

The combinedgraphics package*

Christian Schneider
<software(at)chschneider(dot)eu>

July 13, 2012

Contents

1	The Combined EPS/L^AT_EX format	2
2	Usage	3
2.1	The macro	3
2.2	Basic macro options	3
2.3	Extended macro options	4
2.4	Package options	5
3	Bugs, problems, and suggestions	5
4	Implementation	5
4.1	Package options	6
4.2	Basic macro options	6
4.3	Extended macro options	8
4.4	The macro	10
4.5	Helper macros	12

*This document corresponds to combinedgraphics v0.2.2, dated 2012/07/13. Copyright 2009,2011,2012 Christian Schneider <software(at)chschneider(dot)eu>.

Abstract

This package provides a macro (`\includecombinedgraphics`) for the inclusion of combined EPS/L^AT_EX and PDF/L^AT_EX graphics (an export format of Gnuplot, Xfig, and maybe other programs). Instead of including the graphics with a simple `\input`, the `\includecombinedgraphics` macro has some benefits:

- changing the font and color of the text of the L^AT_EX parts
- rescaling the graphics without affecting the font of the L^AT_EX parts
- automatic inclusion of the vector graphics parts, as far as L^AT_EX parts do not do it (e. g., for files exported from Gnuplot before version 4.2 or Xfig)
- changing the inclusion order of L^AT_EX and vector graphics parts (e. g., Gnuplot 4.4 exports files in a way that the vector graphics part overlays the L^AT_EX part, which means that text may be hidden under shaded areas)
- rescaling and rotating of complete graphics (similar to `\includegraphics` from `graphicx` package)
- scaling the vector graphics part (without affecting the font) to a defined width or height of the graphics, which might, e. g., be of importance for graphics that are intended for publication in a journal requesting a certain size of graphics

A test PDF file with extensively commented examples is shipped with this package and a BASH script (`texpic2epspdf.sh`) to convert combined EPS/L^AT_EX and PDF/L^AT_EX graphics to EPS or PDF files with text part included.

1 The Combined EPS/L^AT_EX format

Graphics in combined EPS/L^AT_EX or PDF/L^AT_EX format consist of two files:

1. an EPS or PDF file containing the vector graphics part
2. an L^AT_EX file containing the text part

This format has the advantage that it is possible to generate a high-quality vector graphics with text containing all symbols, macros and other stuff provided by L^AT_EX. The text is formatted by L^AT_EX itself when including the graphics (*not* during creation of the graphics!) and thus it is possible to use everything provided by L^AT_EX up to the font of the document for text inside the graphics.

Typically, the L^AT_EX part of the graphics is included by `\input`. The vector graphics part is either included automatically by the L^AT_EX part with an `\includegraphics` command (or something similar) and overlaid by the text or the user will have to combine both parts manually like this (e. g., for gnuplot before version 4.2 or Xfig) for a pair of files `foobar.tex/foobar.eps`:

```
\begin{picture}(0,0)%
  \includegraphics{foobar}%
\end{picture}%
\input{foobar}
```

Although the format has some nice advantages compared to other formats, the user has to know whether to include the vector graphics part for (some of) his graphics or not, font changes sometimes require some knowledge of the internal commands of the \LaTeX part (depending on the tool that generated the graphics), and the user is not intended to change the scaling of just the vector graphics part without affecting the text format. Furthermore, rotating, scaling and resizing the whole graphics requires extra-macros, e.g. `\rotatebox`, `\scalebox`, or `\resizebox` from the `graphics/graphicsx` package.

The intention of this package is to provide a macro for easier handling of combined EPS/ \LaTeX or PDF/ \LaTeX graphics, just as easy as `\includegraphics`, and removing the limitations mentioned above.

2 Usage

2.1 The macro

`\includecombinedgraphics` The macro `\includecombinedgraphics[<option list>]{<graphics file>}` is used to include a combined EPS/ \LaTeX or PDF/ \LaTeX file, where *<graphics file>* is the name of the \LaTeX part of the graphics (*without* `.tex` extension) and *<option list>* is an optional list of *<key>=<value>* pairs (see below).

2.2 Basic macro options

The following options are processed first. If an option is specified several times, the last appearance will overwrite the previous ones.

- `textfont=<value>` one or more commands for reformatting the text in \LaTeX part (e.g., `textfont=\Large\bfseries` for large bold-face font or `textfont={}` for default font of graphics overwriting package option)
- `textcolor=<value>` one or more commands for recoloring the text in \LaTeX part (e.g., `textcolor=\color{red}` for red text or `textcolor={}` for default text color of graphics overwriting package option)
- `vecscale=<value>` scaling factor for rescaling the graphics without affecting the font of the \LaTeX part (Note that this will overwrite `vecwidth` or `vecheight`, if they are specified before.)
- `vecwidth=<value>` analogous to `vecscale`, but the scaling factor will be calculated such that the new width of the total graphics will be *<value>*; the aspect ratio will be preserved (Note that this will overwrite `vecscale` or `vecheight`, if they are specified before.)
- `vecheight=<value>` analogous to `vecscale`, but the scaling factor will be calculated such that the new (total) height of the total graphics will be *<value>*; the aspect ratio will be preserved (Note that this will overwrite `vecscale` or `vecwidth`, if they are specified before.)
- `vecinclude=<value>` generate code to include the vector graphics part; *<value>*s are: `auto` to automatically determine if code is required (default), `true/false` to always/never generate

<code>vecfirst=<i>value</i></code>	that code, or <code>overwrite</code> to disable any inclusion of a vector graphics part in the <code>L^AT_EX</code> part and generate new code
<code>vecfile=<i>value</i></code>	inclusion order of vector graphics part; valid <i>value</i> s are: <code>true</code> (default) or <code>false</code> to include the vector graphics part before or after the <code>L^AT_EX</code> part, respectively
	filename of vector graphics part, if different from filename of <code>L^AT_EX</code> part (except for extension); implies <code>vecinclude=overwrite</code>

Note: If the vector graphics part is already included by the `LATEX` part, `vecinclude` and `vecscale` will require the vector graphics part to be included by an `\includegraphics` command for proper functioning. Other graphics inclusion macros will not be detected. Furthermore, `vecscale` also requires that the `LATEX` part puts its text inside a `picture` environment whose scaling is set with `\setlength{\unitlength}{some length}`. These two prerequisites are fulfilled by graphics exported from Gnuplot and Xfig.

