

GIF animation support in PDF mode

Michael Vulis

March 01, 2000

Abstract

This document explains the low-level support for Gif animation in the PDF mode. The information as provided here is preliminary and subject to changes.

1 Credits

The author is indebted to Denis Girou for suggesting the original idea and his prototype implementation.

2 How does it work?

Before getting into specifics and examples, it is important to realize that Gif animation is impossible. There is no support in the Pdf format for inclusion of animated Gifs; the movie support as provided by Pdf is so platform-dependant as to be avoided; and the JavaScript support is inadequate for achieving animation by switching frames via a JavaScript timer. What you are about to see is really a fake composed of overlapping frames, and made possible only by the slowness of Pdf rendering.

One the positive side, it is a working fake which is rather easy to use, and which delivers most of what the users expect from animated Gifs.

3 `special's`

Gif Animation is implemented in VTeX by means of several additional `\special's`; these modify the behavior of the `\includegraphics` command rather than do anything themselves. *At some point in the future, we expect to incorporate these \special's directly into the `graphics` driver, but for now one needs to code them directly.* The most important `\special` is

```
\special{!supportgifani+}
```

which enables Gif animation. This is because the default in VTeX is to process animated Gif's as un-animated, by taking only the first frame.

This `\special` must be issued before animated gif(s) is actually embedded; using the `\immediate` prefix is recommended in cases when Gif animations may reside within Pdf forms.

```
\immediate\special{!supportgifani+}
```

Like most other V_TE_X toggle `\special`'s, this `\special` can be followed by “+” (to turn on animating), by “-” (to turn it off), or by “^” (to pop the previous state).

With Gif animation support enabled, usual `\includegraphics` will transfer the animation to the Pdf output:

```
\includegraphics[width=2in]{mail.gif}
```

```
\special{!supportgifani+}%
\includegraphics[width=2in]{mail.gif}
```



Notice that the first image is non-animated; while the second is animated, the animation plays only once and perhaps too quickly to be noticed. [You can make the animation play again by forcing the Acrobat to redraw the area where the animation resides; for example, by changing the zoom.]

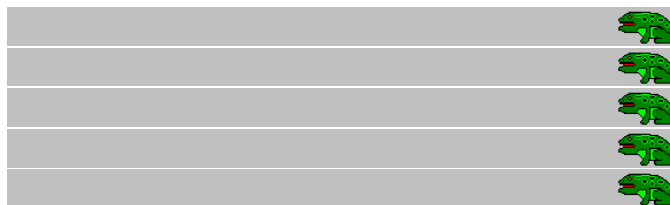
The additional problem is that multiple animations that may reside on a page are played sequentially (and, again, only once each).

The next page demonstrates this:

```

\special{!supportgifani+}%
\includegraphics[width=3.5in]{mail.gif}\par
\includegraphics[width=3.5in]{mail.gif}\par
\includegraphics[width=3.5in]{mail.gif}\par
\includegraphics[width=3.5in]{mail.gif}\par
\includegraphics[width=3.5in]{mail.gif}\par

```



All three of the undesirable effects

- Animations are played sequentially
- Animations are played only once
- Animations' timing could be wrong

come from the absence of a **true** animation support in the Pdf format. The additional `\special`'s provided in `VTEX` can be used to alleviate these problems.

4 Controlling the animation speed

By default, `VTEX` retrieves the animation time from the Gif image itself and converts it to the number of duplications of each frame with the formula

$$\text{Dupl} = \text{GifTime} \times \frac{C}{1000} \quad (1)$$

where C defaults to 500 [which seems to be a good default for most pictures]. The

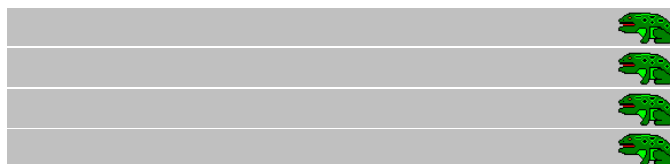
```
\special{!gifanidup ...}
```

can be used to alter the value of the coefficient. For example,

```

\special{!supportgifani+}%
\special{!gifanidup 300}%
\includegraphics[width=3.5in]{mail.gif}\par
\special{!gifanidup 500}%
\includegraphics[width=3.5in]{mail.gif}\par
\special{!gifanidup 1000}%
\includegraphics[width=3.5in]{mail.gif}\par
\special{!gifanidup 1500}%
\includegraphics[width=3.5in]{mail.gif}\par

```



If you alter this coefficient to change the play speed of a picture, make sure to reset it back to 500 afterwards.

```

\special{!gifanidup 500}%

```

5 Controlling the animation duration

In a similar fashion, you can control the duration of animations. While there is no way to make an animation play “forever”, you can make it play a given (and, possibly, large) number of times.

