Simple Slides
A Context presentation module

1 Introduction

This module provides an easy-to-use interface for creating simple slides/presentations in Context. The salient features of this module are:

- The module is meant for presentations which will be shown on a digital projector. They have no interactive elements (such as buttons or hyperlinks) and no navigational tools (such as table of contents).
- The module comes with several predefined styles; these styles are sober in appearance and meant for academic presentations. It also provides some macros to help in presenting slides with both pictures and text.
- Most styles allow for some degree of user-reconfigurability. Designing a new style is also easy.

This module provides a simple structure that will be suitable for beginning or intermediate users of Context, or someone who does not want to spend too much time playing around with different configuration options for Context. As such it focusses on different users than Hans's presentation modules that provide more and fancier features. This module also offers much less features than the Latex `beamer` package. Its main strength is its ease of use; you should be able to write your first presentation after spending five minutes with this manual.

2 A bit of history

The idea of a module suitable for simple presentations took shape when Thomas started using Context for preparing his course presentations. Context comes with a bunch of modules for presentations (the files `s-pre-??.tex` in `$TEXMF/tex/context/base`) which are written by Hans Hagen. Hans usually creates a new presentation style whenever he

gives a talk about Context. As such, his presentation styles highlight the fancy and bleeding edge features of Context, and are not the most suitable starting point for academic presentations.

Context does make creating your own presentation style relatively easy. So Thomas wrote some presentation related macros (see the PracTex article http://www.tug.org/pracjourn/2006-2/schmitz/). With time, he extended these macros into a collection of styles providing different visual effects, and later collected all of them in the `taspresent` module. He gave a talk about the `taspresent` module at the second Context user meeting at Bohinj, and in the ensuing discussions, Aditya and Thomas decided to modularize and "Contextize" some of the internals of the module, giving rise to the current module. Most of the code in the current release has been contributed by Aditya.

3 Installation

The module is installed in the usual way: simply unzip the archive `t-simpleslides-<date>.zip` into one of your `$TEXMF` trees, and from a terminal run `mktexlsr` (for MkII) and `mtxrun --generate` (for MkIV).

To verify that everything was installed correctly, run `kpsewhich t-simpleslides.tex` from a terminal (for MkII) and `mtxrun --locate t-simpleslides.tex` (for MkIV); these commands should return the complete path of the files that you just installed.

A note about Tex-engines

We have extensively tested this module with pdftex and Luatex (that is, with MkII and MkIV). In spite of our best efforts, we have not been able to get this module to work reliably with Xetex. If you are a Xetex guru, and know how to fix some of the errors with Xetex, we will appreciate the help.

4 Quick start

First you must tell Context that you want to use this module. To do this simply write:

```
\usemodule[simpleslides]
```

The module sets the paper size and font sizes to values that are suitable for presentations. Everything else is left like a default Context document. The module comes with the following styles that change the visual appearance of the presentation.

- BigNumber
- BottomSquares
- Boxed
- Ellipse
- Embossed
- Framed
- FramedTitle
- HorizontalStripes
- NarrowStripes
- PlainCounter
- RainbowStripe
- Rounded
- Shaded
- SideSquares
- SideToc
- Split
- Sunrise
- Swoosh
- ThickStripes

To use a style, say `BigNumber`, pass the `style=BigNumber` option to the `simpleslides` module:

```
\usemodule[simpleslides]
          [style=BigNumber]
```

Some of the styles have a few variants that can be chosen using `color` and `alternative` keys. These are explained in Section 6.

By default, the Latin Modern Sans font is used. The module makes it easy to use other fonts that come with a typical Tex distribution. The following fonts are provided:

- `LatinModern`
- `LatinModernSans`
- `Bookman`
- `Chancery`
- `Gothic`
- `Helvetica`
- `Palatino`
- `Schoolbook`
- `Times`

To choose a font, say `Helvetica`, pass `font=Helvetica` option to the `simpleslides` module as follows.

```
\usemodule[simpleslides]
          [style=BigNumber,
            font=Helvetica]
```

The default font size is 17pt. Font size can be changed using the `size` key.

More details about the fonts, including information on how to use your own fonts is given in Section 7.

The complete setup for using this module is

```
\usemodule [...,...] [..,..=..,..]

1 simpleslides
2 style       = BigNumber BottomSquares Boxed Ellipse Embossed Framed FramedTitle
                HorizontalStripes NarrowStripes PlainCounter RainbowStripe Rounded
                Shaded SideSquares SideToc Split Sunrise Swoosh ThickStripes
  size        = DIMENSION
  font        = LatinModern LatinModernSans Bookman Chancery Gothic Helvetica Palatino
                Schoolbook Times
  color       = red blue green bluered
  alternative = square stripe
```

Structure of a presentation

The `simpleslides` module has a very simple model of a presentation. A presentation consists of a title followed by a series of slides; the module provides macros to help create

a presentation title page and slides. A minimal presentation is shown below. The result is shown in Figure 1.

```
\usemodule[simpleslides]
          [style=BigNumber]

\setupTitle
  [ title={Title of the presentation},
   author={Name of authors},
     date={Date of presentation},
  ]

\starttext
\placeTitle

\SlideTitle {The first slide}
Write whatever you want.

\SlideTitle {The second slide}
Continue writing.

\stoptext
```

Presentation title page

A presentation title page displays the title of the presentation, the names of the authors, and the date. These can be specified using **\setupTitle** as follows:

```
\setupTitle
  [ title={Title of the presentation},
   author={Name of authors},
     date={Date of presentation},
  ]
```

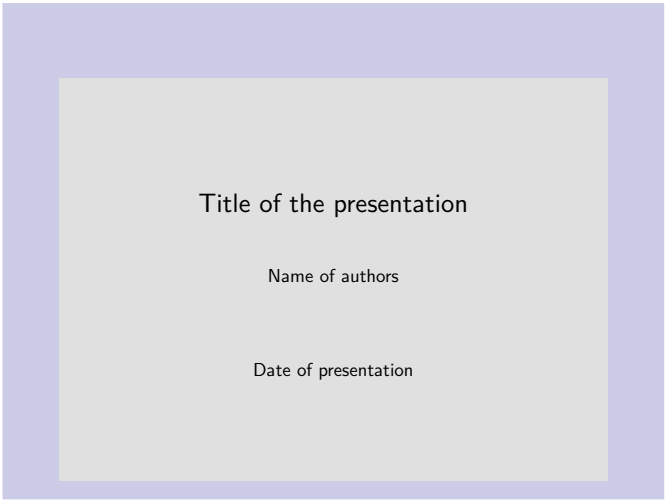
The macro **\placeTitle** places the title page in the presentation. It is possible to change the look of **\placeTitle** using some additional arguments to **\setupTitle**. These are explained in Section 8.

Presentation slide

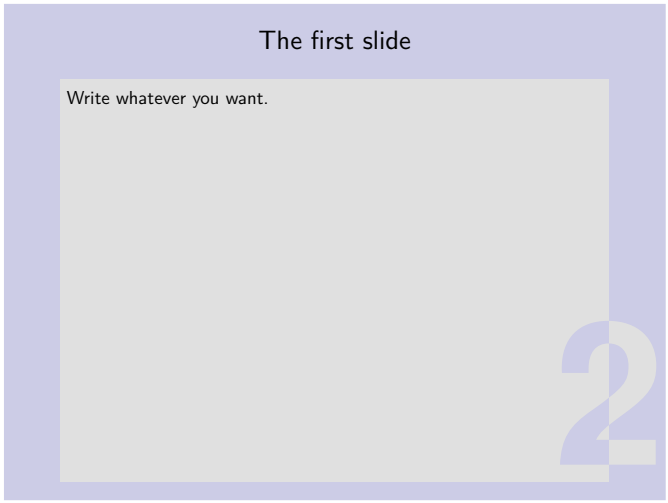
The `simpleslides` module provides a **\SlideTitle** macro, which starts a new slide (basically a new page), and typesets its argument as the title of the slide. It also takes care of increasing the page counters and progress bars, and setting up the background. The content of the slides follows after this command.

```
\usemodule[simpleslides]
          [style=BigNumber]
\setupTitle[...]

\starttext
\SlideTitle{...}
...
\SlideTitle{...}
...
\stoptext
```

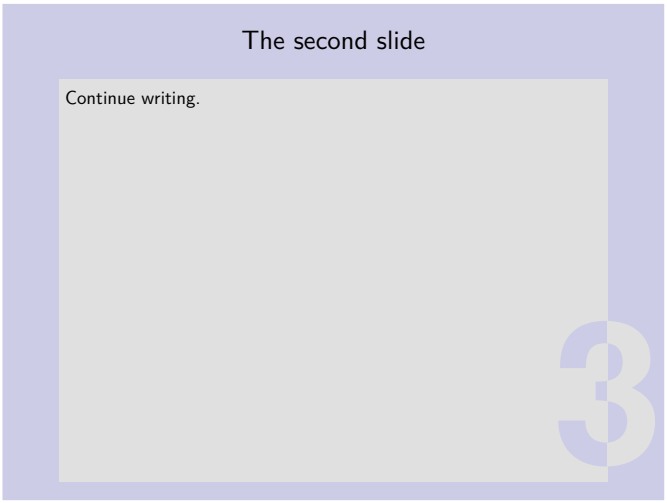
A minimal example



Title page



First slide



Second slide

Figure 1 A minimal presentation

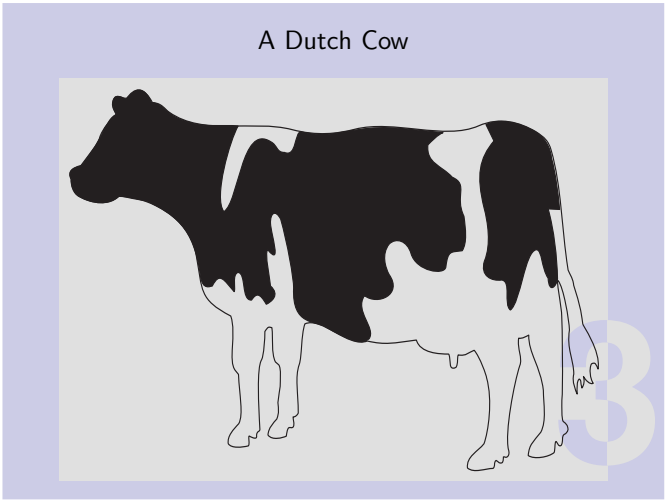
A slide is a normal Context page, so you can use any command or environment that you want. Each presentation style sets up a style for itemizations, and provides useful macros for placing pictures. These macros will be explained later.

5 Placing pictures

If you want to place pictures in your slides, you can always use Context's `\externalfigure` macro. This module also provides a macro, `\IncludePicture`, for preconfigured picture layouts. Two layouts are provided:

- `horizontal`: the picture is placed under the title of the slide, so that it fits in the available space.
- `vertical`: the slide is divided into two columns; the picture is placed on the left column and text is placed on the right column.

```
\usemodule[simpleslides]
          [...]
\starttext
...
\IncludePicture
    [horizontal]
    [cow]
    {A Dutch Cow}
...
\stoptext
```



A horizontal picture

A horizontal picture

```
\usemodule[simpleslides]
          [...]
\starttext
...
\IncludePicture
    [vertical]
    [mill]
    {The windmills are an example of
a green energy source}
...
\stoptext
```

A vertical picture

A vertical picture

Figure 2 Example of `horizontal` and `vertical` options for `IncludePicture` macro

These layouts are shown in Figure 2.

A horizontal picture is placed as follows:

```
\IncludePicture
  [horizontal]
  [filename]  % Name of the file that contains the picture
  {Title of the slide}
```

while a vertical picture is placed as follows:

```
\IncludePicture
  [vertical]
  [filename]  % Name of the file that contains the picture
```

```
{Text that is placed on the right of the picture}
```

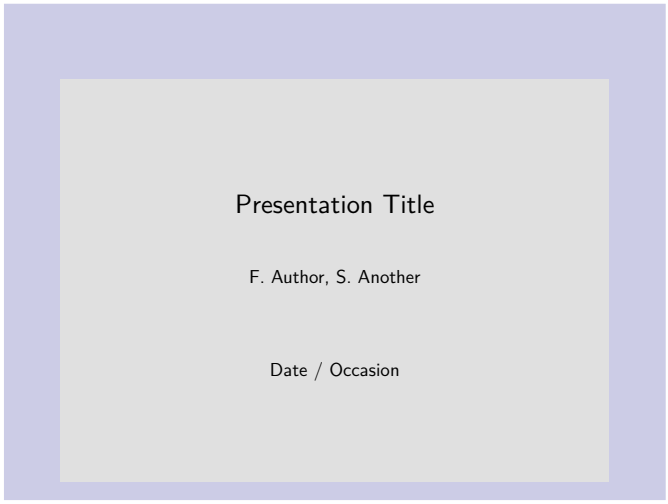
It is possible to change the height and width of the pictures, or highlight them with circles and arrows. These details can be found in Section 10

6 Changing presentation styles

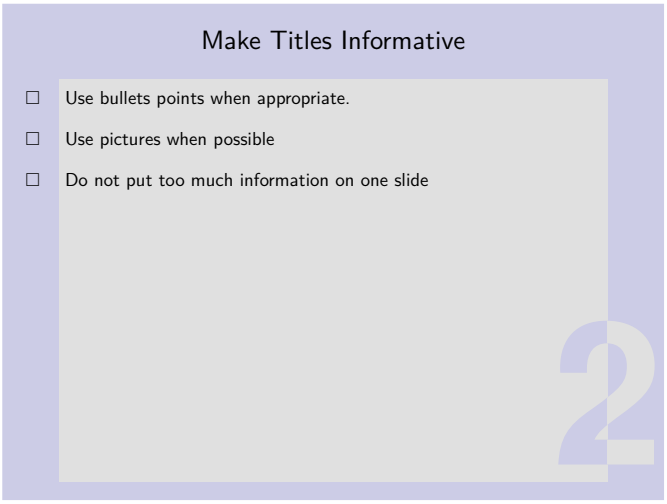
The `style` key to `\setupmodule[simpleslides]` determines the look of the presentation. Some of the styles come with variants, that can be chosen using `color` and `alternative` keys. The available styles are shown below along with the details of their variants.

BigNumber: with color=blue (also accepts color=red)

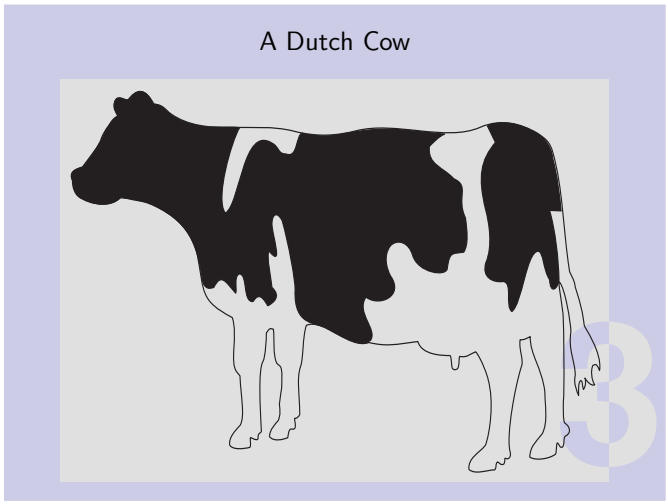
This is a style with subdued and quiet colors; its characteristic feature is the page number on the lower right border of the text area. This detail was inspired by the *split* style (`s-pre-14`) by Hans.



Title Page



Normal Slide



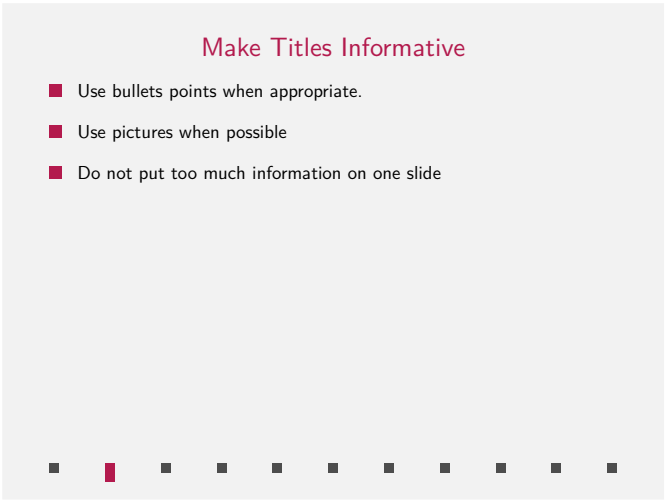
Horizontal Picture

Vertical Picture

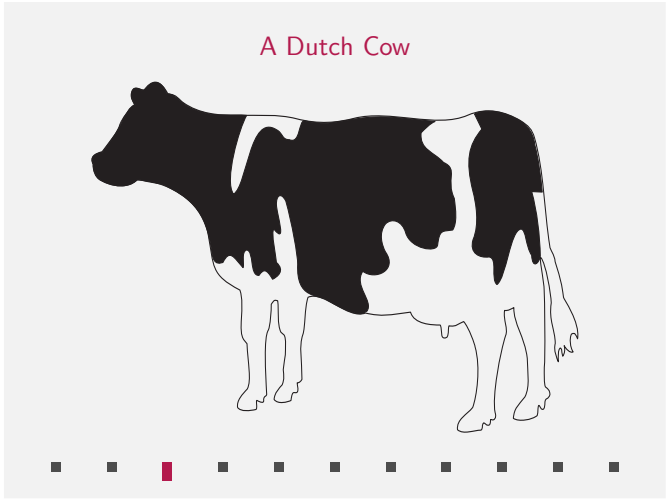
BottomSquares

This minimalistic style is inspired by a presentation Taco gave at EuroTeX 2006.

Title Page



Normal Slide

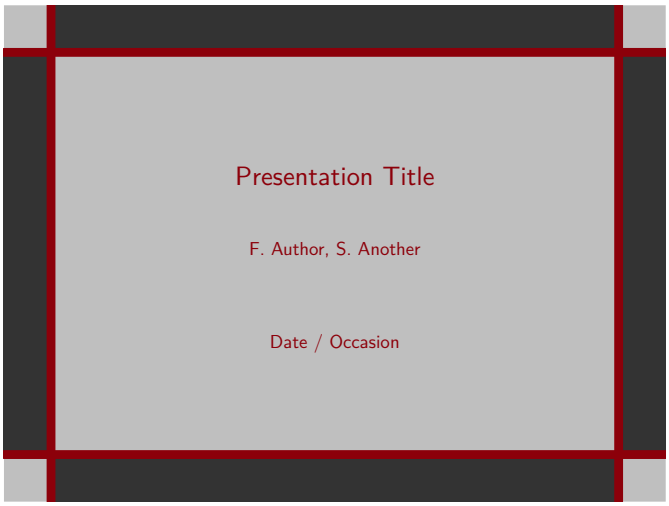


Horizontal Picture

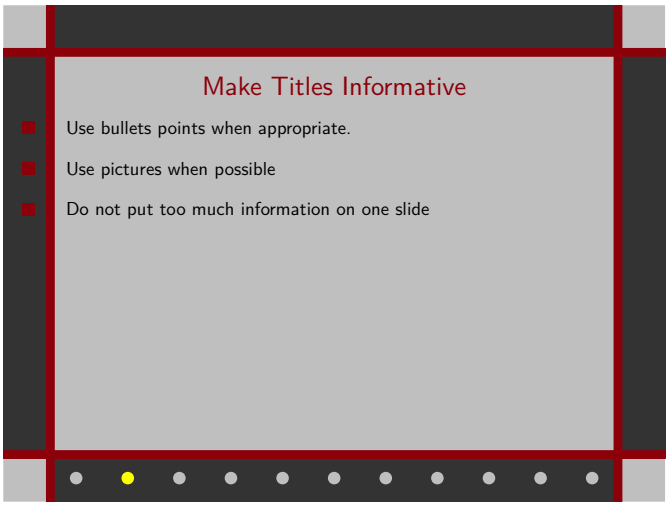
Vertical Picture

Boxed

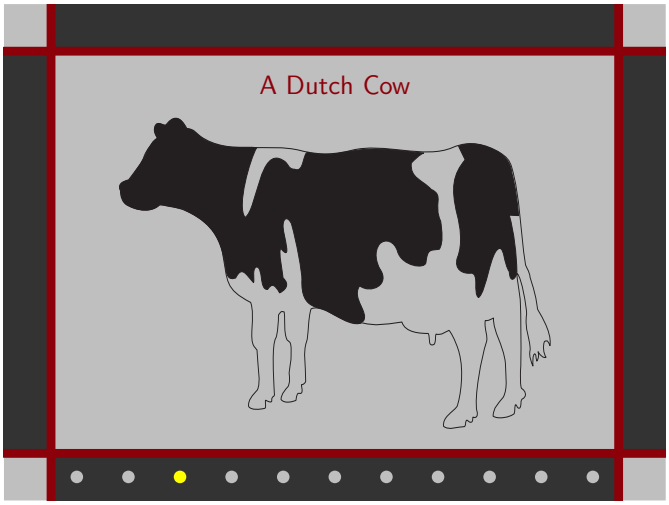
This style is inspired by the screen version of the Metafun manual. Watch the small gray circles at the bottom!



Title Page



Normal Slide

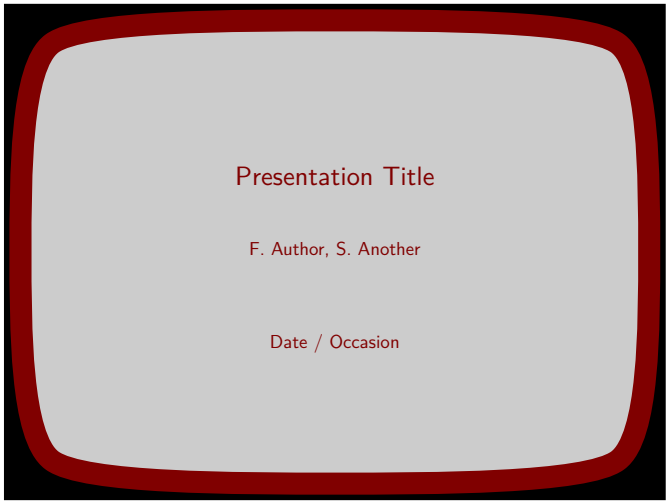


Horizontal Picture

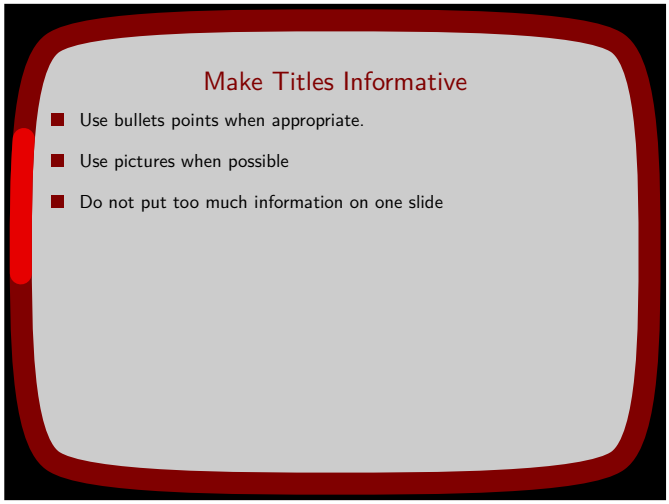
Vertical Picture

Ellipse

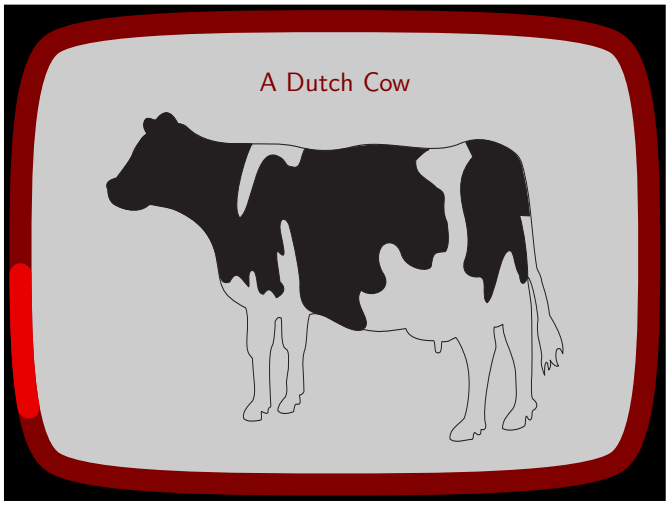
This style is inspired by *funny* style (`s-pre-03`) by Hans. The light red stripe marks the progress.



