the frame) which will become columns
   The headline of column environments won't be outputted in the PDF file.
2. Specify the column width as a percentage of `\textwidth`
   !CAUTION! No absolute width, such as 4cm, which wouldn't be correctly translated...

Instead of `block`, those structural environments will become `column` (with the width parameter as a factor of `\textwidth`). Consecutive `column` environments will be put in a `columns` environment.

Two lines.

One line (but aligned).

Multiple columns

column

columns

Frame structure (Explicit page breaking) I

If the text does not fit on a single slide, all you have to do to automatically break up the frame into several frames, is set the option `allowframebreaks`.

- For allowing frame breaks on a frame by frame basis[1]

```
** A very long "frame" with breaks
   :PROPERTIES:
   :BEAMER_opt: allowframebreaks,label=
   :END:
```

  XXX This property shouldn't be interpreted for the current slide!

- For allowing frame breaks for the whole document[2]

```
#+BIND: org-beamer-frame-default-options "allowframebreaks"
```

[1] Until the Beamer issue #265 is solved, we need to unset the `framelabel` as shown above (`label=`).

[2] Till Tantau (author of Beamer) was really not keen on the idea of setting the auto-breaking frames option globally.

Vertical alignment

You can specify *top* vertical alignment globally by the `t` class option:

```
#+LaTeX_CLASS_OPTIONS: [t]
```

For single frames, you can use the same option locally:

```
* Vertically top-aligned
  :PROPERTIES:
  :BEAMER_opt: t
  :END:

...
```

You can add that special property by editing the `Opt` column within the "column view" (first press `C-c C-x C-c`) .

Result of an evaluation on two columns

Balancing text in columns.

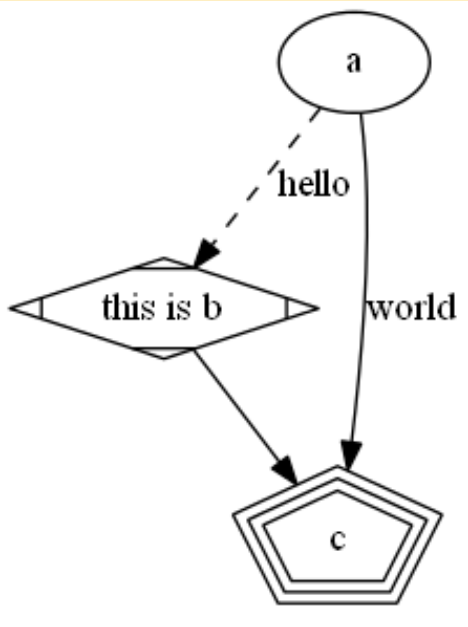
. . . a fancy verbatim block ea commodo consequat.

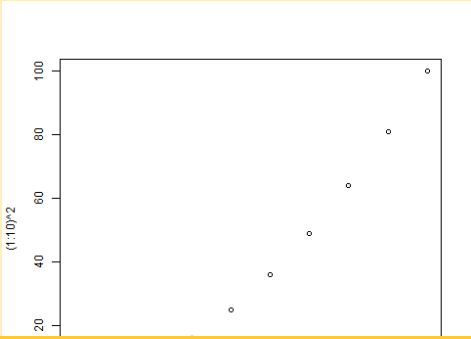
. . .

Using graphics

How to center pictures horizontally?

Figures





Absolute positioning

You can also place the logo on an absolute position of the titlepage using `tikz` or `textpos`.
Note – `textpos` is incompatible to `pgfpages`, even though it is mentioned in the beamer userguide as the way to go for absolute positioning.
Here an example using `tikz`:

```
\XXXbegin{frame}
   \tikz [remember picture,overlay]
    \node at
        ([yshift=3cm]current page.south)
        %or: (current page.center)
        {\includegraphics[width=\textwidth,height=.5\textheight]{someimage}};
   \titlepage
\XXXend{frame}
```

More on Org: Exporting a subtree

Skip proof

nil

Summary

For further reading


A. Salomaa.
Formal Languages.
Academic Press, 1973.

For further reading

A. Salomaa.
Formal Languages.
Academic Press, 1973.

E. Dijkstra.
Smoothsort, an alternative for sorting in situ.
Science of Computer Programming, 1(3):223–233, 1982.

For further reading

A. Salomaa.
Formal Languages.
Academic Press, 1973.

E. Dijkstra.
Smoothsort, an alternative for sorting in situ.
Science of Computer Programming, 1(3):223–233, 1982.

E. Feldman and J. Owings, Jr.
A class of universal linear bounded automata.
Information Sciences, 6:187–190, 1973.

For further reading

A. Salomaa.
Formal Languages.
Academic Press, 1973.

E. Dijkstra.
Smoothsort, an alternative for sorting in situ.
Science of Computer Programming, 1(3):223–233, 1982.

E. Feldman and J. Owings, Jr.
A class of universal linear bounded automata.
Information Sciences, 6:187–190, 1973.

P. Jančar, F. Mráz, M. Plátek, and J. Vogel.
Restarting automata.
FCT Conference 1995, LNCS 985, pages 282–292. 1995.

Proof details

Text omitted in main talk.

More details

Even more additional material.

Abbreviations