

The pgf-cmykshadings package

David Purton*

2018/10/24 v1.1a

Abstract

The `pgf-cmykshadings` package provides support for CMYK and grayscale shadings for the `pgf` package. By default `pgf` only supports RGB shadings. `pgf-cmykshadings` attempts to produce shadings consistent with the currently selected `xcolor` colour model. The `rgb`, `cmyk`, and `gray` colour models from the `xcolor` package are supported.

Contents

1	Introduction	2
2	Acknowledgements	2
3	Bug Reports and Feature Requests	2
4	Documentation	2
4.1	Basic Usage	2
4.1.1	Package options	3
4.1.2	Load order	3
4.1.3	Colour models	3
4.1.4	Functional shadings	4
4.2	Main Interface	4
4.2.1	Declaring shadings	4
4.2.2	Using shadings	5
4.2.3	Utility functions	5
5	Implementation	7
5.1	Main Package	7
5.2	Drivers	16
	Change History	31

*Email: dcpurton@marshwiggles.net

1 Introduction

The `pgf` package, and other packages built on top of it, only support RGB shadings (colour gradients). This means that printing applications requiring CMYK shadings can not easily be produced. It also can lead to unexpected colour mismatches in documents when attempting to define a shading from colours defined in CMYK. This can occur when the `natural` colour model of the `xcolor` package is in use and colours like cyan and magenta are defined as CMYK. An attempt to produce a shading using these colours will be silently converted to RGB, but RGB cyan and RGB magenta look significantly different from CMYK cyan and magenta. This is a significant cause of confusion for end users.

The following example illustrates this problem and the corresponding solution provided by the `pgf-cmykshadings` package.

```
\begin{tikzpicture}
  \fill[cyan] (0,0) rectangle (1,1);
  \shade[left color=cyan, right color=magenta]
    (1.25,0) rectangle (3.75,1);
  \fill[magenta] (4,0) rectangle (5,1);
\end{tikzpicture}
```

`pgf` behaviour:



`pgf-cmykshadings` behaviour:



2 Acknowledgements

Substantial parts of the code for the `pgf-cmykshadings` package are taken from the `pgf` package file `pgfcoreshade.code.tex` along with the driver files `pgfsys-*.def` copyright © 2006 Till Tantau and then slightly modified to support CMYK and grayscale shadings.

3 Bug Reports and Feature Requests

Bug reports and feature requests can be made at the `pgf-cmykshadings` package GitHub repository. See <https://github.com/dcpurton/pgf-cmykshadings>.

4 Documentation

4.1 Basic Usage

All that is required to use CMYK shadings instead of RGB shadings is to include the package in your document preamble:

```
\usepackage{pgf-cmykshadings}
```

However, there are some caveats in using the package, which are outlined below.

4.1.1 Package options

`pgd-cmykshadings` supports the following package options:

`cmyk` (default) to use CMYK shadings when the `xcolor` package `natural` colour model is in use.

`rgb` to use RGB shadings when the `xcolor` package `natural` colour model is in use.

4.1.2 Load order

`pgf-cmykshadings` should be loaded *before* any shadings are defined otherwise these will be defined as RGB. This means you should load `pgf-cmykshadings` before (for example) `tikz` and `beamer`.

If you want to pass custom options to `xcolor` (e.g., a colour model or set of named colours), you should load `pgf-cmykshadings` *after* `xcolor` or use `\PassOptionsToPackage` *before* loading `pgf-cmykshadings`.

4.1.3 Colour models

`pgf-cmykshadings` attempts to produce shadings consistent with the currently selected `xcolor` package colour model. The `rgb`, `cmyk`, and `gray` colour models from the `xcolor` package are supported.

Note: The colour model chosen for a shading is based on the `xcolor` colour model *at the time the shading is created*. This is either when `\pgfdeclare*shading` is called with no optional argument or when `\pgfuseshading` is called if `\pgfdeclare*shading` was called with an optional argument.

If the `xcolor` package `natural` colour model is in use then the shading colour model will be CMYK by default (equivalent to passing the `cmyk` option to the `pgf-cmykshadings` package). RGB shadings can be output by default instead by passing the `rgb` option to the `pgf-cmykshadings` package.

In practice this means that if you are using the `natural` colour model of the `xcolor` package you can still get mismatched colours if you, for example, create a shading from green (which is defined as RGB) to magenta (which is defined as CMYK). The shading has to pick one colour model and will look different to one of the solid colours.

For this reason it is recommended to always load the `xcolor` package before the `pgf-cmykshadings` package with either the `rgb`, `cmyk`, or `gray` options to avoid colour surprises.

```
\begin{tikzpicture}
\fill[green] (0,0) rectangle (1,1);
\shade[left color=green, right color=magenta]
(1.25,0) rectangle (3.75,1);
\fill[magenta] (4,0) rectangle (5,1);
\end{tikzpicture}
```

`xcolor natural` colour model:



`xcolor cmyk` colour model:



`xcolor rgb` colour model:



`xcolor gray` colour model:



4.1.4 Functional shadings

By nature, the PostScript® code used to generate functional shadings must output either RGB or CMYK data. For this reason, `\pgfdeclarefunctionalshading` is *not* portable across colour models.