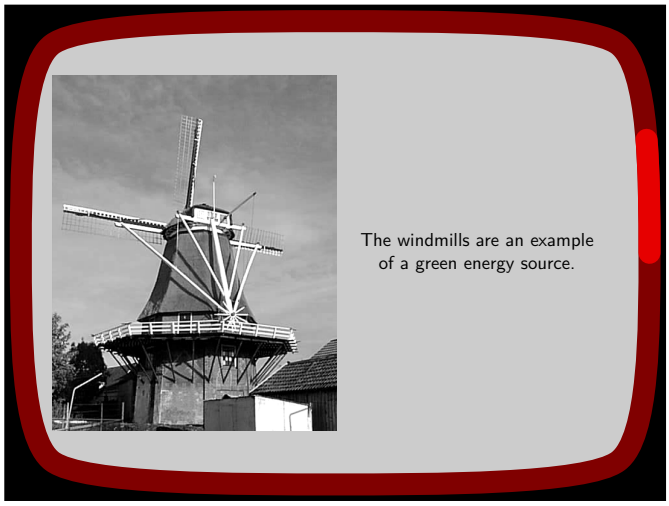
Title Page



Normal Slide



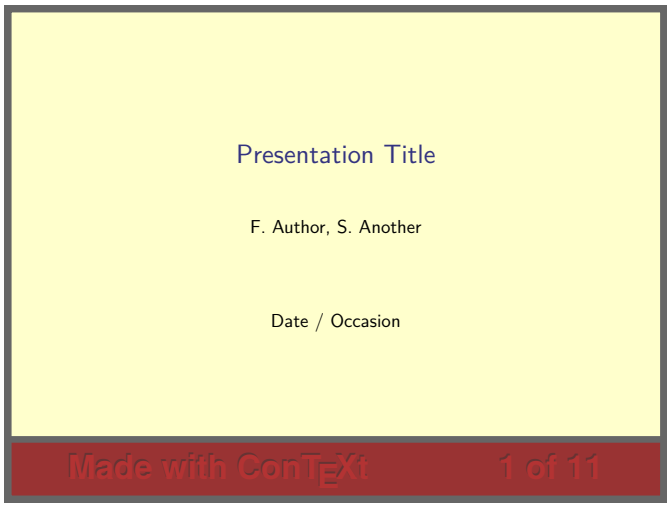
Horizontal Picture



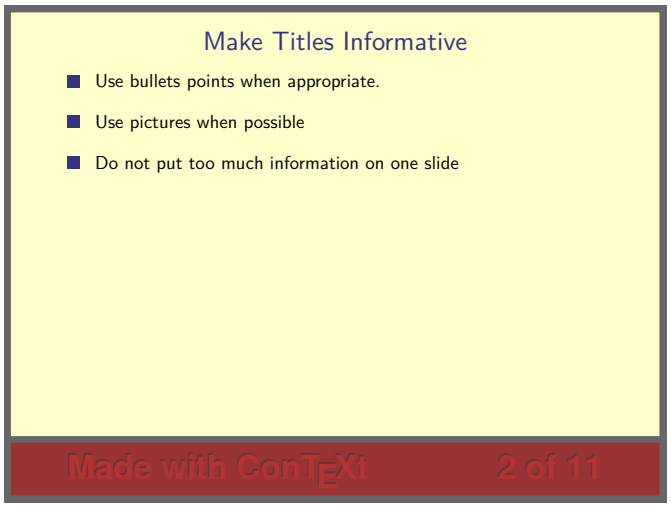
Vertical Picture

Embossed

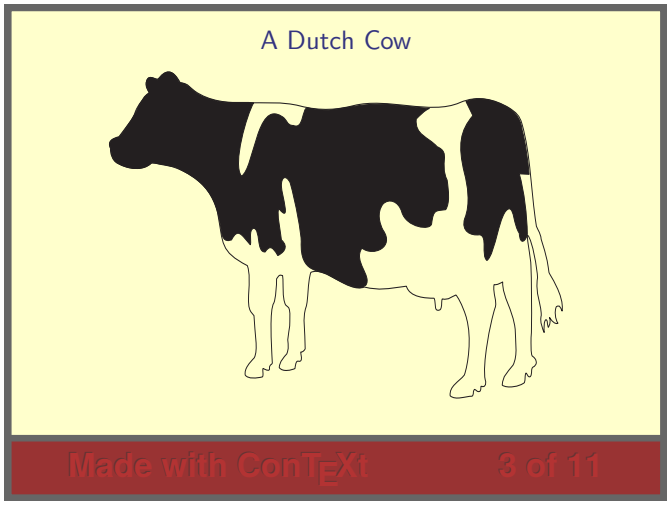
Spread the word, don't be shy! Show your pride in using Context. The color theme will probably look familiar; we copied it from the `enattab` manual.



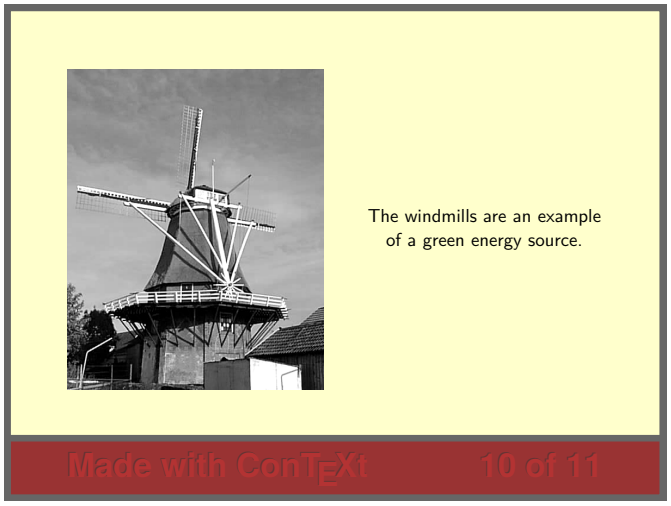
Title Page



Normal Slide



Horizontal Picture



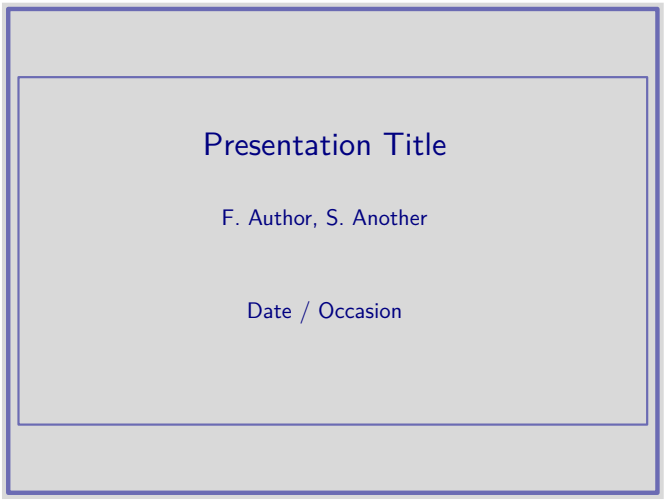
Vertical Picture

If you are shy, or narcissistic, you can change the emblem by

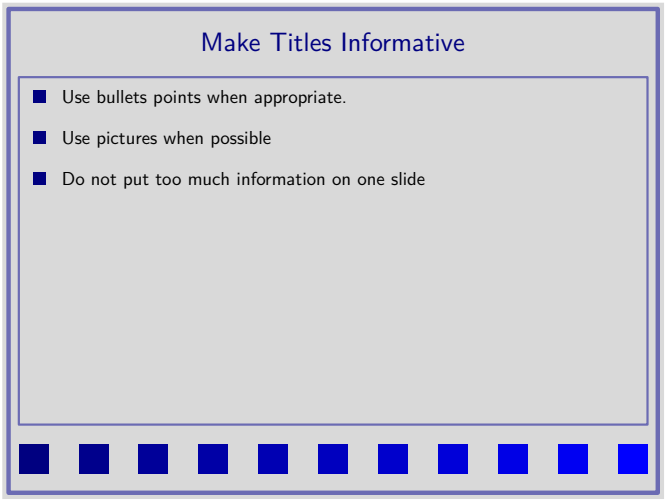
```
\setuplabeltext  [simpleslidesemblem={I made this presentation}]
```

Framed: with alternative=square

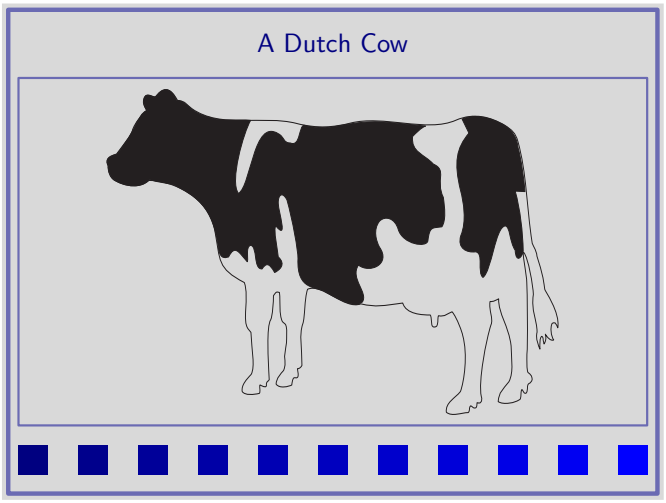
This style was inspired by the *green* style (`s-pre-02`) by Hans. It has a thick blue frame around the entire slide area and a thinner frame around the text area. The style has two options for alternative: `alternative=stripe` will display a shaded blue area which will grow with each slide; `alternative=square` displays a row of blue squares at the bottom which also measure the presentation's progress.