2.3 Extended macro options

The following options are processed *after* the basic options *in the order* of their appearances and affect the *whole* graphics (`LATEX` and vector graphics part). They may be specified several times.

Rotating

Rotating is basically performed with the means of `\rotatebox` from the `graphics/graphicsx` package.

`angle=value` angle of rotation

The following options can be used to fine adjust the effect of a rotation and apply to only the next `angle` value specified in the options list.

<code>origin=<i>value</i></code>	up to two of <code>lrctbB</code> (B stands for the baseline)
<code>x=<i>value</i></code>	the x coordinate of the center of rotation
<code>y=<i>value</i></code>	the y coordinate of the center of rotation
<code>units=<i>value</i></code>	units of rotation angle (e. g., -360 denotes clockwise rotation)

Rescaling

Rescaling is performed with the means of `\scalebox` from the `graphics` package.

<code>scale=<i>value</i></code>	scaling factor
<code>hscale=<i>value</i></code>	horizontal scaling factor
<code>vscale=<i>value</i></code>	vertical scaling factor

Note: A scaling via `hscale` and `vscale` will only be performed, if a pair of both values is specified.

Resizing

Resizing is performed with the means of `\resizebox` from the `graphics` package.

<code>height=<i>value</i></code>	height of graphics
<code>totalheight=<i>value</i></code>	height + depth of graphics
<code>width=<i>value</i></code>	width of graphics
<code>keepaspectratio</code>	if only one of the values for <code>height/totalheight</code> or <code>width</code> is specified, the other one will be calculated to keep the aspect ratio

Note: Resizing will only be performed, if a pair of `height/totalheight` and `width` is specified or one of them is specified followed by `keepaspectratio`.

2.4 Package options

The following options can be passed to the package as defaults for the macro options. If an option is specified several times, the last appearance will overwrite the previous ones.

<code>textfontcmd=<i>value</i></code>	name of a command (<i>without</i> leading backslash) for re-formatting text in L ^A T _E X part
<code>textcolorcmd=<i>value</i></code>	name of a command (<i>without</i> leading backslash) for re-coloring text in L ^A T _E X part
<code>vecscale=<i>value</i></code>	see section 2.2
<code>vecwidth=<i>value</i></code>	see section 2.2; in contrast to the macro options, you must not use a length as <i>value</i> here (workaround: use <code>\the\somelength</code>)
<code>vecheight=<i>value</i></code>	see section 2.2; in contrast to the macro options, you must not use a length as <i>value</i> here (workaround: use <code>\the\somelength</code>)
<code>vecinclude=<i>value</i></code>	see section 2.2
<code>vecfirst=<i>value</i></code>	see section 2.2

3 Bugs, problems, and suggestions

Please report bugs and problems or send suggestions for this package to Christian Schneider. Check for updates before reporting bugs at the website mentioned above.

4 Implementation

Load required packages for graphics, color and *key=value* pairs:

```
1 \RequirePackage{keyval}
2 \RequirePackage{graphicx}
3 \RequirePackage{color}
```

4.1 Package options

This if is required to remember the inclusion order. The macro specifies the package option from `vecscale`, `vecwidth`, `vecheight` used last.

```
4 \newif\ifcgr@vecfirst
5 \def\cgr@vecscale@type{0}
```

Definitions of package options as $\langle key \rangle = \langle value \rangle$ pairs. The $\langle value \rangle$ s are saved in the specified macros that are undefined by default.

```
6 \newcommand*\cgr@defopts{\define@key{cgr}}
7 \newcommand*\cgr@setopts{\setkeys{cgr}}
8 \cgr@defopts{textfontcmd}{\gdef\cgr@textfont@default{\@nameuse{#1}}}
9 \cgr@defopts{textcolorcmd}{\gdef\cgr@textcolor@default{\@nameuse{#1}}}
10 \cgr@defopts{vecscale}{\gdef\cgr@vecscale@default{#1}%
11   \gdef\cgr@vecscale@type{0}}
12 \cgr@defopts{vecwidth}{\gdef\cgr@vecwidth@default{#1}%
13   \gdef\cgr@vecscale@type{1}}
14 \cgr@defopts{vecheight}{\gdef\cgr@vecheight@default{#1}%
15   \gdef\cgr@vecscale@type{2}}
16 \cgr@defopts{vecinclude}{\gdef\cgr@vecinclude@default{#1}}
17 \cgr@defopts{vecfirst}{\gdef\cgr@vecfirst@default{#1}}
```

Next the package options are processed.

```
18 \DeclareOption*{%
19   \expandafter\cgr@setopts\expandafter{\CurrentOption}%
20 }
21 \ProcessOptions\relax
```

4.2 Basic macro options

Now the $\langle key \rangle = \langle value \rangle$ pairs for the optional argument of `\includecombinedgraphics` are defined. The following two macro are just shortcuts for this purpose.

```
22 \newcommand*\cgr@defopts@combgrphcs{\define@key{cgr@combgrphcs}}
23 \newcommand*\cgr@setopts@combgrphcs{\setkeys{cgr@combgrphcs}}
```

The basic macro options set some macros that are executed while tinkering the graphics from its two parts.

