

Graphics drivers for `pict2e`*

Rolf Niepraschk[†], Hubert Gäßlein, and Josef Tkadlec[‡]

2016/02/05

1 Driver files

This file implements some of the currently supported drivers for the new version of the `pict2e` package. If the driver you use is not in this list then a ‘`.def`’ file may be distributed with the `pict2e` package, or may be distributed with the standard L^AT_EX graphics bundle, or may be distributed with your driver.

If not, send us some details of the driver’s `\special` syntax, and we will try to produce a suitable file.

Note that some of these files are for graphics drivers to which we have no access, so they are untested. Please send any corrections to the latexbugs address or directly to the authors.

1.1 Template

A template for a `pict2e` driver file.

`\pIIE@mode` This macro serves as an indicator to the `pict2e` package which mode the driver supports:

- `-1` inapt/incapable (default, already set in `pict2e`)
 - `0` standard L^AT_EX only
 - `1` PostScript
 - `2` PDF
- (other values are reserved for future use)

Incapable drivers should not alter the default value given by the `pict2e` package, or set it explicitly to `-1`.

```
1 <*template>
2 \def\pIIE@mode{-1}
```

*This document corresponds to `p2e-drivers.dtx` v0.1u, dated 2016/02/05, documentation dated 2016/02/05.

[†]`Rolf.Niepraschk@gmx.de`

[‡]`j.tkadlec@email.cz`

`\pIIE@code` The `pict2e` package expects the driver file to define the `\pIIE@code` command in a suitable way.

This command should locally establish the standard PostScript/PDF coordinate system (i.e., a cartesian coordinate system with positive x-axis pointing right and positive y-axis pointing up, and with unit $1\text{ bp} = 1/72\text{ in}$), albeit with the origin at $\text{\TeX}$'s current point instead of the lower left corner of the page.

Furthermore, it should save and restore the graphics state (`gsave/grestore` in PostScript, `q/Q` in PDF.) This may be achieved by using appropriate `\special` (or `\pdfliteral`, respectively) commands.

Moreover, this command should preserve (i.e., it should not change) the current colour as defined by the user via the commands of the `color` package from the graphics bundle.

Thus, the `\Gin@PS@restored` command that various `<driver>.def` files from the graphics bundle provide should usually come close to what is expected here.

```
3 \def\pIIE@code#1{}
4 </template>
```

1.2 dvips

A `pict2e` driver file for the `dvips` driver.

`\pIIE@mode` We are about to generate PostScript code.

```
5 <*dvips>
6 \def\pIIE@mode{1}
```

`\pIIE@code` In this case the code inserted by the driver on behalf of the `\Gin@PS@restored` command performs a “0 `setgray`” operation, thus resetting any colour the user might have set by means of the `color` package. (See also $\text{\LaTeX}$ problem report graphics/3569.) We therefore have to resort to the following kludge: As long as we output only simple picture objects, our operations are “atomic.” Hence, we won’t need to set colours or gray shades within the PostScript code generated by `pict2e`; thus the offending `setgray` operator may as well be a no-op. To keep this redefinition local, we enclose the call to `\Gin@PS@restored` by a `save/restore` pair.

```
7 \def\pIIE@code#1{%
8   \Gin@PS@raw{save /setgray { pop } def}%
9   \Gin@PS@restored{#1}%
10  \Gin@PS@raw{restore}%
11 }
12 </dvips>
```

1.3 pdfTeX

A `pict2e` driver file for the `pdftex` driver.

`\pIIE@mode` We are about to generate PDF code. (Only, if `pdfTeX` is actually generating PDF; otherwise nothing will be output.)

```

13 <*pdfTeX>
14 \begingroup
15   \@ifundefined{pdfoutput}{}{%
16     \ifnum\pdfoutput>0\relax
17       \gdef\pIe@mode{2}
18     \fi
19   }
20 \endgroup

```

`\pIe@code` The save/restore operators are necessary here to prevent the change of the CTM (scaling and rotation operations) that `pict2e` inserts from propagating.

```

21 \ifcase\pIe@mode\relax \or\or
22   \def\pIe@code#1{\pdfliteral{ q #1 Q }}
23 \fi
24 </pdfTeX>

```

1.4 LuaTeX

A `pict2e` driver file for the `luatex` driver.

`\pIe@mode` We are about to generate PDF code. (Only, if `LuaTeX` is actually generating PDF; otherwise nothing will be output.)

```

25 <*luatex>
26 \begingroup
27   \ifnum\outputmode>0\relax
28     \gdef\pIe@mode{2}
29   \fi
30 \endgroup

```

`\pIe@code` The save/restore operators are necessary here to prevent the change of the CTM (scaling and rotation operations) that `pict2e` inserts from propagating.

```

31 \ifcase\pIe@mode\relax \or\or
32   \def\pIe@code#1{\pdfextension literal{ q #1 Q }}
33   \protected\edef\pIe@pdfliteral{\pdfextension literal}
34 \fi
35 </luatex>

```

1.5 VTeX

A `pict2e` driver file for the `vtex` driver.

`\pIe@mode` With `VTeX`, we should use PostScript code also for PDF mode (Email from Michael Vulis, MicroPress).

```

36 <*vtex>
37 \begingroup
38   \@ifundefined{VTeXversion}{}{%
39     \ifnum\OpMode>0\relax
40       \ifnum\OpMode<3\relax

```

```

41      \gdef\pIIE@mode{1}%
42      \fi
43      \fi
44    }
45 \endgroup

\pIIE@code Here \Gin@PS@restored suffices as provided by the graphics driver file vtex.def.
46 \ifcase\pIIE@mode\relax \or
47   \let\pIIE@code\Gin@PS@restored
48 \fi
49 \</vtex>

```

1.6 dvipdfm

A pict2e driver file for the dvipdfm driver.

```

\pIIE@mode We are about to generate PDF code.
50 <*dvipdfm>
51 \def\pIIE@mode{2}

\pIIE@code This seems to be sufficient.
52 \def\pIIE@code#1{\special{pdf: content #1}}
53 </dvipdfm>

```

1.7 dvipdfmx

A pict2e driver file for the dvipdfmx driver.

```

\pIIE@mode We are about to generate PDF code.
54 <*dvipdfmx>
55 \def\pIIE@mode{2}

\pIIE@code
56 \def\pIIE@code#1{\special{pdf: content #1}}

\pIIE@pdfliteral
57 \def\pIIE@pdfliteral#1{\special{pdf: literal #1}}
58 </dvipdfmx>

```

1.8 xetex

A pict2e driver file for the xetex driver.

```

\pIIE@mode We are about to generate PDF code.
59 <*xetex>
60 \def\pIIE@mode{2}

\pIIE@code
61 \def\pIIE@code#1{\special{pdf: literal q #1 Q}}

```

`\pIIE@pdfliteral`

```
62 \def\pIIE@pdfliteral#1{\special{pdf: literal #1}}
63 \</xetex>
```

1.9 dvipdf

A pict2e driver file for the dvipdf driver (not yet implemented).

`\pIIE@mode`

```
64 \<*dvipdf>
65 % \def\pIIE@mode{-1}
```

`\pIIE@code` This is the same as the definition for `\Gin@PS@restored` in `dvipdf.def` as defined in `drivers.dtx`! Better use the higher-level macro instead of the `\special`?

```
66 % \def\pIIE@code#1{\special{" #1}} % \Gin@PS@restored{#1}
67 \</dvipdf>
```

1.10 textures

A pict2e driver file for the textures driver (not yet implemented).

`\pIIE@mode`

```
68 \<*textures>
69 % \def\pIIE@mode{-1}
```

`\pIIE@code`

```
70 % \def\pIIE@code#1{}
71 \</textures>
```

1.11 dvipsone

A pict2e driver file for the dvipsone driver (not yet implemented).

`\pIIE@mode`

```
72 \<dvipsone>
73 % \def\pIIE@mode{-1}
```

`\pIIE@code`

```
74 % \def\pIIE@code#1{}
75 \</dvipsone>
```

1.12 pctexps

A pict2e driver file for the pctexps driver (not yet implemented).

`\pIIE@mode`

```
76 \<pctexps>
77 % \def\pIIE@mode{-1}
```

`\pIIE@code`

```
78 % \def\pIIE@code#1{}
79 \</pctexps>
```

1.13 `pctex32`

A `pict2e` driver file for the `pctex32` driver (not yet implemented).

`\pIIE@mode`

```
80 \<*pctex32>
81 % \def\pIIE@mode{-1}
```

`\pIIE@code`

```
82 % \def\pIIE@code#1{}
83 \</pctex32>
```

2 A Sample Configuration File

This one is taken from `color.cfg` of the `teTeX`/`TeXlive` distributions.

```
84 \<*cfg>
85 %% Select an appropriate default driver.
86 \begingroup
87   \chardef\x=0 %
88   % check luatex
89   \ifx\luatexversion\undefined
90     % check pdfTeX
91     \@ifundefined{pdfoutput}{-}{%
92       \ifcase\pdfoutput
93       \else
94         \chardef\x=1 %
95       \fi
96     }%
97     % check VTeX
98     \@ifundefined{OpMode}{-}{%
99       \chardef\x=2 %
100     }%
101     % check XeTeX
102     \@ifundefined{XeTeXrevision}{-}{%
103       \chardef\x=3 %
104     }%
105   \else
106     \ifnum\luatexversion>85
107       \ifnum\outputmode>0
108         \chardef\x=4\relax
109       \fi
110     \else
111       \ifnum\pdfoutput>0
112         \chardef\x=1\relax
```

```

113     \fi
114     \fi
115 \fi
116 \expandafter\endgroup
117 \ifcase\x
118   % 0: default case
119   \ExecuteOptions{dvips}%
120 \or
121   % 1: pdfTeX is running in pdf mode
122   \ExecuteOptions{pdftex}%
123 \or
124   % 2: VTeX is running
125   \ExecuteOptions{vtex}%
126 \or
127   % 3: XeTeX is running
128   \ExecuteOptions{xetex}%
129 \or
130   % 4: LuaTeX (> 0.85) is running
131   \ExecuteOptions{luatex}%
132 \fi

You can also specify other options to the pict2e package in the configuration file.
For example, if you prefer PSTricks-like arrows, just uncomment the line below.
133 %% \ExecuteOptions{pstarrows}
134 </cfg>

```