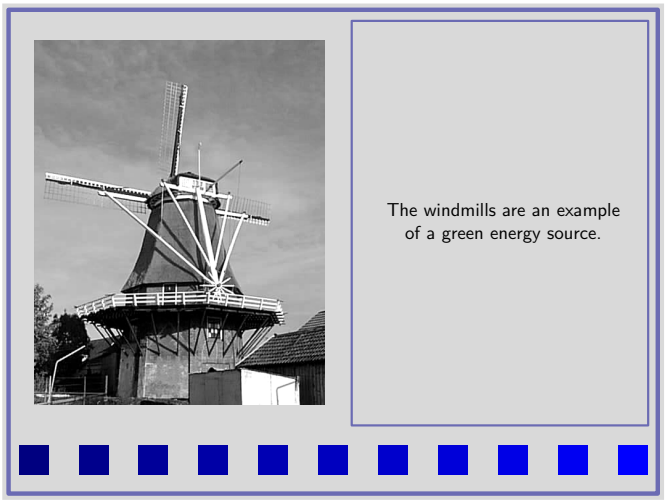
Title Page



Normal Slide

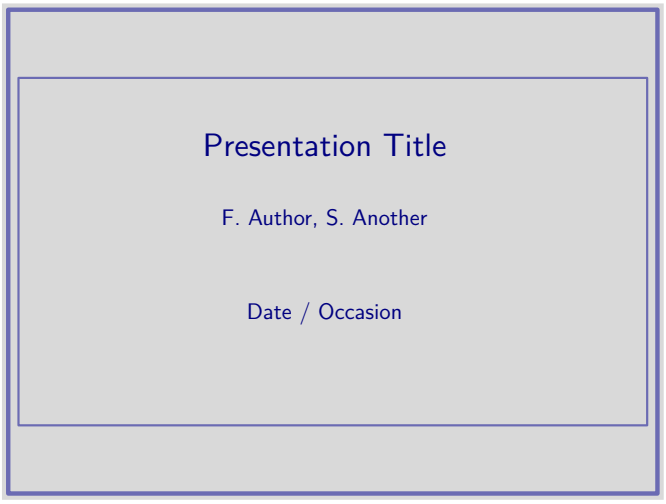


Horizontal Picture

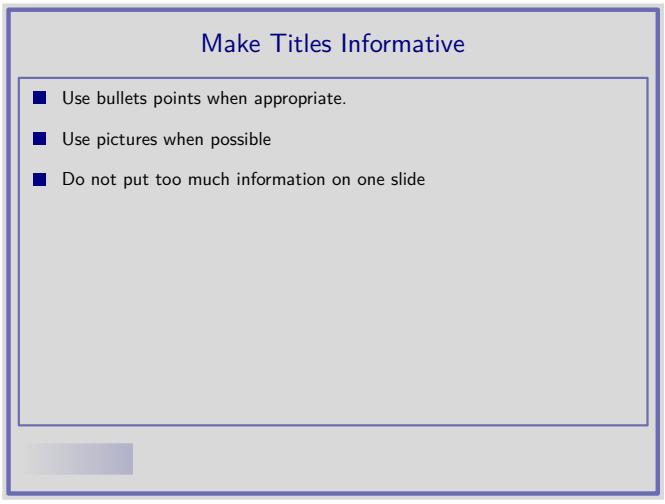


Vertical Picture

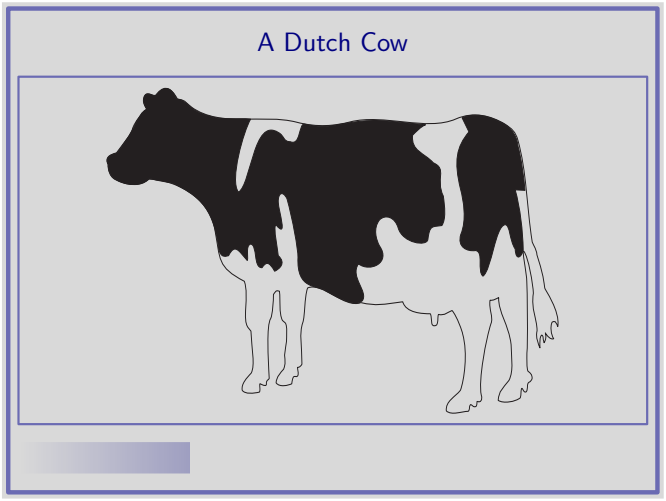
Framed: with alternative=stripe



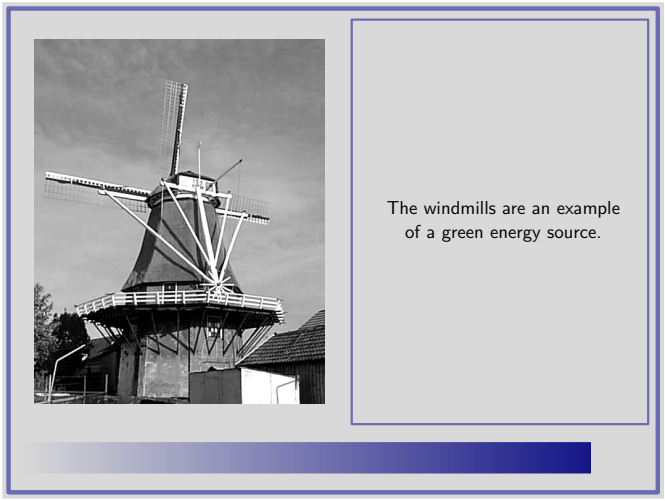
Title Page



Normal Slide



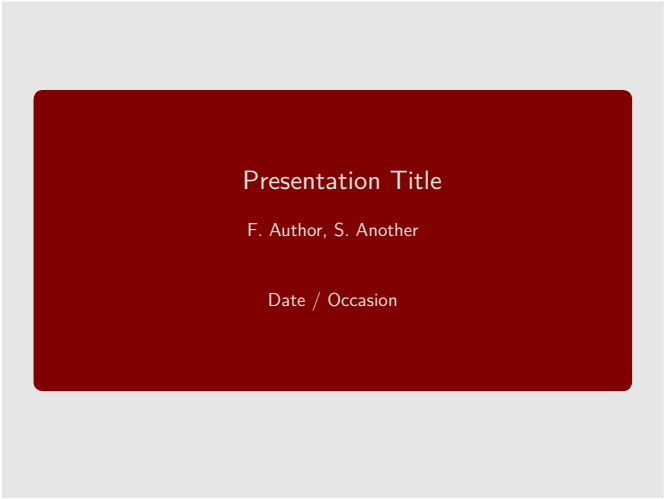
Horizontal Picture



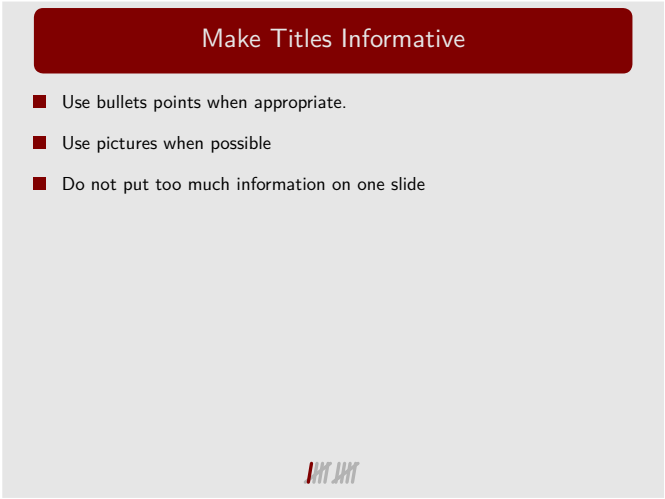
Vertical Picture

FramedTitle

This is a style with loud titles. Its characteristic feature is the *scratch counter* at the bottom, which is derived from Section 7.2 of the Metafun manual.



Title Page



Normal Slide

Horizontal Picture

Vertical Picture

HorizontalStripes: with color=green (also accepts color=blue and color=red)

A sober style with an emphasis on horizontal lines, inspired by the *Szeged* theme in Latex's `beamer` package.

Presentation Title

F. Author, S. Another

Date / Occasion

Presentation Title 1 of 11

Title Page

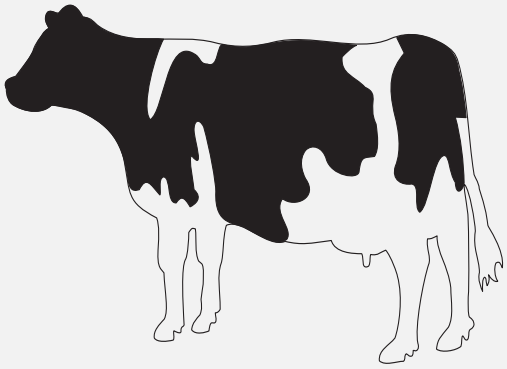
Make Titles Informative

- Use bullets points when appropriate.
- Use pictures when possible
- Do not put too much information on one slide

Presentation Title 2 of 11

Normal Slide

A Dutch Cow



Presentation Title 3 of 11

Horizontal Picture

The windmills are an example of a green energy source.

Presentation Title 10 of 11

Vertical Picture

NarrowStripes: with color=green (also accepts color=blue and color=red)

A very simple and sober style, with shaded narrow stripes.

Presentation Title

F. Author, S. Another

Date / Occasion

Title Page

2

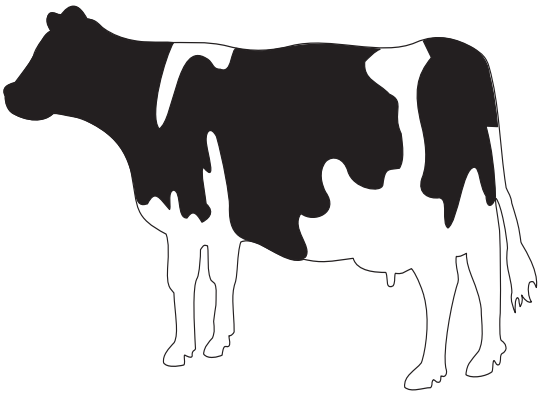
Make Titles Informative

- Use bullets points when appropriate.
- Use pictures when possible
- Do not put too much information on one slide

Normal Slide

3

A Dutch Cow



Horizontal Picture

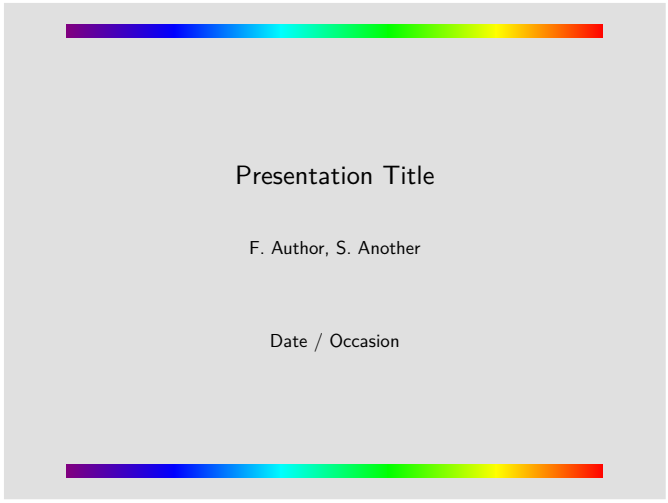
10

The windmills are an example of a green energy source.

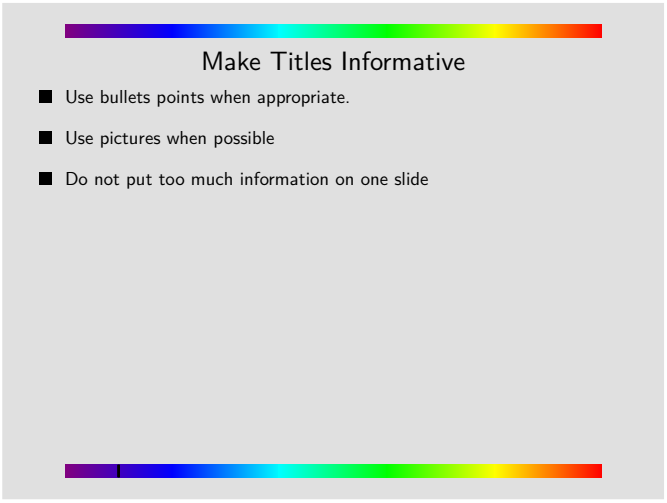
Vertical Picture

RainbowStripe

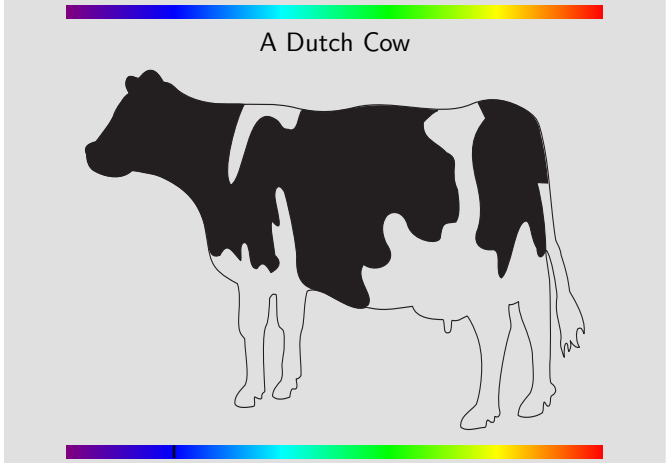
A colorful style for daring presenters. The black line which marks the progress is reminiscent of absorption lines in star spectra, so this style may be apt for astrophysical presentations?