`textfont` and `textcolor` set `\cgr@textfont` and `\cgr@textcolor` macro, respectively, that will be applied to the text in the \LaTeX part of graphics and disable further formatting of fonts and colors inside the \LaTeX part by overwriting some macros afterwards (if not empty).

```
24 \cgr@defopts@combgrphcs{textfont}{%
25   \def\@tempa{#1}%
26   \ifx\@tempa\@empty%
27     \def\cgr@textfont{}%
28   \else%
29     \def\cgr@textfont{%
30       #1%
31       \def\SetFigFont{\cgr@gobblefive}%      %% Xfig <= 3.2.4
32       \def\SetFigFontNFSS{\cgr@gobblefive}% %% Xfig >= 3.2.5-alpha
33     }%
34   \fi%
35 }
36 \cgr@defopts@combgrphcs{textcolor}{%
```

```

37 \def\@tempa{#1}%
38 \ifx\@tempa\@empty%
39   \def\cgr@textcolor{}%
40 \else%
41   \def\cgr@textcolor{%
42     #1%
43   \def\color{\cgr@gobble@optone}%
44   }%
45 \fi%
46 }

```

`vecscale` sets the macro `\cgr@vecscale` to the scaling factor. Furthermore, it overwrites `\setlength` and `\includegraphics` in order to apply the scaling factor to the `picture` environment inside the L^AT_EX part (via `\setlength{\unitlength}...`) and to the `\includegraphics` command for inclusion of the vector graphics part, respectively.

```

47 \cgr@defopts@combgrphcs{vecscale}{%
48   \def\cgr@vecscale{#1}%
49   \def\setlength{\cgr@setlength}%
50   \def\includegraphics{\cgr@includegraphics}%
51 }

```

`vecwidth` and `vecheight`, respectively, calculate the scaling factor for `vecscale` such that the width or (total) height of the vector graphics part equals the specified *<dimen>*. `\vecscale` is set to `-1` to indicate that the factor shall be calculated.

```

52 \cgr@defopts@combgrphcs{vecwidth}{%
53   \cgr@vecscale@wd{\cgr@texfile}{\cgr@vecfile}{#1}%
54   \def\setlength{\cgr@setlength}%
55   \def\includegraphics{\cgr@includegraphics}%
56 }
57 \cgr@defopts@combgrphcs{vecheight}{%
58   \cgr@vecscale@ht{\cgr@texfile}{\cgr@vecfile}{#1}%
59   \def\setlength{\cgr@setlength}%
60   \def\includegraphics{\cgr@includegraphics}%
61 }

```

`vecinclude` switches between the different *<value>*s by setting `\cgr@vecinclude` to one of the for macros `\cgr@requires@graphics<value>`.

```

62 \cgr@defopts@combgrphcs{vecinclude}{%
63   \def\cgr@vecinclude{\@nameuse{cgr@requires@graphics#1}}%
64 }

```

`vecfirst` switches between the *<value>*s `true` and `false` by setting `\cgr@vecfirsttrue` or `\cgr@vecfirstfalse`, respectively.

```

65 \cgr@defopts@combgrphcs{vecfirst}{%
66   \@nameuse{cgr@vecfirst#1}%
67 }

```

`vecfile` resets `\cgr@vecfile` that defaults to the mandatory argument of `\includecombinedgraphics` to something else *and* sets `vecinclude` to `overwrite`.

```

68 \cgr@defopts@combgrphcs{vecfile}{%
69   \def\cgr@vecfile{#1}%
70   \cgr@setopts@combgrphcs{vecinclude=overwrite}%
71 }

```

4.3 Extended macro options

`\includecombinedgraphics` first tinkers the graphics applying the basic macro options and saves the result in `\cgr@curr@pic`. The extended macro options will redefine `\cgr@curr@pic` in the order of their occurrence and put a `\rotatebox`, `\scalebox` or `\resizebox` around it, if all information is already available to do so: `\cgr@curr@pic` \rightarrow `\dots\box{\cgr@curr@pic}`. Otherwise the *value* will be saved in a macro for later processing (e. g., if `hscale` is set without `vscale` being set before).

```

72 \cgr@defopts@combgrphcs{angle}{%
73   \ifx\cgr@curr@scaleopts\@empty%
74     \cgr@raddto@macro[groupfirst]{\cgr@curr@pic}{\rotatebox{#1}}%
75   \else%
76     \cgr@raddto@macro[groupfirst]{\cgr@curr@pic}{}%
77     \cgr@raddto@macro[expand]{\cgr@curr@pic}{\cgr@curr@scaleopts}%
78     \cgr@raddto@macro{\cgr@curr@pic}{\rotatebox[]}%
79     \def\cgr@curr@scaleopts{}%
80   \fi%
81 }
82 \cgr@defopts@combgrphcs{origin}{\cgr@addto@macro{\cgr@curr@scaleopts}{%
83   origin=#1,}%
84 }
85 \cgr@defopts@combgrphcs{x}{\cgr@addto@macro{\cgr@curr@scaleopts}{x=#1,}}
86 \cgr@defopts@combgrphcs{y}{\cgr@addto@macro{\cgr@curr@scaleopts}{y=#1,}}
87 \cgr@defopts@combgrphcs{units}{\cgr@addto@macro{\cgr@curr@scaleopts}{units=#1,}}
88 \cgr@defopts@combgrphcs{scale}{%
89   \cgr@raddto@macro[groupfirst]{\cgr@curr@pic}{\scalebox{#1}}%
90 }
91 \cgr@defopts@combgrphcs{hscale}{%
92   \def\cgr@curr@hscale{#1}%
93   \ifx\cgr@curr@vscale\@empty%
94     \else%
95       \cgr@raddto@macro[groupfirst]{\cgr@curr@pic}{}%
96       \cgr@raddto@macro[expand]{\cgr@curr@pic}{\cgr@curr@vscale}%
97       \cgr@raddto@macro{\cgr@curr@pic}{}%
98       \cgr@raddto@macro[expand,groupsecond]{\cgr@curr@pic}{\cgr@curr@hscale}%
99       \cgr@raddto@macro{\cgr@curr@pic}{\scalebox}%
100     \def\cgr@curr@hscale{}%
101     \def\cgr@curr@vscale{}%
102   \fi%
103 }
104 \cgr@defopts@combgrphcs{vscale}{%
105   \def\cgr@curr@vscale{#1}%
106   \ifx\cgr@curr@hscale\@empty%
107     \else%
108       \cgr@raddto@macro[groupfirst]{\cgr@curr@pic}{}%
109       \cgr@raddto@macro[expand]{\cgr@curr@pic}{\cgr@curr@vscale}%
110       \cgr@raddto@macro{\cgr@curr@pic}{}%
111       \cgr@raddto@macro[expand,groupsecond]{\cgr@curr@pic}{\cgr@curr@hscale}%
112       \cgr@raddto@macro{\cgr@curr@pic}{\scalebox}%
113     \def\cgr@curr@hscale{}%
114     \def\cgr@curr@vscale{}%
115   \fi%