The `\special` to use is

```

\special{!gifanirep ...}

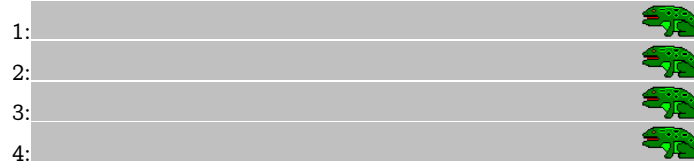
```

with the argument set to the number of repetitions. The default is once (1). For example,

```

\special{!supportgifani+}%
\special{!gifanirep 1}%
1:\includegraphics[width=3.5in]{mail.gif}\par
\special{!gifanirep 2}%
2:\includegraphics[width=3.5in]{mail.gif}\par
\special{!gifanirep 3}%
3:\includegraphics[width=3.5in]{mail.gif}\par
\special{!gifanirep 4}%
4:\includegraphics[width=3.5in]{mail.gif}\par

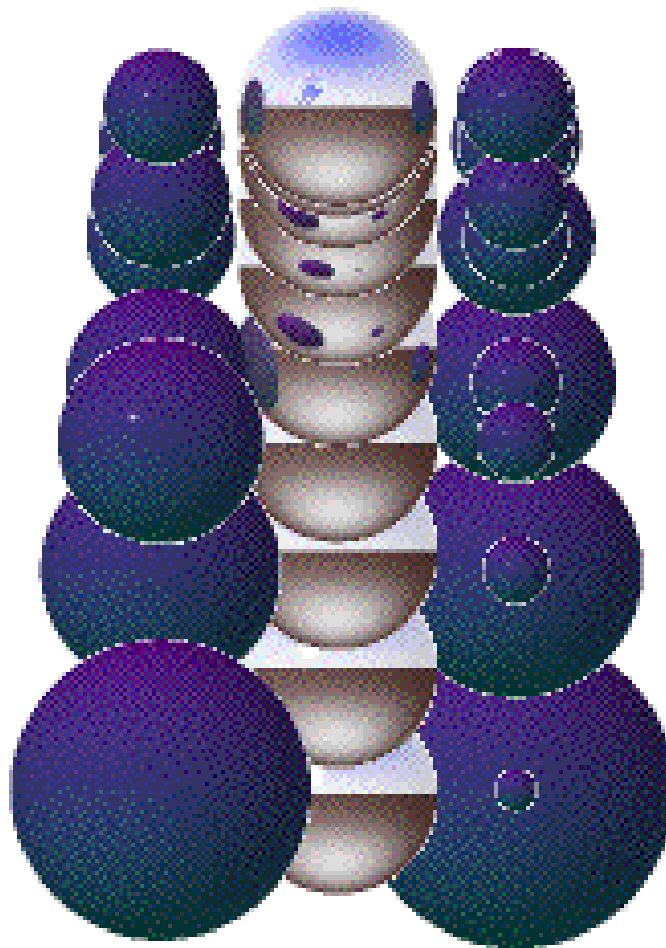
```



6 Correcting the transparency bugs

Many animated Gif do not set the transparency flag correctly; this may result in the background not being cleared between the animations:

```
\includegraphics[width=3.5in]{balls.gif}
```

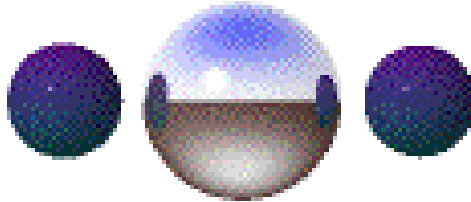


The final `\special` is

```
\special{!readtranspgif-}
```

which forces \TeX to ignore the transparency flag. For pictures like `balls.gif` this makes the background work correctly.

```
\special{!readtranspgif-}
\includegraphics[width=3.5in]{balls.gif}
```



7 Animations and Forms

One can improve the way the animations are managed by placing them inside Pdf form fields. **Unfortunately, this is supported only with Acrobat 4; so if you are using an older version of Acrobat, stop now! Proceeding to the next page may crash your system.**

We will start by defining a simple macro that places animation into a reusable form:

```
\newcount\N
\newcommand{\IncludeAnimatedGif}[2]{%
\setbox0=\hbox to #2{\hss
\hbox{%
\includegraphics[width=#2]{#1.gif}}%
\hss}%
\immediate\shipout\box0
\N=\formcount
\ vbox to \formht\N{\hbox to \formwd\N{%
\special{!field k=b,w=\the\formwd\N,h=\the\formht\N,
pnu=\the\N,pru=\the\N,P}\hss}}}

```

*The details of VT_{EX} form syntax can be found in the *forms.pdf* manual.*

We now use this macro to include the `balls.gif` animation:

```
\immediate\special{!readtranspgif-}
\IncludeAnimatedGif{Balls}{3.5in}

```


There are two advantages to this over the previous approach:

- This animation can be played again; try moving the mouse over it.
- While all the previous examples will yield a mess when printed [the frames will overlap], this animation was set within an unprintable field (the P flag). when printed

A slight variation to the above macro allows us to avoid the initial animation:

```
\newcount\N
\newcount\M
\newcommand{\IncludeAnimatedGif}[2]{%
\setbox0=\hbox to #2{\hss
\hbox{%
\includegraphics[width=#2]{#1.gif}}%
\hss}%
}
```

```

\immediate\shipout\box0
\N=\formcount
\setbox0=\hbox to #2{\hss
  \hbox{%
    \special{!supportgifani-}%
    \includegraphics[width=#2]{#1.gif}}%
    \special{!supportgifani^}%
  \hss}%
\immediate\shipout\box0
\M=\formcount
\ vbox to \formht\N{\hbox to \formwd\N{
\special{!field k=b,w=\the\formwd\N,h=\the\formht\N,
  prn=\the\M,pru=\the\N,P}\hss}}}

and now:

\immediate\special{!readtranspgif-}
\IncludeAnimatedGif{Balls}{3.5in}

```

Now, the animation will appear only when the mouse is moved over the image; by default, the image shows only the first frame. It is also possible to have the animation activated by clicking on the image; see **forms.pdf** for additional information.

This technique allows to place more than one animated image on a page and have the animations activated individually. Here are nine images (*not eight*) to show the concept:

