A Glance at Animate

Tobias Nähring

03-12-2008

Intro

- Animation of **objects set with LaTeX** (e.g. PSTricks/Ti*k*Z)
- Animation of image files
- Large set of animation options

Prerequisites

You always need to input the `animate`-package:

```
\usepackage{animate}
```

Prerequisites

You always need to input the `animate`-package:

```
\usepackage{animate}
```

Optional args may be set *globally* as package arguments.

Example (option `controls` explained later in detail):

```
\usepackage[controls]{animate}
```

Most Simple Inline-Animation

Syntax of the `animateinline` environment:

```
\begin{animateinline}[options]{framerate}
...stuff... \newframe ...more stuff...
\end{animateinline}
```

framerate frames per second

\newframe separates contents of successive frames

Click here: 1st Frame

```
\begin{animateinline}{1}
  % one frame per second
  1st Frame
  \newframe
  2nd Frame
  \newframe
  3rd Frame
\end{animateinline}
```

Most Simple Animated Image-File Sequence

Syntax of the `\animategraphics` macro:
`\animategraphics[options]{framerate}{name}{first}{last}`

framerate: frames per second

name: file name without *image number* and without *extension*

first & last: numbers of first and last image

Click here:

```
\animategraphics{1}%
  {images/numbers-}% name
  {1}{3}% first and last
```

- Here: Files `numbers-1.pdf`, `numbers-2.pdf`, and `numbers-3.pdf` exist in sub-directory `images`
- In general: All image formats of the `graphicx`-package are usable.

Pausing

The most simple inline animation revisited:

Pausing

\newframe* animation pauses until mouse-click

Click here: 1st Frame

```
\begin{animateinline}{1}
  1st Frame
  \newframe
  2nd Frame
  \newframe*
  3rd Frame
  \newframe
  4th Frame
\end{animateinline}
```

Inline Animation Sequences

The most simple inline animation revisited:

Programming Loops for Animations

Click here:
Frame 1

```
\begin{animateinline}{1}
  \multiframe{10}{
    iCount=1+1,
    dLength=0cm+0.2cm
  }{
    \rule{\dLength}{1ex}
    Frame~\iCount
  }
\end{animateinline}
```

Interesting: The animated box keeps its dimensions.

Important Animation Options

autoplay	start without mouse-click
autopause	don't reset but only pause when leaving slide
autoresume	automatic continue where stopped
loop	loop animation
palindrome	play forward and backward consecutively
draft, final	just bounding box or animation
every	take only every *n*-th figure from image-sequence
controls	buttons for animation control
buttonsize, buttonbg, buttonfg	manipulation of the buttons
step	one slide per click
poster=*first(default)—none—last*	show this slide as default

Animation Options in Action (autoplay)

autoplay : Animation starts without clicking

No need to click:
Frame 1

```
\begin{animateinline}[autoplay]{1}
  \multiframe{5}{iCount=1+1}{
    Frame \iCount}
\end{animateinline}
```

Animation Options in Action (autopause)

autopause leaving the slide only stops the animation (animation is not reset)

Switch to the previous slide and come back. When returning here you can resume the animation where you left.

Click on **Frame** go to previous page and come back Frame 1

```
\begin{animateinline}[autopause]{1}
  \multiframe{10}{iCount=1+1}{
    Frame \iCount}
\end{animateinline}
```

Animation Options in Action (autoresume)

autoresume in conjunction with `autopause`, automatically start animation when coming back

Click on **Frame** go to previous page and come back Frame 1

```
\begin{animateinline}[autoresume,
  autopause]{1}
  \multiframe{10}{iCount=1+1}{
    Frame \iCount}
\end{animateinline}
```

Animation Options in Action (loop and palindrome)

loop — loop animation

palindrome — play animation forward then backward and so on

Loop:
Frame 1

Palindrome:
Frame 1

```
\begin{animateinline}[loop]{1}
  \multiframe{5}{iCount=1+1}{
    Frame \iCount}
\end{animateinline}

\begin{animateinline}[palindrome]{1}
  \multiframe{5}{iCount=1+1}{
    Frame \iCount}
\end{animateinline}
```

Animation Options in Action (step)

step one frame per mouse-click

Palindrome + Step:
Frame 1

```
\begin{animateinline}[step,
  palindrome]{1}
  \multiframe{5}{iCount=1+1}{
    Frame \iCount}
\end{animateinline}
```

Animation Options in Action (controls)

controls show buttons for animation control

buttonfg, -bg, -size manipulation of button color and button size


```
\begin{animateinline}
  [controls,buttonsize=2ex,
   buttonfg=1.0:0.0:0.0,
   buttonbg=0.8]{1}
  \multiframe{5}{iCount=1+1}{
    Frame \iCount}
\end{animateinline}
```

Animation Options in Action (draft/final)

draft, final shows only bounding box or full animation

Option `draft` may save a lot of translation time for presentations with many animations.