```



```

116 }
117 \cgr@defopts@combgrphcs{height}{%
118   \def\cgr@curr@height{#1}%
119   \ifx\cgr@curr@width\@empty%
120     \else%
121       \cgr@raddto@macro[groupfirst,groupsecond,expand]{\cgr@curr@pic}{%
122         \cgr@curr@height%
123       }%
124       \cgr@raddto@macro[groupsecond,expand]{\cgr@curr@pic}{\cgr@curr@width}%
125       \cgr@raddto@macro{\cgr@curr@pic}{\resizebox}%
126       \def\cgr@curr@height{}%
127       \def\cgr@curr@width{}%
128     \fi%
129 }
130 \cgr@defopts@combgrphcs{totalheight}{%
131   \def\cgr@curr@totalheight{#1}%
132   \ifx\cgr@curr@width\@empty%
133     \else%
134       \cgr@raddto@macro[groupfirst,groupsecond,expand]{\cgr@curr@pic}{%
135         \cgr@curr@totalheight%
136       }%
137       \cgr@raddto@macro[groupsecond,expand]{\cgr@curr@pic}{\cgr@curr@width}%
138       \cgr@raddto@macro{\cgr@curr@pic}{\resizebox*}%
139       \def\cgr@curr@totalheight{}%
140       \def\cgr@curr@width{}%
141     \fi%
142 }
143 \cgr@defopts@combgrphcs{width}{%
144   \def\cgr@curr@width{#1}%
145   \ifx\cgr@curr@height\@empty%
146     \ifx\cgr@curr@totalheight\@empty%
147       \else%
148         \cgr@raddto@macro[groupfirst,groupsecond,expand]{\cgr@curr@pic}{%
149           \cgr@curr@totalheight%
150         }%
151         \cgr@raddto@macro[groupsecond,expand]{\cgr@curr@pic}{\cgr@curr@width}%
152         \cgr@raddto@macro{\cgr@curr@pic}{\resizebox*}%
153         \def\cgr@curr@totalheight{}%
154         \def\cgr@curr@width{}%
155       \fi%
156     \else%
157       \cgr@raddto@macro[groupfirst,groupsecond,expand]{\cgr@curr@pic}{%
158         \cgr@curr@height%
159       }%
160       \cgr@raddto@macro[groupsecond,expand]{\cgr@curr@pic}{\cgr@curr@width}%
161       \cgr@raddto@macro{\cgr@curr@pic}{\resizebox}%
162       \def\cgr@curr@height{}%
163       \def\cgr@curr@totalheight{}%
164       \def\cgr@curr@width{}%
165     \fi%
166 }
167 \cgr@defopts@combgrphcs{keepaspectratio}[]{%
168   \ifx\cgr@curr@height\@empty%
169     \ifx\cgr@curr@totalheight\@empty%

```

```

170     \ifx\cgr@curr@width\@empty%
171     \else%
172         \cgr@setopts@combgrphcs{height=!}%
173     \fi%
174 \else%
175     \cgr@setopts@combgrphcs{width=!}%
176     \fi%
177 \else%
178     \cgr@setopts@combgrphcs{width=!}%
179     \def\cgr@curr@totalheight{}%
180 \fi%
181 }

```

4.4 The macro

First of all, `\cgr@curr@pic` and the `\cgr@requires@graphics...` macros are initialized.

```

182 \long\def\cgr@curr@pic{}
183 \newif\ifcgr@requires@graphics\cgr@requires@graphicstrue
184 \def\cgr@requires@graphicsauto{}
185 \def\cgr@requires@graphicsoverwrite{%
186     \cgr@requires@graphicstrue%
187     \def\includegraphics{\cgr@gobble@optone}%
188 }

```

Now the macros and `if` for the basic macro options and for temporarily saving *<value>*s of the extended macro options are initialized.

```

189 \def\cgr@textfont{}
190 \def\cgr@textcolor{}
191 \def\cgr@vecscale{1}
192 \def\cgr@vecinclude{\cgr@requires@graphicsauto}
193 \cgr@vecfirsttrue
194 \def\cgr@texfile{}
195 \def\cgr@vecfile{}
196 \def\cgr@curr@hscale{}
197 \def\cgr@curr@vscale{}
198 \def\cgr@curr@scaleopts{}
199 \def\cgr@curr@height{}
200 \def\cgr@curr@width{}
201 \def\cgr@curr@totalheight{}