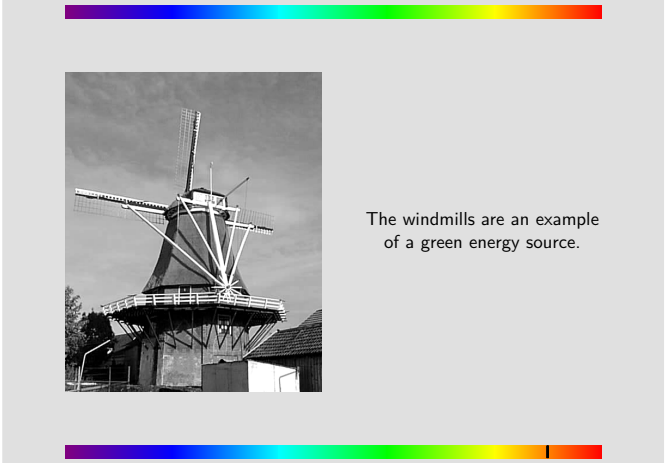
Title Page



Normal Slide



Horizontal Picture

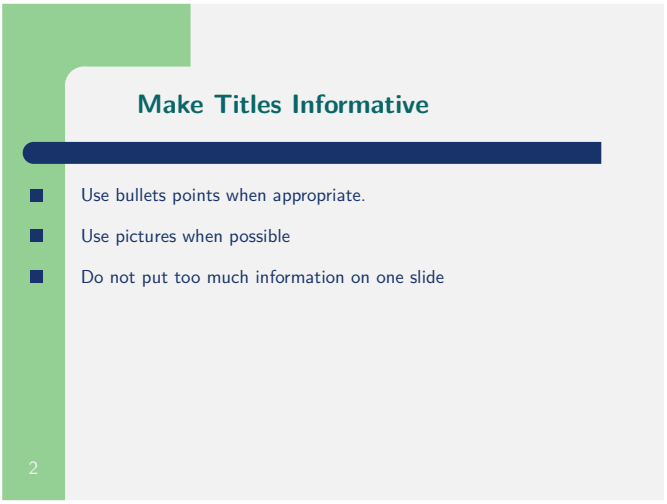


Vertical Picture

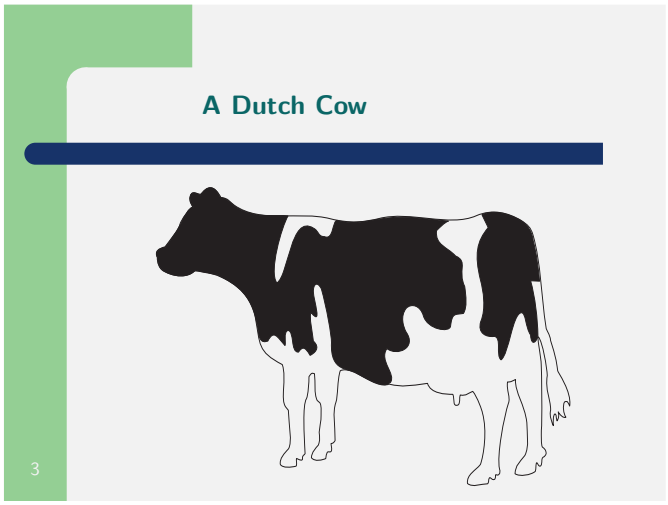
Rounded

This style has cool colors and lots of white space; it is probably best suited for presentations with relatively little text.

Title Page



Normal Slide

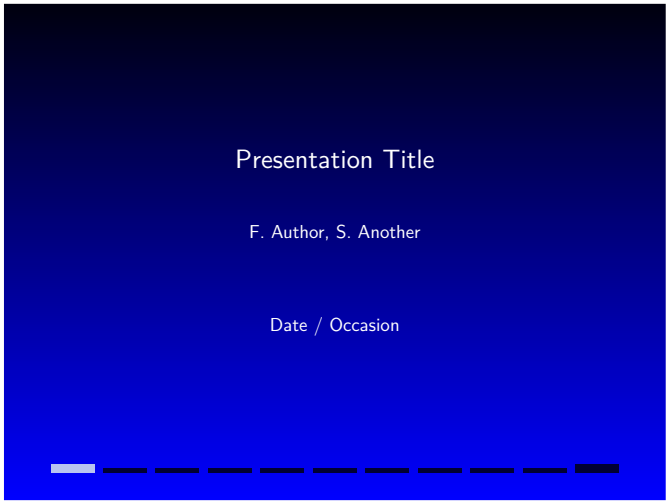


Horizontal Picture

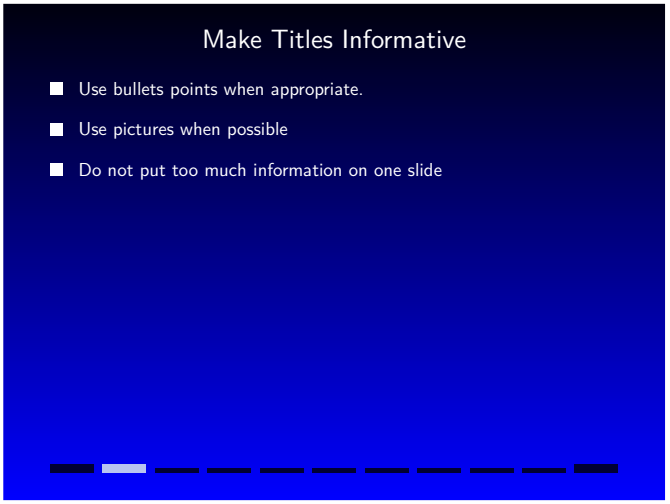
Vertical Picture

Shaded: with color=blue (also accepts color=green and color=bluered)

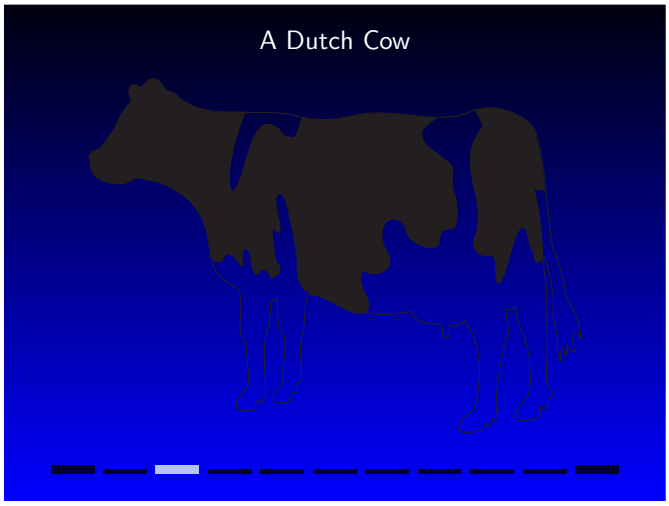
The only ornament to this style is the dark shaded background. It uses Context's `interactionbar` mechanism to show the progress of the presentation. It provides much space for text.



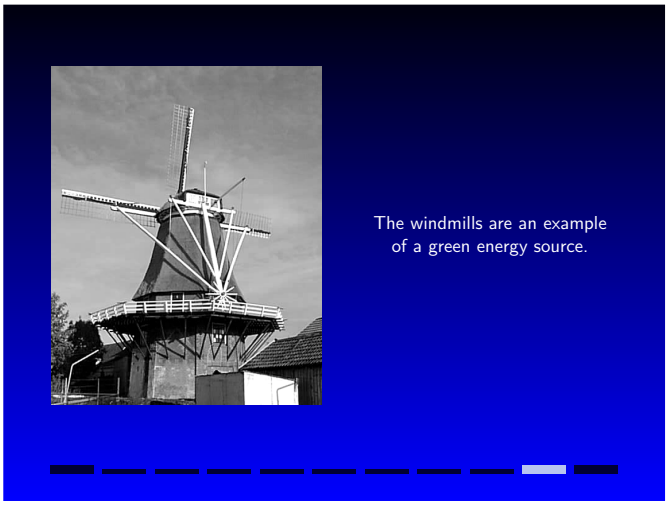
Title Page



Normal Slide



Horizontal Picture

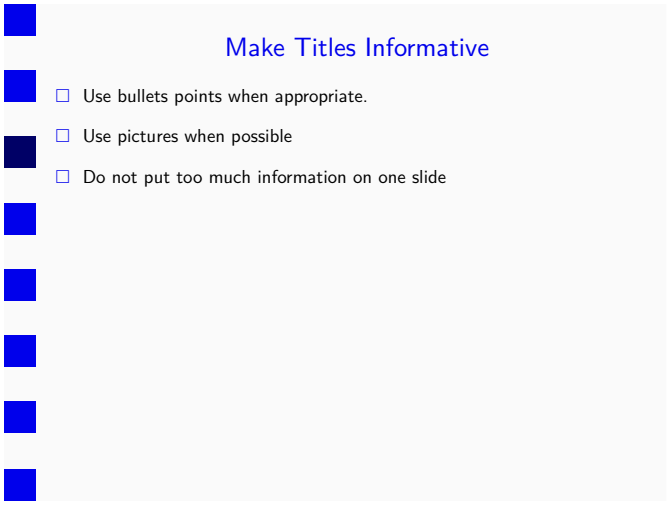


Vertical Picture

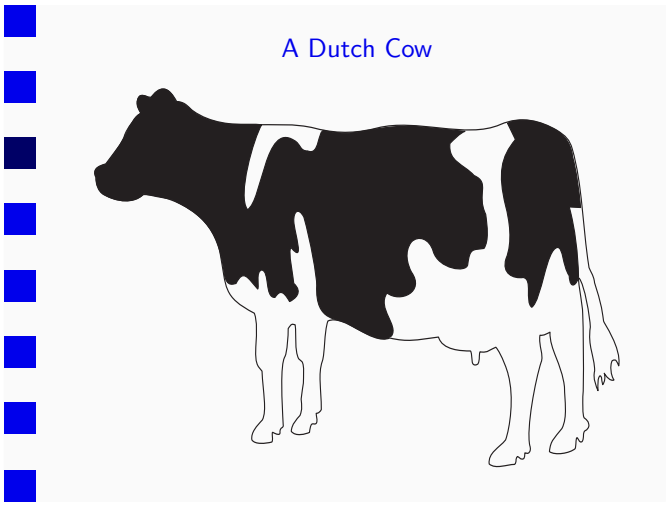
SideSquares

This style is inspired by the colors and corporate look of Thomas's university. It is very sober and offers much space for text and images. There is a rough progress meter built into the blue quadrangles.

Title Page



Normal Slide

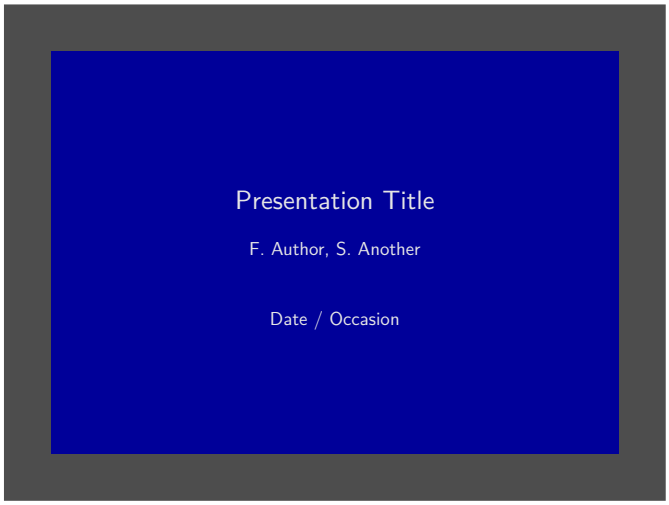


Horizontal Picture

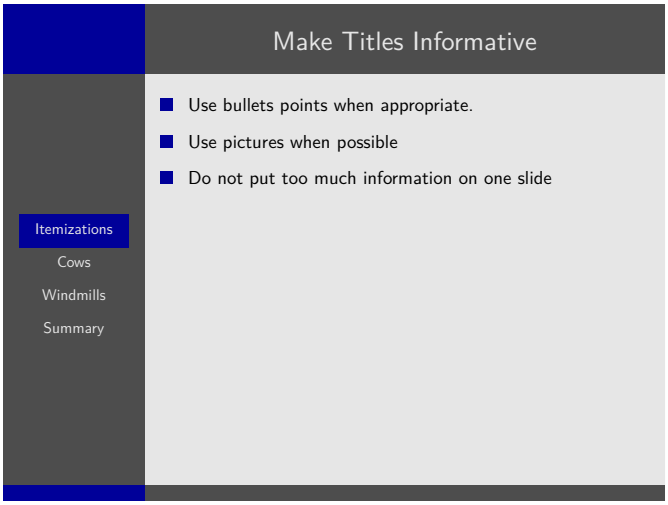
Vertical Picture

SideToc

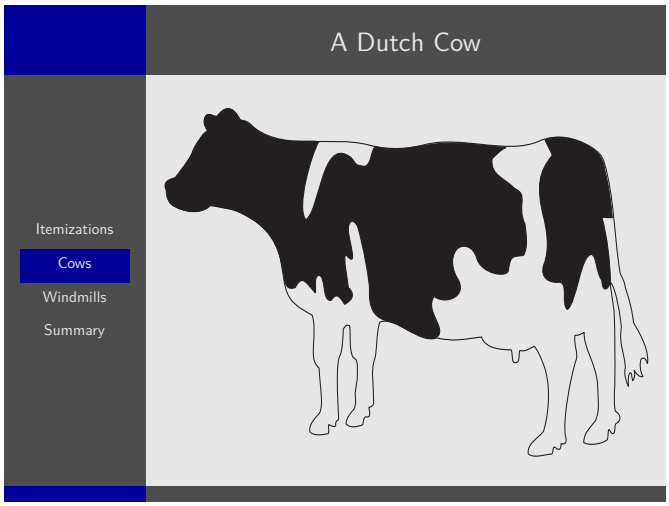
This Style has a list of Topics in its left margin; the current topic is automatically highlighted. To set a topic and add it to this table simple type `\Topic[TopicName]` in your source file where the new topic begins.