Draft:

Final: Frame 1

```
\begin{animateinline}[draft]{1}
  \multiframe{5}{i=1+1}{Frame \i}
\end{animateinline}

\begin{animateinline}[final]{1}
  \multiframe{5}{i=1+1}{Frame \i}
\end{animateinline}
```

graphicx-Options for animate

scale, bb, viewport, trim known options from graphicx

without viewport:

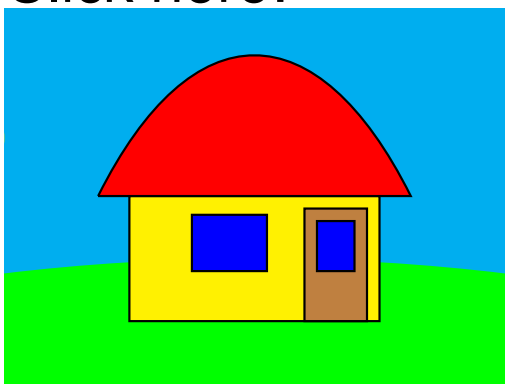
with viewport:

```
\animategraphics
  [scale=0.25]
  {1}{images/numbers-}{1}{3}

\animategraphics
  [viewport=0 0 110 153,
  scale=0.25]{1}
  {images/numbers-}{1}{3}
```

Using TikZ for Inline Graphics

Click here:



```
\begin{animateinline}[
  begin={ % header of each frame
          \begin{tikzpicture}
          [line width=1pt]
          \path[clip] (0,0) rectangle (8,6)
        },
  end={\end{tikzpicture}} also each frame
  ]{3}
  \multiframe{20}{iAngle=120+-5}{
    ...
    \fill[fill=yellow] % the sun
    (\iAngle:8cm) circle (1);
    ...
  } % end of multiframe
\end{animateinline}
```

Timeline-Option (Intro)

Timelines. . .

- give the user full control over the *order* and *combination* of frames in the actual shown sequence of pictures

Timeline-Option (Intro)

Timelines...

- give the user full control over the *order* and *combination* of frames in the actual shown sequence of pictures
- are stored in separate plain text files with special syntax

Timeline-Option (Definitions)

Definition (corresponding to the documentation of `animate`):

Transparency stuff from one image file or from one ‘frame’ in a `animateinline` environment

Timeline-Option (Definitions)

Definition (corresponding to the documentation of `animate`):

Transparency stuff from one image file or from one ‘frame’ in a `animateinline` environment

Frame combination of transparencies which are displayed at one point of time (transparencies can overlap each other).

Timeline-Option (Definitions)

Definition (corresponding to the documentation of `animate`):

Transparency stuff from one image file or from one 'frame' in a `animateinline` environment

Frame combination of transparencies which are displayed at one point of time (transparencies can overlap each other).

- the image-files of one `\aimategraphics` command are numbered consecutively with $0, 1, 2, \ldots$. Same goes for the transparencies from one `animateinline` environment

Timeline-Option (Definitions)

Definition (corresponding to the documentation of `animate`):

Transparency stuff from one image file or from one 'frame' in a `animateinline` environment

Frame combination of transparencies which are displayed at one point of time (transparencies can overlap each other).

- the image-files of one `\aimategraphics` command are numbered consecutively with $0, 1, 2, \ldots$. Same goes for the transparencies from one `animateinline` environment
- transparencies are addressed by their numbers for reordering or combining into frames.

Timeline-Option (Syntax)

The timeline syntax...

- % starts a LaTeX-like line comment

Timeline-Option (Syntax)

The timeline syntax...

- % starts a LaTeX-like line comment
- Each non-void line stands for a complete frame

Timeline-Option (Syntax)

The timeline syntax...

- % starts a LaTeX-like line comment
- Each non-void line stands for a complete frame

Syntax of one line:

Pausing : Framerate : TranspariesInFrame

Timeline-Option (Syntax)

The timeline syntax...

- % starts a LaTeX-like line comment
- Each non-void line stands for a complete frame

Syntax of one line:

Pausing : Framerate : TranspariesInFrame

Pausing a signle star * pauses animation if present like `\newframe*`

Timeline-Option (Syntax)

The timeline syntax...

- % starts a LaTeX-like line comment
- Each non-void line stands for a complete frame

Syntax of one line:

Pausing : Framerate : TranspariesInFrame

Pausing a signle star `*` pauses animation if present like `\newframe*`

Framerate changes number of frames per second if present (like framerate-argument of `\animateinline`)

Timeline-Option (Syntax)

The timeline syntax...

- % starts a LaTeX-like line comment
- Each non-void line stands for a complete frame

Syntax of one line:

Pausing : Framerate : TranspariesInFrame

Pausing — a signle star * pauses animation if present like `\newframe*`

Framerate — changes number of frames per second if present (like framerate-argument of `\animateinline`)

TransparenciesInFrame — simplest case: comma separated list of transparencies to be overlayed within current frame

Timeline-Option (Simple Example)

The contents of the timeline-file '`simple.timeline`':

```
::1 % 1st frame: transparency 1
::0 % 2nd frame: transparency 0
::1,2 % 3rd frame: transparencies 1 and 2
```

Usage example of '`simple.timeline`' within LaTeX:

Displays numbers of transparencies in current frame.