```

At this point the macro itself is defined.

```

202 \newcommand{\includecombinedgraphics}[2] [] {%
203     \begingroup%

```

To get to know, whether the vector graphics are already included in the \LaTeX parts or not, a check for an appearance of `\includegraphics` in the \LaTeX part is performed: the \LaTeX parts are expanded inside a box with `\includegraphics` being redefined to set an appropriate conditional (and eat the arguments of `\includegraphics`).

```

204     \global\cgr@requires@graphicstrue%
205     \setbox\@tempboxa\hbox{%
206         \def\includegraphics{%
207             \global\cgr@requires@graphicsfalse\cgr@gobble@optone%

```

```

208     }%
209     \input{#2}%
210 }%

```

Now the the macros resulting from the basic macro options, the vector graphics part and the \LaTeX part are added to the (empty) `\cgr@curr@pic` macro. We have to distinguish between two cases: (1) If the vector graphics part is included before the \LaTeX part (`vecfirst=true`), the vector graphics file will simply be loaded by `\includegraphics` inside a `picture` environment followed by the \LaTeX part included by `\input`. (2) If the vector graphics part is included after the \LaTeX part (`vecfirst=false`), the `\includegraphics` macro will be put inside the `picture` environment of the \LaTeX part at the offset position passed to the `picture` environment. Therefore, we will have to wrap the `\picture` macro to gain access to the offsets passed to the `picture` environment in the \LaTeX part and wrap the `\endpicture` macro to add `\includegraphics` into this `picture` environment.

```

211 \cgr@addto@macro{\cgr@curr@pic}{%
212   %% from basic macro options
213   \cgr@vecinclude\cgr@textfont\cgr@textcolor%
214   %% inclusion of vector graphics part
215   \ifcgr@requires@graphics%
216     \ifcgr@vecfirst%
217       \begin{picture}(0,0)%
218         \cgr@includegraphics@orig[scale=\cgr@vecscale]{\cgr@vecfile}%
219       \end{picture}%
220     \else%
221       \def\picture{\cgr@picture}%
222       \def\endpicture{%
223         \put(\cgr@picture@xoffs,\cgr@picture@yoffs)%
224           {\cgr@includegraphics@orig[scale=\cgr@vecscale]{\cgr@vecfile}}%
225         \cgr@endpicture@orig%
226       }%
227     \fi%
228   \fi%
229   %% inclusion of \LaTeX{} part
230   \input{#2}%
231 }%

```

Afterwards, a macro containing the name of the \LaTeX and vector graphics part, respectively, is initialized.

```

232 \def\cgr@vecfile{#2}%
233 \def\cgr@texfile{#2}%

```

If set, the package options will be processed. (Suggestions for easier handling of package options are welcome.)

```

234 \if0\cgr@vecscale@type%
235   \@ifundefined{cgr@vecscale@default}{}{%
236     \cgr@setopts@combgrphcs{vecscale=\cgr@vecscale@default}%
237   }%
238 \fi%
239 \if1\cgr@vecscale@type%
240   \@ifundefined{cgr@vecwidth@default}{}{%
241     \cgr@setopts@combgrphcs{vecwidth=\cgr@vecwidth@default}%
242   }%

```

```

243 \fi%
244 \if2\cgr@vecscale@type%
245   \@ifundefined{cgr@vecheight@default}{}{%
246     \cgr@setopts@combgrphcs{vecheight=\cgr@vecheight@default}%
247   }%
248 \fi%
249 \@ifundefined{cgr@textfont@default}{}{%
250   \cgr@setopts@combgrphcs{textfont=\cgr@textfont@default}%
251 }%
252 \@ifundefined{cgr@textcolor@default}{}{%
253   \cgr@setopts@combgrphcs{textcolor=\cgr@textcolor@default}%
254 }%
255 \@ifundefined{cgr@vecinclude@default}{}{%
256   \cgr@setopts@combgrphcs{vecinclude=\cgr@vecinclude@default}%
257 }%
258 \@ifundefined{cgr@vecfirst@default}{}{%
259   \cgr@setopts@combgrphcs{vecfirst=\cgr@vecfirst@default}%
260 }%

```

Now, the options passed the the macro are processed.

```

261 \cgr@setopts@combgrphcs{#1}%

```

Finally, the macro `\cgr@curr@pic` is complete and can be output.

```

262 \cgr@curr@pic%
263 \endgroup%
264 }

```

4.5 Helper macros

This macro eats five arguments (analogous to `\@gobble` or `\@gobbletwo` from the \LaTeX kernel).

```

265 \long\def\cgr@gobblefive#1#2#3#4#5{}

```

This macro eats all stars (if any), all arguments in square brackets (if any) and one mandatory argument.

```

266 \long\def\cgr@gobble@optone{\@ifstar{\cgr@gobble@optone}{\cgr@gobble@optone@}}
267 \long\def\cgr@gobble@optone@{\@ifnextchar [{\cgr@gobble@optone@@}{\@gobble}}
268 \long\def\cgr@gobble@optone@@[#1]{\cgr@gobble@optone@}

```

In order to rescale the `picture` environment inside the \LaTeX part, the `\setlength{\unitlength}{\langle some dimen \rangle}` must be changed to scale $\langle some dimen \rangle$ by `\cgr@vecscale` at its first appearance in the \LaTeX part. This is the `\setlength` substitute to do so.

```

269 \let\cgr@setlength@orig=\setlength
270 \def\cgr@setlength#1#2{%
271   \ifx#1\unitlength%
272     \@tempdima=#2%
273     \cgr@setlength@orig{#1}{\cgr@vecscale\@tempdima}%
274     \def\setlength{\cgr@setlength@orig}%
275   \else%
276     \cgr@setlength@orig{#1}{#2}%
277   \fi%
278 }