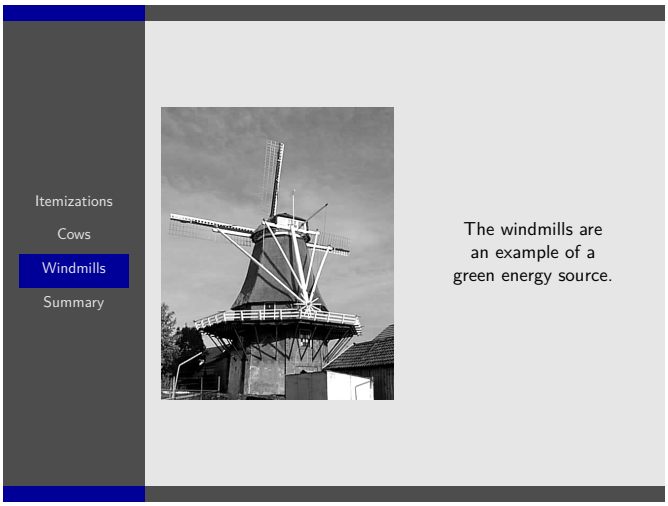
Title Page



Normal Slide



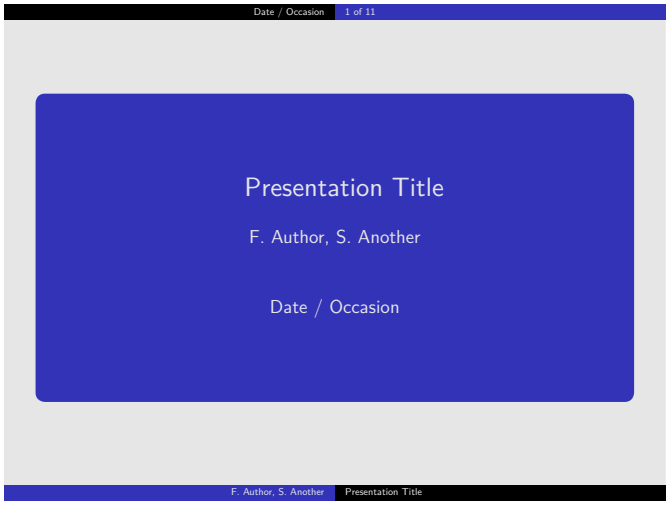
Horizontal Picture



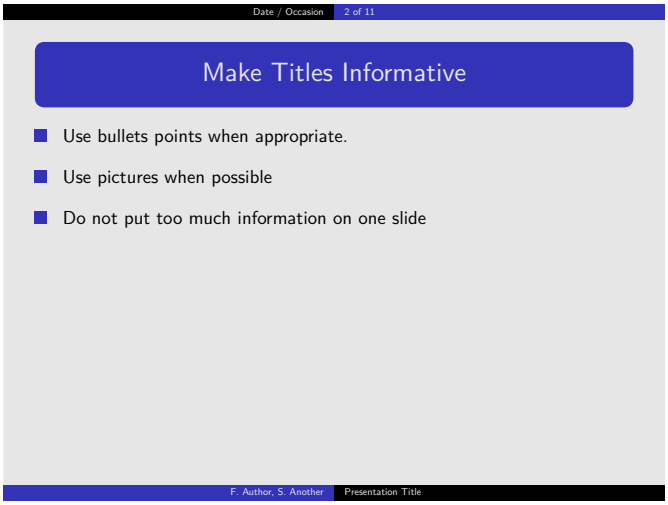
Vertical Picture

Split

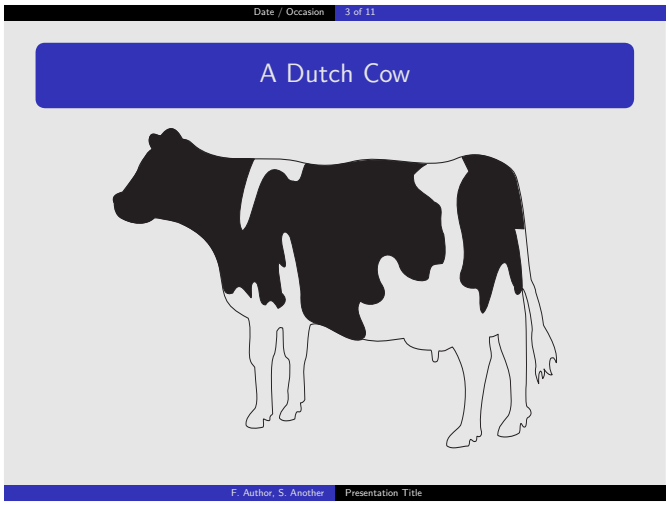
This style is inspired by the *Copenhagen* theme of the Latex's `beamer` package. The narrow blue and black stripes at the top and the bottom of the slides display the date and slide number (top) and the title and author of the presentation.



Title Page



Normal Slide

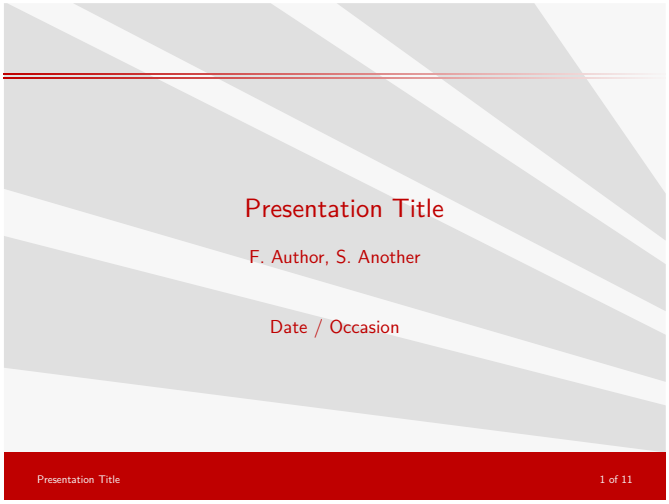


Horizontal Picture

Vertical Picture

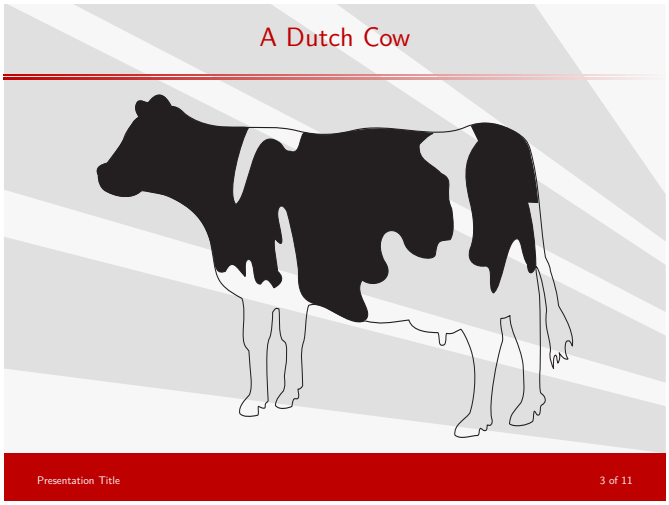
Sunrise

This style is inspired by the *husky* theme of the Latex's `powerdot` package.



Title Page

Normal Slide

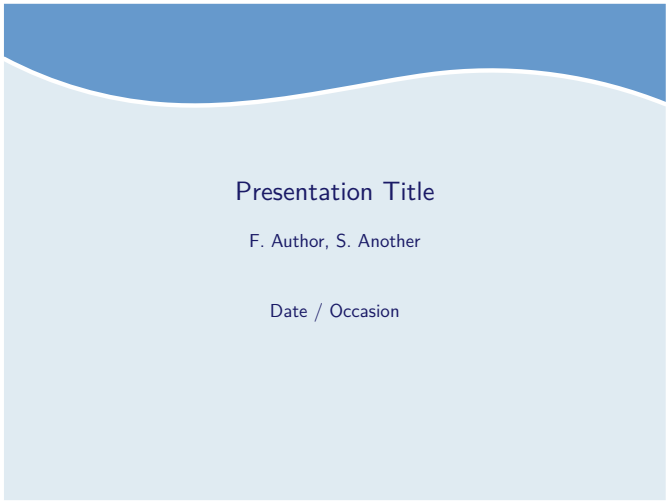


Horizontal Picture

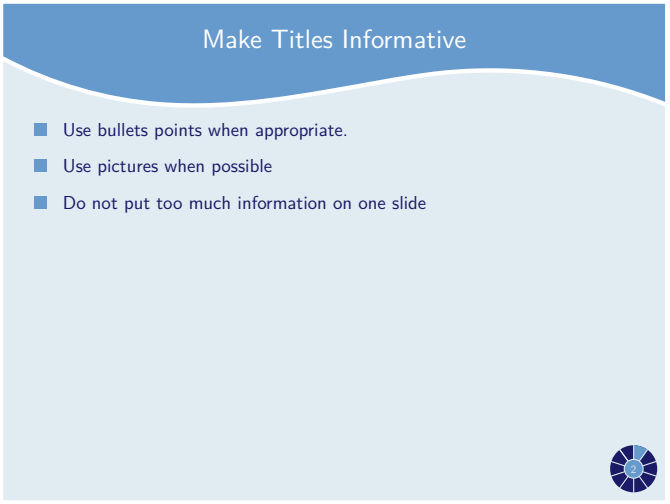
Vertical Picture

Swoosh

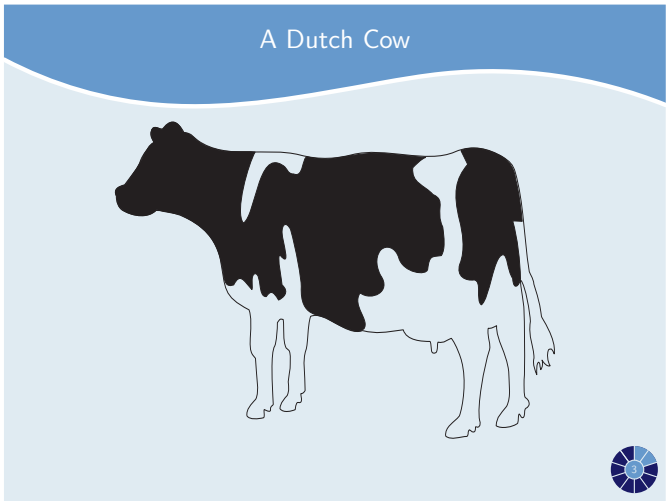
Take a break from the right angles and straight lines. Use swooshy curves. This style also has a fancy page counter at the bottom.



