

# beamer-rl class

Salim Bou

Repository: <https://github.com/seloumi/beamer-rl>  
Bug tracker: <https://github.com/seloumi/beamer-rl/issues>

v2 - 2025/10/15

- 1 Introduction
- 2 How to use beamer-rl
- 3 Some notes
- 4 pgfpages-rl package
- 5 Examples
  - Blocks
  - Lists
  - Hyperlinks
  - Theorems
  - Zooming

Creating beamer presentation for languages with script from right to left (like arabic) using pdf $\text{\TeX}$  or X $\text{\TeX}$  still poses many problems due to bugs not currently resolved especially for colors.

The Lua $\text{\TeX}$  team set solutions for these issues thanks to them and to *Javier Bezos* for his works on the package `babel` and `bidir` writing

This class provides patches of some beamer templates and commands to create right to left beamer presentation, the class call `babel` with `bidir=basic` option and require Lua $\text{\TeX}$  engine

# How to use beamer-rl I

```
\documentclass{beamer-rl}  
  
% import language  
\babelprovide[import=ar-DZ, main]{arabic}  
  
\usetheme{Madrid}  
  
\begin{document}  
...  
\end{document}
```

We get a similar result by adding the main language of the presentation (language with right-to-left script) as option of class as follows:

# How to use beamer-rl II

```
\documentclass[arabic]{beamer-rl}  
  
\usetheme{CambridgeUS}  
  
\begin{document}  
...  
\end{document}
```

We can also add more language options that the command `\babelprovide` provides as follows:

```
\documentclass[arabic={mapdigits}]{beamer-rl}  
  
% equivalent to  
% \babelprovide[import,main,mapdigits]{arabic}
```

# How to use beamer-rl III

The class define in the same way as options (languages supported by the package babel with script from right to left)

arabic	arabic-ps	pashto
arabic-dz	arabic-jo	persian
arabic-tn	centralkurdish	punjabi-arab
arabic-ma	hebrew	syriac
arabic-eg	kashmiri	urdu
arabic-sa	mazanderani	uyghur
arabic-iq	malayalam	uzbek-arab
arabic-sy	northernkurdish-	yiddish
arabic-lb	arab	

# Some notes I

- The class define Amiri as default sans serif font, we can modify this in the preamble with

```
\babelfont{sf}{<font name>}
```

- The class defines option layout which passes its content to babel

```
\documentclass[layout={<babel layout>}]{beamer-rl}
```

More on the subject can be found in the manual of babel package [▶ link](#)

- The beamer-rl class swap the definition of `\blacktriangleright` with `\blacktriangleleft` in RTL context

	<code>\blacktriangleright</code>	<code>\blacktriangleleft</code>
LTR context	▶	◀
RTL context	▶	◀

- In some cases you need to use `\babelsublr` command from `babel` package to insert a left to right text within your right to left text, e.g if you need to insert a `pspicture` drawing in RTL context

```
\babelsublr{LTR context ... }
```

# pgfpages-rl package

pgfpages-rl adds to pgfpages the ability to support TRT pagedir, the package requires Lua<sup>A</sup>T<sub>E</sub>X engine. It can also be used with other document classes besides beamer-rl

```
\documentclass{beamer-rl}  
\babelprovide[import=ar-DZ, main]{arabic}  
\usetheme{Warsaw}  
\usepackage{pgfpages-rl} % adapt pgfpages to TRT pagedir  
\setbeamertemplate{note page}[]  
\setbeameroption{show notes on second screen=right}  
\begin{document}  
...  
\end{document}
```

# Examples

```
\setbeamertemplate{blocks}[default]
```

## Lorem

On 21 April 1820, during a lecture, Ørsted noticed a compass needle deflected from magnetic north when an electric current from a battery was switched on and off.

```
\setbeamertemplate{blocks}[rounded] [shadow=true]
```

## Lorem

On 21 April 1820, during a lecture, Ørsted noticed a compass needle deflected from magnetic north when an electric current from a battery was switched on and off.

# enumerate, itemize I

```
\setbeamertemplate{enumerate item}[ball]  
\begin{enumerate}  
\item First  
\item Second  
\end{enumerate}
```

First ❶  
Second ❷

```
% in RTL context  
\setbeamertemplate{itemize item}[triangle]  
\begin{itemize}  
\item First  
\item Second  
\end{itemize}
```

First ◀  
Second ▶

# enumerate, itemize II

- ▶ First
  - ▶ Second
- ```
% in LTR context
\setbeamertemplate{itemize item}[triangle]
\begin{itemize}
\item First
\item Second
\end{itemize}
```

.First ●

.Second ●

return to first slide ◀

```
\hyperlink{jumptofirst}  
{\beamergotobutton{return to first slide}}  
\hypertarget<1>{jumptofirst}{}  

```

.First ●

.Second ●

[return to first slide ◀](#)

```
\hyperlink{jumptofirst}  
{\beamergotobutton{return to first slide}}  
\hypertarget<1>{jumptofirst}{}  

```

# Theorems

The proof uses *reductio ad absurdum*.

نظرية

.There is no largest prime number

برهان.

.were the largest prime number  $p$  Suppose ①

.numbers  $p$  be the product of the first  $q$  Let ②

.is not divisible by any of them  $q + 1$  Then ③

thus divisible by some prime number not in ,1 is greater than  $q + 1$  But ④



.numbers  $p$  the first

# Theorems

The proof uses *reductio ad absurdum*.

نظرية

.There is no largest prime number

برهان.

.were the largest prime number  $p$  Suppose ①

.numbers  $p$  be the product of the first  $q$  Let ②

.is not divisible by any of them  $q + 1$  Then ③

thus divisible by some prime number not in ,1 is greater than  $q + 1$  But ④



.numbers  $p$  the first

# Theorems

The proof uses *reductio ad absurdum*.

نظرية

.There is no largest prime number

برهان.

.were the largest prime number  $p$  Suppose ①

.numbers  $p$  be the product of the first  $q$  Let ②

.is not divisible by any of them  $q + 1$  Then ③

thus divisible by some prime number not in ,1 is greater than  $q + 1$  But ④



.numbers  $p$  the first

# Theorems

The proof uses *reductio ad absurdum*.

نظرية

.There is no largest prime number

برهان.

.were the largest prime number  $p$  Suppose ①

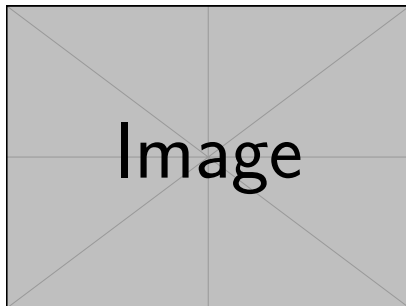
.numbers  $p$  be the product of the first  $q$  Let ②

.is not divisible by any of them  $q + 1$  Then ③

thus divisible by some prime number not in ,1 is greater than  $q + 1$  But ④

.numbers  $p$  the first





```
\framezoom<1><2>[border=2](1cm,1cm)(2cm,2cm)  
% (1cm,1cm)=(<upper right x>,<upper right y>)  
% (2cm,2cm)=(<zoom area width>,<zoom area depth>)  
\pgfimage[height=5cm]{example-image}
```



# Image