```

Additionally, a `scale=\cgr@vecscale` option must be passed to the `\includegraphics` macro inside the \LaTeX part. This is the substitute of `\includegraphics` for this purpose.

```

279 \let\cgr@includegraphics@orig=\includegraphics
280 \def\cgr@includegraphics{%
281   \ifstar{%
282     \cgr@includegraphics@s%
283   }{%
284     \cgr@includegraphics%
285   }%
286 }
287 \def\cgr@includegraphics@{%
288   \ifnextchar [{%
289     \cgr@includegraphics@@%
290   }{%
291     \cgr@includegraphics@orig[scale=\cgr@vecscale]%
292   }%
293 }
294 \def\cgr@includegraphics@@[#1]{%
295   \ifnextchar [{%
296     \cgr@includegraphics@@@[#1]%
297   }{%
298     \cgr@includegraphics@orig[#1,scale=\cgr@vecscale]%
299   }%
300 }
301 \def\cgr@includegraphics@@@[#1][#2]{%
302   \cgr@includegraphics@orig[#1][#2,scale=\cgr@vecscale]%
303 }
304 \def\cgr@includegraphics@s@{%
305   \ifnextchar [{%
306     \cgr@includegraphics@s@@%
307   }{%
308     \cgr@includegraphics@orig*[scale=\cgr@vecscale]%
309   }%
310 }
311 \def\cgr@includegraphics@s@@[#1]{%
312   \ifnextchar [{%
313     \cgr@includegraphics@s@@@[#1]%
314   }{%
315     \cgr@includegraphics@orig*[#1,scale=\cgr@vecscale]%
316   }%
317 }
318 \def\cgr@includegraphics@s@@@[#1][#2]{%
319   \cgr@includegraphics@orig*[#1][#2,scale=\cgr@vecscale]%
320 }

```

To play the trick of including the vector graphics part after the \LaTeX part (`vecfirst=false`), we need to know the offset passed to the picture environment in the \LaTeX part. Therefore, the `\picture` macro is redefined.

```

321 \let\cgr@picture@orig=\picture
322 \let\cgr@endpicture@orig=\endpicture
323 \long\def\cgr@picture(#1,#2){%
324   \ifnextchar({%
325     \cgr@picture@(#1,#2)%

```

```

326 }{%
327   \cgr@picture@(#1,#2)(0,0)%
328 }%
329 }
330 \def\cgr@picture@(#1,#2)(#3,#4){%
331   \def\cgr@picture@xoffs{#3}%
332   \def\cgr@picture@yoffs{#4}%
333   \cgr@picture@orig(#1,#2)(#3,#4)%
334 }

```

This macro is identical to `\l@addto@macro` from koma-script bundle. It adds the stuff passed to its second argument to the end of the macro from its first argument.

```

335 \newcommand{\cgr@addto@macro}[2]{%
336   \begingroup\toks@\expandafter{#1#2}%
337   \edef\@tempa{\endgroup\def\noexpand#1{\the\toks@}}%
338   \@tempa%
339 }

```

The next macro is similar, but adds the stuff passed to its second argument to the *beginning* of the macro from its first argument. An optional argument allows for fine tuning: A comma-separated list containing `expand` (expands the stuff from the second argument before adding it), `groupfirst`, and/or `groupsecond` (puts the stuff from the first/second argument in braces before adding) may be passed.

```

340 \newcommand*\cgr@defopts@raddto{\define@key{cgr@raddto}}
341 \newcommand*\cgr@setopts@raddto{\setkeys{cgr@raddto}}
342 \newif\ifcgr@raddto@expand\cgr@raddto@expandfalse
343 \newif\ifcgr@raddto@groupfirst\cgr@raddto@groupfirstfalse
344 \newif\ifcgr@raddto@groupsecond\cgr@raddto@groupsecondfalse
345 \cgr@defopts@raddto{expand}[true]{\@nameuse{cgr@raddto@expand#1}}
346 \cgr@defopts@raddto{groupfirst}[true]{\@nameuse{cgr@raddto@groupfirst#1}}
347 \cgr@defopts@raddto{groupsecond}[true]{\@nameuse{cgr@raddto@groupsecond#1}}
348 \newtoks\cgr@token@a
349 \newtoks\cgr@token@b
350 \newcommand{\cgr@raddto@macro}[3][ ]{%
351   \begingroup%
352     \cgr@setopts@raddto{#1}%
353     \cgr@token@a\expandafter{#2}%
354     \ifcgr@raddto@expand%
355       \cgr@token@b\expandafter{#3}%
356     \else%
357       \cgr@token@b{#3}%
358     \fi%
359     \ifcgr@raddto@groupfirst%
360       \ifcgr@raddto@groupsecond%
361         \edef\@tempa{\endgroup%
362           \def\noexpand#2{\the\cgr@token@b}{\the\cgr@token@a}}%
363         }%
364       \else%
365         \edef\@tempa{\endgroup%
366           \def\noexpand#2{\the\cgr@token@b{\the\cgr@token@a}}%
367         }%
368       \fi%
369     \else%

```

```

370     \ifcgr@raddto@groupsecond%
371     \edef\@tempa{\endgroup%
372       \def\noexpand#2{\the\cgr@token@b}\the\cgr@token@a}%
373     }%
374   \else%
375     \edef\@tempa{\endgroup%
376       \def\noexpand#2{\the\cgr@token@b\the\cgr@token@a}%
377     }%
378   \fi%
379   \fi%
380 \@tempa%
381 }