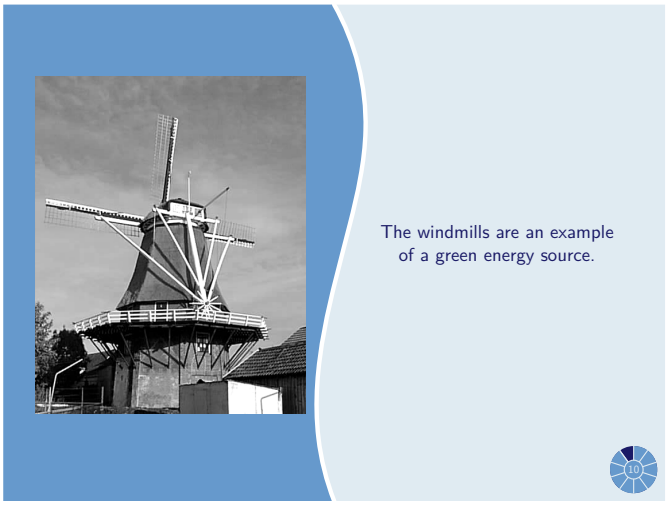
Title Page



Normal Slide



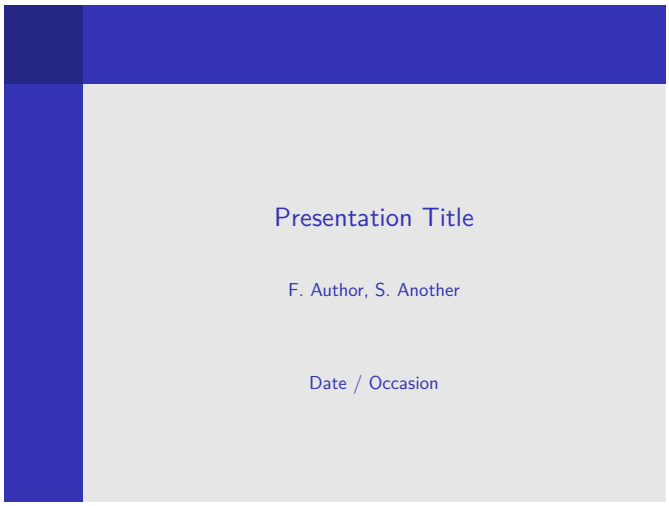
Horizontal Picture



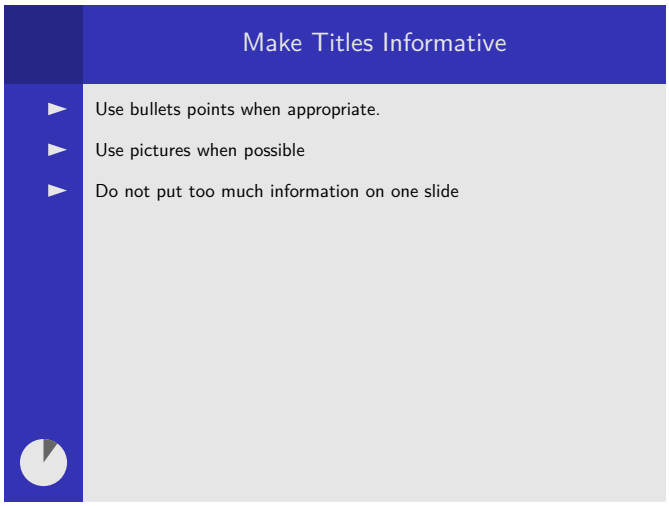
Vertical Picture

ThickStripes

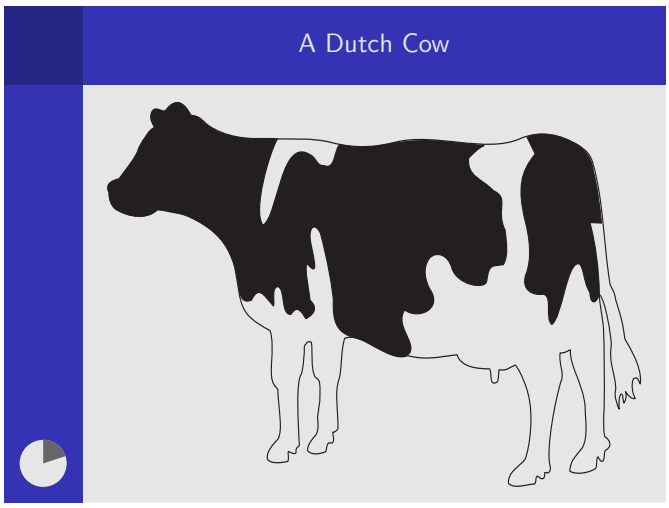
This theme is inspired by the *Berkeley* style of the Latex's `beamer` package. It has a stop watch at the bottom, which keeps track of the number of slides.



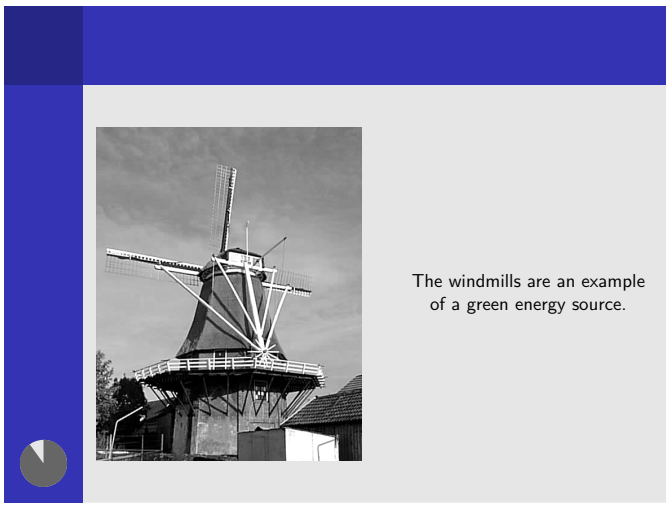
Title Page



Normal Slide



Horizontal Picture



Vertical Picture

7 Changing presentation fonts

The `font` and the `size` keys to `\setupmodule[simpleslides]` determine the font and font size for the main text of the presentation. The default font is Latin Modern Sans at 17pt.

- The `font` key can take the following values.

`LatinModern`	typesets in Latin Modern Serif
`LatinModernSans`	typesets in Latin Modern Sans
`Bookman`	typesets in TexGyre Bonum (a Bookman clone)
`Chancery`	typesets in TexGyre Chorus[1] (a Zapf Chancery clone)
`Gothic`	typesets in TexGyre Adventor (a Gothic clone)
`Helvetica`	typesets in TexGyre Heros (a Helvetica clone)
`Palatino`	typesets in TexGyre Pagella (a Palatino clone)
`Schoolbook`	typesets in TexGyre Schola (a Schoolbook clone)
`Times`	typesets in TexGyre Termes (a Times clone)

- The `size` key can be any valid Tex dimension.

Choosing your own font

If you want to set up your own font, pick any value for the `font` key (or leave it empty). Use the `size` key to choose the font size. Then *after* loading the module, choose any font using the normal Context commands. Make sure to set the bodyfont at size `\NormalSize`. So, if you have your own typescript for a font, your setup will look like this:

```
\usemodule[simpleslides][...]
....
\usetypescriptfile[type-myfont]    % The typescript for your font
\usetypescript[Mytypescript]       % As set in your typescript file
\setupbodyfont[Myfont,\NormalSize] % Note the \NormalSize here
```

Internally, the font size is stored in the macro `\NormalSize`. The main text is set at size `\NormalSize`; the main title is set at `\TitleSize` while the author and date on the title page, and the slide title are set at `\SlideTitleSize`.

`\NormalSize`, `\TitleSize`, and `\SlideTitleSize` are defined in terms of the dimensions `\simpleslidesNormalSize`, `\simpleslidesTitleSize`, and `\simpleslidesSlideTitleSize`. `\simpleslidesNormalSize` is equal to the `size` option. The module uses some heuristics to select a reasonable value of `\simpleslidesTitleSize` and `\simpleslidesSlideTitleSize`. If you do not like the size of the title page and slide titles, you can change their value to whatever you like.

[1] Please be aware that Chorus is a calligraphic font. It has no italic or bold.

8 Changing the title page

It is possible to change the look of **\placeTitle** using **\setupTitle**. This feature is intended for authors creating a new style, but may also be useful for someone who likes to tweak the presentation style. You should normally only set the title, authors, and date keys. If date is not set, then the module will default to **\currentdate**.

```
\setupTitle [..,.*=.,..]

* title        = TEXT
  author       = TEXT
  date         = TEXT
  style        = normal bold slanted boldslanted type cap small... COMMAND
  color        = IDENTIFIER
  align        = inner outer left right flushleft flushright middle center normal no
                 yes broad last r2l l2r
  before       = COMMAND
  after        = COMMAND
  titlestyle   = normal bold slanted boldslanted type cap small... COMMAND
  titlecolor   = IDENTIFIER
  titlealign   = inner outer left right flushleft flushright middle center normal no
                 yes broad last r2l l2r
  authorstyle  = normal bold slanted boldslanted type cap small... COMMAND
  authorcolor  = IDENTIFIER
  authoralign  = inner outer left right flushleft flushright middle center normal no
                 yes broad last r2l l2r
  titlestyle   = normal bold slanted boldslanted type cap small... COMMAND
  titlecolor   = IDENTIFIER
  titlealign   = inner outer left right flushleft flushright middle center normal no
                 yes broad last r2l l2r
  datestyle    = normal bold slanted boldslanted type cap small... COMMAND
  datecolor    = IDENTIFIER
  datealign    = inner outer left right flushleft flushright middle center normal no
                 yes broad last r2l l2r
  beforetitle  = COMMAND
  beforeauthor = COMMAND
  beforedate   = COMMAND
  aftertitle   = COMMAND
  afterauthor  = COMMAND
  afterdate    = COMMAND
```

9 Changing the slide titles

It is possible to change the look of **\SlideTitle** using **\setupSlideTitle**. Like **\setupTitle**, this feature is intended for authors creating a new style. You can use this command to make a minor change in an existing style, if you want.

```
\setupSlideTitle [..,.*=.,..]

* page        = yes no IDENTIFIER
  style       = normal bold slanted boldslanted type cap small... COMMAND
  before      = COMMAND
  after       = COMMAND
  align       = TEXT
  alternative = normal layer
```

10 Special macro for including pictures

As explained earlier, the **\IncludePicture** macro facilitates the placement of pictures. It takes four arguments (one of which is optional, and as such wasn't mentioned in the previous description).

```
\IncludePicture [...,...1] [...,...2] [..,.=.3,..] {.. ...4 ..}

1 horizontal vertical
2 IDENTIFIER
3 inherits from \setupPicture
4 CONTENT
```

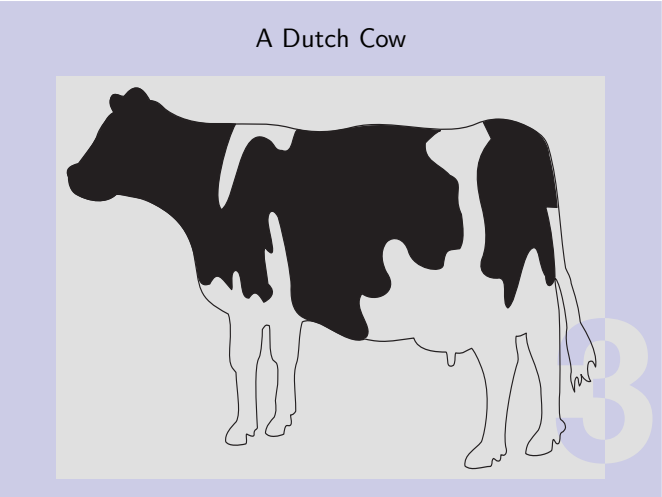
As explained earlier, the first argument determines whether the picture will be placed in horizontal or vertical layout; for examples, see Figure 2. The second argument is the filename of the picture that you want to include. The third argument is an optional argument useful for highlighting the picture. The fourth argument (in braces) is the text accompanying the picture. For horizontal pictures, this text is placed as a **\SlideTitle**; for vertical pictures this text is placed opposite to the picture, centered horizontally and vertically.

The third argument is the most complex. It specifies picture dimensions and highlights. If you want all pictures to share a common value (like `color` or `shadow`), specify them using **\setupPicture**.