Click here: 1

```
\begin{animateinline}
  [timeline=simple.timeline]{1}
  % Note:\phantom stuff not printed.
  % It just keeps space.
  0\phantom{ 1 2}
  \newframe
  \phantom{0 }1\phantom{ 2}
  \newframe
  \phantom{0 1 }2
\end{animateinline}
```

Timeline-Option (Multi-Frame Transparency)

Goal: Let some transparencies keep staying for more than one frame without need to repeat its TransparenciesInFrame entry.

Extended syntax for TransparenciesInFrame entry:

TransparencyNumberxNumberOfFrames

Example ‘multipleFrames.timeline’:

```
::0x2,1 % let 0 stay for two frames, additionally show 1
::2
::1
```

Here, the LaTeX-example showing transparency numbers gives:

Click here: 0 1

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Earlier transparencies are overprinted by later ones.

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Earlier transparencies are overprinted by later ones.

- The timeline is processed line-wise.

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Earlier transparencies are overprinted by later ones.

- The timeline is processed line-wise.
- Each line is processed from left to right

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Earlier transparencies are overprinted by later ones.

- The timeline is processed line-wise.
- Each line is processed from left to right

- This way it is easy to keep the background staying
  - example on next slide

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Earlier transparencies are overprinted by later ones.

- The timeline is processed line-wise.
- Each line is processed from left to right

- This way it is easy to keep the background staying
  - example on next slide
- Overlapping problems arise if one wants to keep the foreground staying

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Earlier transparencies are overprinted by later ones.

- The timeline is processed line-wise.
- Each line is processed from left to right

- This way it is easy to keep the background staying
  - example on next slide
- Overlapping problems arise if one wants to keep the foreground staying
  - example for the problem: slide after the next one

Timeline-Option (Overlapping Problem)

The multi-frame extension introduces an overlapping problem:

Earlier transparencies are overprinted by later ones.

- The timeline is processed line-wise.
- Each line is processed from left to right

- This way it is easy to keep the background staying
  - example on next slide
- Overlapping problems arise if one wants to keep the foreground staying
  - example for the problem: slide after the next one
  - solution: thereafter

Timeline-Option (Overlapping Problem)

Extended TransparenciesInFrame-syntax enables you to keep the background staying.

Click here: fore

Content of Frames:

0. yellow background
1. word `fore`
2. word `ground`

Timeline:

```
::0x2,1
::2
```

Timeline-Option (Overlapping Problem)

Up to now we are not able to keep the foreground picture staying and change the background.

Click here: foreground

Contents of frames:

0. yellow background
1. red background
2. word `foreground`

Timeline:

```
::0,2x2
::1
```

Timeline-Option (Overlapping Problem/Layers)

Solution: *layers*.

Syntax:

- we say layer to the TransparenciesInFrame-stuff we know up to now (transparencies & multiframe transparencies)

Timeline-Option (Overlapping Problem/Layers)

Solution: *layers*.

Syntax:

- we say layer to the TransparenciesInFrame-stuff we know up to now (transparencies & multiframe transparencies)
- the new TransparenciesInFrame-entry may be composed of several ;-separated layers:
  1st layer;2nd layer;...;last layer

Timeline-Option (Overlapping Problem/Layers)

Solution: *layers*.

Syntax:

- we say layer to the TransparenciesInFrame-stuff we know up to now (transparencies & multiframe transparencies)
- the new TransparenciesInFrame-entry may be composed of several ;-separated layers:
  1st layer;2nd layer;...;last layer

Each line in the timeline file is processed layerwise:

1st the first layer is printed with all its multiframe transparencies.

Timeline-Option (Overlapping Problem/Layers)

Solution: *layers*.

Syntax:

- we say layer to the TransparenciesInFrame-stuff we know up to now (transparencies & multiframe transparencies)
- the new TransparenciesInFrame-entry may be composed of several ;-separated layers:
  1st layer;2nd layer;...;last layer

Each line in the timeline file is processed layerwise:

1st the first layer is printed with all its multiframe transparencies.

2nd the second layer with all its multiframe transparencies

Timeline-Option (Overlapping Problem/Layers)

Solution: *layers*.

Syntax:

- we say layer to the TransparenciesInFrame-stuff we know up to now (transparencies & multiframe transparencies)
- the new TransparenciesInFrame-entry may be composed of several ;-separated layers:
  1st layer;2nd layer;...;last layer

Each line in the timeline file is processed layerwise:

1st the first layer is printed with all its multiframe transparencies.

2nd the second layer with all its multiframe transparencies

... and so on

Timeline-Option (Layers)

With the help of layers we can let stay the foreground as a multiframe transparency and change the background.

The last example revisited:

Click here: foreground

Contents of frames:

0. yellow background
1. red background
2. word `foreground`

Timeline:

```
::0;2x2
::1
```

Bye

Thank you for your attention.