```

The way of calculating the scaling factor for `vecwidth` and `vecheight` is a simplified and adapted version of `\ratio` from the `calc.sty` package. First, the width or (total) height of the original graphics part is measured. In order to do this, the graphics are put into a box. Afterwards, the places before the decimal point are calculated by TeX's built-in integer division. Finally, the decimal places are calculated with an algorithm as division has been taught at school.

```

382 \newcount\cgr@tempdigit
383 \newcommand*\cgr@vecscale@wd[3]{%
384   \cgr@vecscale@graphicsbox{#1}{#2}%
385   \cgr@vecscale@calc{\wd\@tempboxa}{#3}%
386 }
387 \newcommand*\cgr@vecscale@ht[3]{%
388   \cgr@vecscale@graphicsbox{#1}{#2}%
389   \@tempdimc=\ht\@tempboxa%
390   \advance\@tempdimc by\dp\@tempboxa%
391   \cgr@vecscale@calc{\@tempdimc}{#3}%
392 }
393 \newcommand*\cgr@vecscale@graphicsbox[2]{%
394   \setbox\@tempboxa\hbox{%
395     \begingroup%
396       \def\setlength{\cgr@setlength@orig}%
397       \def\includegraphics{\cgr@gobble@optone}%
398       \input{#1}%
399     \endgroup%
400   }%
401 }
402 \newcommand*\cgr@vecscale@calc[2]{%
403   \@tempdimb=#1%
404   \@tempcntb=\@tempdimb%
405   \@tempdima=#2%
406   \@tempcnta=\@tempdima%
407   \cgr@tempdigit=\@tempcnta%
408   \divide\cgr@tempdigit by\@tempcntb%
409   \edef\cgr@vecscale{\the\cgr@tempdigit.}%
410   \cgr@next@digit\cgr@next@digit\cgr@next@digit%
411   \cgr@next@digit\cgr@next@digit\cgr@next@digit%
412 }
413 \newcommand*\cgr@next@digit{%
414   \multiply\cgr@tempdigit by\@tempcntb%
415   \advance\@tempcnta by-\cgr@tempdigit%
416   \multiply\@tempcnta by10%

```

```

417 \cgr@tempdigit=\@tempcnta%
418 \divide\cgr@tempdigit by\@tempcntb%
419 \edef\cgr@vecyscale{\cgr@vecyscale\the\cgr@tempdigit}%
420 }

```

Change History

v0.0.1-alpha		v0.2.0	
General: initial .dtx version	1	General: added options for scaling	
v0.0.2-alpha		vector graphics part to a certain	
General: initial release	1	width or height, respectively,	
v0.1.0-alpha		plus removed “alpha” from ver-	
General: added options for chang-		sion	1
ing the inclusion order of L ^A T _E X		v0.2.1	
and vector graphics parts	1	General: simplified a macro	1
v0.1.1-alpha		v0.2.2	
General: fix in Makefiles of package	1	General: added comments to test	
		file	1

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	B	163, 169, 179, 201
\@empty 26, 38, 73, 93,	\begin	217
106, 119, 132,		
145, 146, 168–170	C	
\@gobble	\cgr@addto@macro	..
\@ifnextchar 267, 288,	82, 85–87, 211, 335	\cgr@curr@vscale ..
295, 305, 312, 324	\cgr@curr@height 118,	... 93, 96, 101,
\@ifstar	122, 126, 145,	105, 109, 114, 197
266, 281	158, 162, 168, 199	\cgr@curr@width ...
\@ifundefined	\cgr@curr@hscale ..	. 119, 124, 127,
. 235, 240, 245,	... 92, 98, 100,	132, 137, 140,
249, 252, 255, 258	106, 111, 113, 196	144, 151, 154,
\@nameuse	\cgr@curr@pic	160, 164, 170, 200
8,	... 74, 76–78,	\cgr@defopts
\@tempa 25, 26, 37, 38,	89, 95–99, 108–	... 6,
337, 338, 361,	112, 121, 124,	8–10, 12, 14, 16, 17
365, 371, 375, 380	125, 134, 137,	\cgr@defopts@combgrphcs
\@tempboxa ... 205,	138, 148, 151,	... 22, 24, 36,
385, 389, 390, 394	152, 157, 160,	47, 52, 57, 62,
\@tempcnta	161, 182, 211, 262	65, 68, 72, 82,
. 406, 407, 415–417	\cgr@curr@scaleopts	85–88, 91, 104,
\@tempcntb	... 73, 77,	117, 130, 143, 167
. 404, 408, 414, 418	\cgr@curr@totalheight	\cgr@defopts@raddto
\@tempdima	. 131, 135, 139,	... 340, 345–347
. 272, 273, 405, 406	146, 149, 153,	\cgr@endpicture@orig
\@tempdimb ... 403, 404		... 225, 322
\@tempdimc ... 389–391		\cgr@gobble@optone .
		... 43,
		187, 207, 266, 397
		\cgr@gobble@optone@
		... 266–268