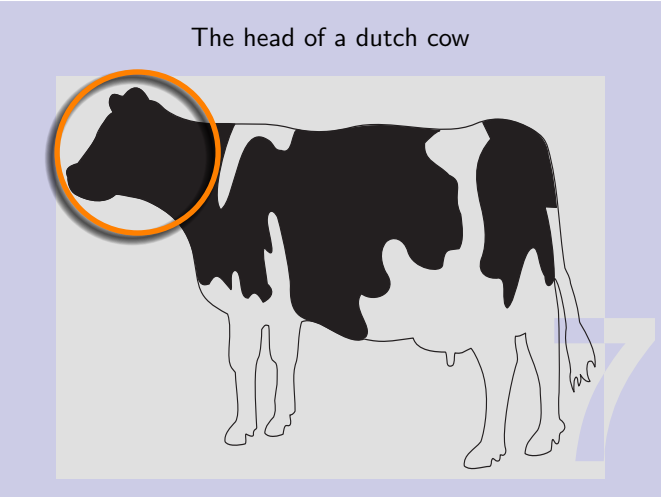
```
\setupPicture [..,.*=.,..]

* height         = DIMENSION
  width          = DIMENSION
  highlight      = yes no
  alternative    = none circle arrow focus
  color          = IDENTIFIER
  rulethickness  = DIMENSION
  x              = NUMBER
  y              = NUMBER
  xscale         = NUMBER
  yscale         = NUMBER
  length         = DIMENSION
  direction      = NUMBER
  opacity        = NUMBER
  length         = DIMENSION
  shadow         = yes topleft topright bottomright bottomleft no
  shadowcolor    = IDENTIFIER
  grid           = yes no
  subgrid        = yes no
  gridcolor      = IDENTIFIER
  steps          = NUMBER
```

Below is a brief explanation of what the different parameters do:

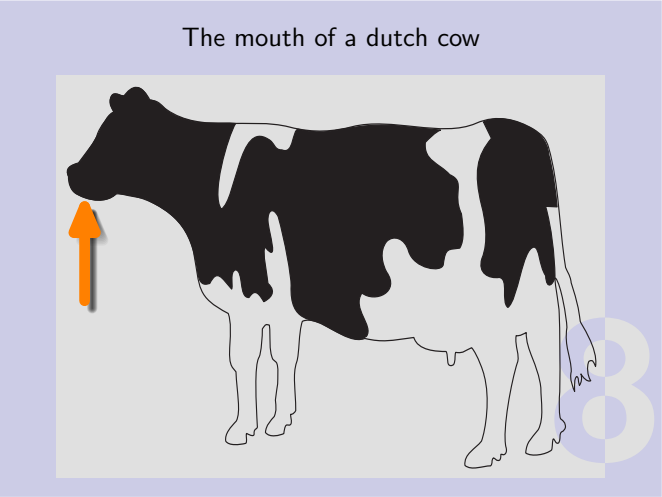
- `width` and `height`
  Unsurprisingly, these set the width and height of the picture. Normally, the module will automatically scale your pictures to fill the available space, so you only need to set one of these values if you want to override this mechanism.
- `highlight`
  This key determines the highlighting of the picture. If you set `highlight=yes`, then you can use one of the three available highlights: `circle`, `arrow`, and `focus`. These highlights are shown in Figure 3. The specific highlight is chosen using the `alternative` key. The location of the highlight is specified using the `x` and `y` keys. The scaling and rotation of the highlights is set using `xscale`, `yscale`, `length` and `direction`.
- `alternative`
  When `highlight=yes`, three different highlights are available: `circle`, `arrow`, and `focus`.



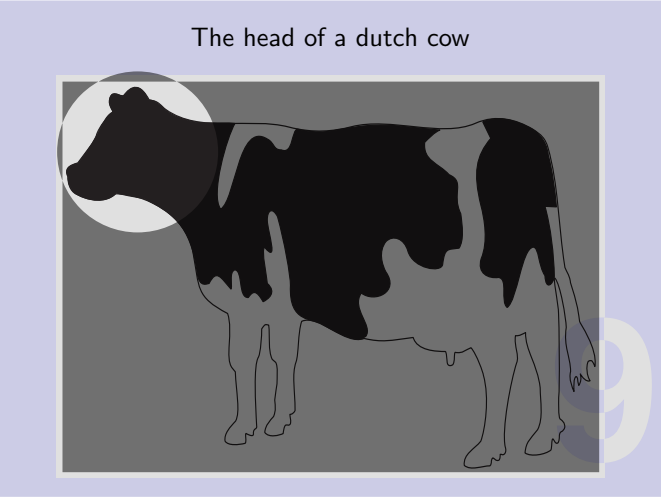
Picture with `highlight=no` (default)



Picture with `highlight=yes` and `alternative=circle`



Picture with `highlight=yes` and `alternative=arrow`



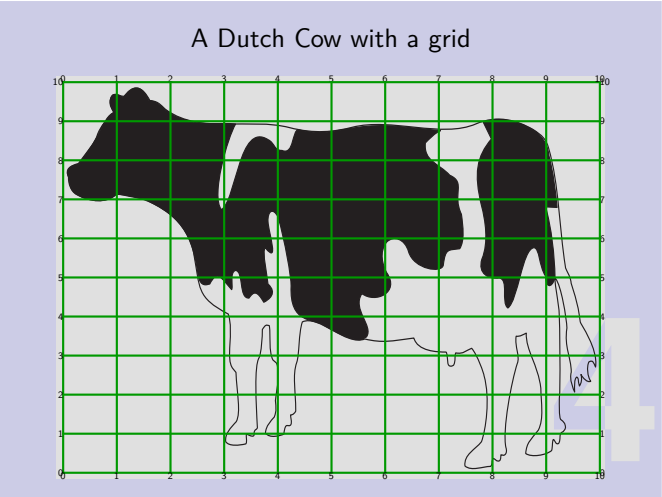
Picture with `highlight=yes` and `alternative=focus`

Figure 3 Different highlight options available

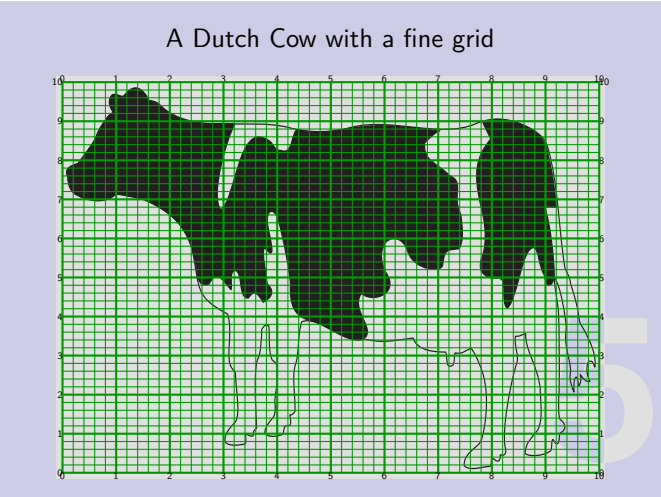
Units for dimensions

All dimensions are specified relative to the width and height of the image, so you do not have to change the location of your highlights if you change the presentation style. The dimensions `x` and `y` should be a number between 0 and 10. The `x` is scaled by 1/10 times the width of the image; the `y` value is scaled by 1/10 times the height of the image. The easiest way to understand this is to look at a scaled grid superimposed on the picture, as in Figure 4. The grid is configured as follows:

- `grid` and `subgrid`
  These options determine whether or not to show the grid and sub-grid. The `grid` divides the height and width of the picture into 10 sections; this is helpful for determining the exact position where you want to place circles and arrows. The `subgrid` divides the grid into a finer grid. Each cell is divided into `steps` times `steps` cells.



Picture with `highlight=yes` and `grid=yes`



Picture with `highlight=yes`, `grid=yes` and `subgrid=yes`

Figure 4 Grids for help in determining the location of highlight

- `gridcolor`
  This option determines the color in which the grid is drawn. It can be any Context color identifier. The default value is green.
- `steps`
  The number of subdivisions for the `subgrid`. The default value is 5.

Highlighting by a circle

Now lets see how different highlight alternatives are specified. Suppose we want to place the picture of a cow and highlight its head. To help determine the center of the circle, we can first superimpose a fine grid on the picture, and read the value for the center. From Figure 4, `x=1.4` and `y=8.2` seems like a good value. Next we need to decide on the radius of the circle. The radius can either be specified in terms of the "x units" (1/10th of the picture width) or "y units" (1/10th of the picture height). Lets try a radius of 1.5 "x units". This can be specified as `xscale=1.5`. If we wanted something in terms of "y units", we could have used `yscale`. If both `xscale` and `yscale` are specified, we will get an ellipse. Thus, to draw the circle highlight shown in Figure 3, we wrote

```
\IncludePicture
  [horizontal]
  [cow] % Name of the image
  [highlight=yes,
   alternative=circle,
   x=1.4,
   y=8.2,
   xscale=1.5,
   shadow=bottomleft]
```

```
  {The head of a dutch cow}
```

If `direction` key is specified, the circle (or the ellipse) will be rotated by that amount (in degrees) in the counter clockwise direction. The color in which the circle is drawn is specified using `color` key. The thickness of the line is determined by `rulethickness` key. By default, `color=orange` and `rulethickness` is 1/100th of the picture width.

In summary, the different keys related to `alternative=circle` are:

- `highlight=yes` and `alternative=circle`
  These are needed to specify a circle highlight.
- `x` and `y`
  The center of the circle in terms of scaled units. Their values should be between 0 and 10.
- `xscale` and `yscale`
  The radius of the circle if only one option is specified. The major and minor radii of the ellipse if both options are specified.
- `direction`
  The amount by which the circle is rotated. This only makes sense if we are actually drawing an ellipse.
- `rulethickness`
  The line width of the circle.
- `color`
  The color of the circle.

Highlighting by an arrow

Suppose we want to include a picture of a cow and point out its mouth using an arrow. An arrow is specified by three things, the location of its tip, given by `x` and `y` keys; the length of the arrow, given by `length` key; and the direction of the tail, given by `direction` key. Thus, to draw the arrow highlight shown in Figure 3, we wrote

```
\IncludePicture
  [horizontal]
  [cow] % Name of the image
  [highlight=yes,
   alternative=arrow,
   x=0.4,
   y=6.8,
   direction= 90,
   length=3cm,
   shadow=bottomleft]
  {The mouth of a dutch cow}
```

The different keys related to `alternative=arrow` are:

- `highlight=yes` and `alternative=arrow`

These are needed to specify a arrow highlight.

- `x` and `y`
  The tip of the arrow in terms of scaled units. Their values should be between 0 and 10.
- `length`
  The length of the arrow. This is a dimension.
- `direction`
  The amount by which the arrow is rotated.
- `rulethickness`
  The line width of the arrow. (Actually the line width of the arrow is twice the given value. This is so that both arrows and circles look good with the same value of rulethickness.)
- `color`
  The color of the arrow.

Highlighting by focus

Suppose we want place the picture of a cow, focus its head, and dull out rest of the picture. The area to be focused is a circle (or an ellipse) and it can be specified using `x` and `y` to indicate the center, `xscale` and `yscale` to indicate the radius, and `direction` to indicate the rotation. The keys `rulethickness` and `color` do not have any effect. The area other than the focussed area is washed out with a transparent color. The degree to which it is washed out is determined by `opacity` (default value 0.5), and the color of the unfocussed area is determined by `shadowcolor` (default value black). Thus, to draw the focus highlight shown in Figure 3, we wrote

```
\IncludePicture
  [horizontal]
  [cow] % Name of the image
  [highlight=yes,
   alternative=focus,
   x=1.4,
   y=8.2,
   xscale=1.5]
  {The head of a dutch cow}
```

The different keys related to `alternative=focus` are:

- `highlight=yes` and `alternative=focus`
  These are needed to specify a focus highlight.
- `x` and `y`
  The center of the circle in terms of scaled units. Their values should be between 0 and 10.
- `xscale` and `yscale`
  The radius of the circle if only one options is specified. The major and minor radii of the ellipse if both options are specified.
- `direction`

The amount by which the circle is rotated. This only makes sense if we are actually drawing an ellipse.

- `opacity`
  The opacity of the unfocussed area. `opacity=0` is transparent, while `opacity=1` is completely opaque.
- `shadowcolor`
  The color of the unfocused area.

Adding shadows

When a circle or arrow highlight is used, adding a shadow to the highlight makes them stand out more. The key related to shadows is:

- `shadow`
  This key determines whether shadows are placed or not. By default, shadows are disabled. If not set to `no`, this key determines where the shadow is placed: at `topleft`, `topright`, `bottomleft`, or `bottomright`. Setting this key to `yes` puts the shadow at `bottomright`.

Adding a specific page

To select a specific page from a multi-page pdf file, you can use `page=<number>` option.

Using your own style

The module makes it easy to write your own style or to tweak one of the provided styles beyond the configuration options provided by the module. Simply copy the style which is closest in appearance to what you want to obtain. Give it a filename `s-myownstyle.tex`, `myownstyle` being any name you like. Put this file into a directory where Context will find it, either the directory where you will process your presentation or somewhere in your personal `$TEXMF` tree. Then, let the module know that you want to use your own style:

```
\usemodule[simpleslides]
          [style=myownstyle]
```

The module will read your file and apply your settings.