<code>\cgr@gobble@optone@@</code>	<code>\cgr@setlength</code>	<code>\cgr@vecscale@ht</code> 58, 387
. 267, 268	. . . 49, 54, 59, 270	<code>\cgr@vecscale@type</code> .
<code>\cgr@gobblefive</code> . . .	<code>\cgr@setlength@orig</code> 5, 11,
. 31, 32, 265 269,	13, 15, 234, 239, 244
<code>\cgr@includegraphics</code>	273, 274, 276, 396	<code>\cgr@vecscale@wd</code> 53, 383
. . . 50, 55, 60, 280	<code>\cgr@setopts</code> 7, 19	<code>\cgr@vecwidth@default</code>
<code>\cgr@includegraphics@</code>	<code>\cgr@setopts@combgrphcs</code> 12, 241
. 284, 287	. . . 23, 70, 172,	<code>\color</code> 43
<code>\cgr@includegraphics@@</code>	175, 178, 236,	<code>\CurrentOption</code> 19
. 289, 294	241, 246, 250,	
<code>\cgr@includegraphics@@@</code>	253, 256, 259, 261	D
. 296, 301	<code>\cgr@setopts@raddto</code>	<code>\DeclareOption</code> 18
<code>\cgr@includegraphics@orig</code> 341, 352	<code>\define@key</code> . . 6, 22, 340
. . . . 218, 224,	<code>\cgr@tempdigit</code>	<code>\divide</code> 408, 418
279, 291, 298,	. 382, 407–409,	<code>\dp</code> 390
302, 308, 315, 319	414, 415, 417–419	
<code>\cgr@includegraphics@s@</code>	<code>\cgr@texfile</code>	E
. 282, 304	. . 53, 58, 194, 233	<code>\end</code> 219
<code>\cgr@includegraphics@s@@</code>	<code>\cgr@textcolor</code>	<code>\endpicture</code> . . . 222, 322
. 306, 311	. . 39, 41, 190, 213	<code>\expandafter</code>
<code>\cgr@includegraphics@s@@@</code>	<code>\cgr@textcolor@default</code>	. 19, 336, 353, 355
. 313, 318 9, 253	
<code>\cgr@next@digit</code> . . .	<code>\cgr@textfont</code>	F
. . . . 410, 411, 413	. . 27, 29, 189, 213	<code>\fi</code> 34, 45, 80,
<code>\cgr@picture</code> . . 221, 323	<code>\cgr@textfont@default</code>	102, 115, 128,
<code>\cgr@picture@</code> 8, 250	141, 155, 165,
. . . . 325, 327, 330	<code>\cgr@token@a</code> 348, 353,	173, 176, 180,
<code>\cgr@picture@orig</code> .	362, 366, 372, 376	227, 228, 238,
. 321, 333	<code>\cgr@token@b</code>	243, 248, 277,
<code>\cgr@picture@xoffs</code> .	. 349, 355, 357,	358, 368, 378, 379
. 223, 331	362, 366, 372, 376	
<code>\cgr@picture@yoffs</code> .	<code>\cgr@vecfile</code>	G
. 223, 332	. . . 53, 58, 69,	<code>\gdef</code> 8–17
<code>\cgr@raddto@expandfalse</code>	195, 218, 224, 232	<code>\global</code> 204, 207
. 342	<code>\cgr@vecfirst@default</code>	
<code>\cgr@raddto@groupfirstfalse</code> 17, 259	H
. 343	<code>\cgr@vecfirsttrue</code> . 193	<code>\hbox</code> 205, 394
<code>\cgr@raddto@groupsecondfalse</code>	<code>\cgr@vecheight@default</code>	<code>\ht</code> 389
. 344 14, 246	
<code>\cgr@raddto@macro</code> .	<code>\cgr@vecinclude</code>	I
. 74, 76–78, 89, 63, 192, 213	<code>\if</code> 234, 239, 244
95–99, 108–112,	<code>\cgr@vecinclude@default</code>	<code>\ifcgr@raddto@expand</code>
121, 124, 125, 16, 256 342, 354
134, 137, 138,	<code>\cgr@vecscale</code>	<code>\ifcgr@raddto@groupfirst</code>
148, 151, 152,	. . 48, 191, 218, 343, 359
157, 160, 161, 350	224, 273, 291,	<code>\ifcgr@raddto@groupsecond</code>
<code>\cgr@requires@graphicsauto</code>	298, 302, 308, 344, 360, 370
. 184, 192	315, 319, 409, 419	<code>\ifcgr@requires@graphics</code>
<code>\cgr@requires@graphicsfalse</code>	<code>\cgr@vecscale@calc</code> 183, 215
. 207 385, 391, 402	<code>\ifcgr@vecfirst</code> . 4, 216
<code>\cgr@requires@graphicsoverwrite</code>	<code>\cgr@vecscale@default</code>	<code>\ifx</code> 26,
. 185 10, 236	38, 73, 93, 106,
<code>\cgr@requires@graphicstrue</code>	<code>\cgr@vecscale@graphicsbox</code>	119, 132, 145,
. . . . 183, 186, 204 384, 388, 393	146, 168–170, 271

<code>\includecombinedgraphics</code>	341, 350, 383, 387, 393, 402, 413	S
.....	3, 202	<code>\scalebox</code> .. 89, 99, 112
<code>\includegraphics</code> ..		<code>\setbox</code> 205, 394
...	50, 55, 60, 187, 206, 279, 397	<code>\SetFigFont</code> 31
<code>\input</code>	209, 230, 398	<code>\SetFigFontNFSS</code> ... 32
		<code>\setkeys</code> 7, 23, 341
		<code>\setlength</code> 49, 54, 59, 269, 274, 396
L		
<code>\LaTeX</code>	229	T
<code>\let</code> ..	269, 279, 321, 322	<code>\the</code> .. 337, 362, 366, 372, 376, 409, 419
<code>\long</code> .	182, 265–268, 323	<code>\toks@</code> 336, 337
M		R
<code>\multiply</code>	414, 416	<code>\relax</code>
		21
		<code>\RequirePackage</code> ... 1–3
N		<code>\resizebox</code>
<code>\newcommand</code> 125, 138, 152, 161
..	6, 7, 22, 23, 202, 335, 340,	<code>\rotatebox</code> 74, 78
		U
		<code>\unitlength</code> 271
		W
		<code>\wd</code> 385