Take particular care that the same colour model is in use at declaration time and use time for functional shadings declared with an optional argument as otherwise the PostScript® data will not match the declared colour space and you will end up with a malformed PDF.

This also means that you should *not* use the functional shadings from the `tikz` shading library (`bilinear interpolation`, `color wheel`, `color wheel black center`, `color wheel white center`, and `Mandelbrot set`) except when the `xcolor rgb` colour model is in use, otherwise you will end up with a malformed PDF.

Having said this, it *is* possible to create portable functional shadings by providing conditional code to append colour transformations to the PostScript® data. A variety of `\pgffuncshading*to*` (e.g., `\pgffuncshadingrgbtocmyk`) macros along with `\ifpgfshadingmodel*` (e.g., `\ifpgfshadingmodelcmyk`) conditionals are provided to assist with these transformations.

```
\pgfdeclarefunctionalshading[black]{twospots}
  {\pgfpointorigin}{\pgfpoint{3.5cm}{3.5cm}}{ }{
  2 copy
  45 sub dup mul exch
  40 sub dup mul 0.5 mul add sqrt
  dup mul neg 1.0005 exch exp 1.0 exch sub
  3 1 roll
  70 sub dup mul .5 mul exch
  70 sub dup mul add sqrt
  dup mul neg 1.002 exch exp 1.0 exch sub
  1.0 3 1 roll
  \ifpgfshadingmodelcmyk
    \pgffuncshadingrgbtocmyk
  \fi
  \ifpgfshadingmodelgray
    \pgffuncshadingrgbtogray
  \fi
}
```

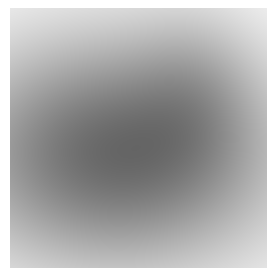
xcolor cmyk model:



xcolor rgb model:



xcolor gray model:



4.2 Main Interface

4.2.1 Declaring shadings

The four standard `pgf` functions for declaring shadings are supported as documented in the `pgf` manual.

There is one extension provided by the `pgf-cmykshadings` package. It is possible to specify CMYK colours directly in the colour specification argument using a syntax analogous to the RGB, Gray, and named colours already supported by the `pgf` package. i.e., `cmk(\position)=(\C),(\M),(\Y),(\K))`.

Shadings declared *without* an optional argument are created immediately in the currently active `xcolor` colour model. Shadings declared *with* and optional argument are created at the time they are actually used (using `\pgfuseshading`).

`\pgfdeclarehorizontalshading` `\pgfdeclarehorizontalshading[\color list]{\shading name}{\shading height}{\color specification}`

Declare a horizontal shading.

`\pgfdeclareverticalshading` `\pgfdeclareverticalshading[\color list]{\shading name}{\shading width}{\color specification}`

Declare a vertical shading.

`\pgfdeclareradialshading` `\pgfdeclareradialshading[\color list]{\shading name}{\center point}{\color specification}`

Declare a radial shading.

`\pgfdeclarefunctionalshading` `\pgfdeclarefunctionalshading[\color list]{\shading name}{\lower left corner}{\upper right corner}{\init code}{\type 4 function}`

Declare a functional shading.

4.2.2 Using shadings

Shadings are used as documented in the `pgf` manual.

`\pgfuseshading` `\pgfuseshading{\shading name}`

Use a previously declared shading. If the specified shading was declared with an optional argument, then the shading will be created at this point in the currently active `xcolor` colour space.

`\pgfshadepath` `\pgfshadepath{\shading name}{\angle}`

Shade the currently active `pgf` path using the specified shading at the specified angle.

`\pgfadditionalshadetransform` `\pgfadditionalshadetransform{\transformation}`

This command is used to specify an additional transformation that should be applied to shadings when `\pgfshadepath` used.

4.2.3 Utility functions

The following functions are mainly useful for in declaring functional shadings.

`\pgfshadecolor torgb` `\pgfshadecolor torgb { $\langle color name \rangle$ }{ $\langle macro \rangle$ }`

This command takes $\langle color name \rangle$ as input and stores the colour's red/green/blue components as real numbers between 0.0 and 1.0 separated by spaces (which is exactly what you need if you want to push it on a stack) in $\langle macro \rangle$. This macro can then be used inside the $\langle type 4 function \rangle$ argument for `\pgfdeclarefunctionalshading`.

In addition, three macros suffixed with **red**, **green** and **blue** are defined, which store the individual components of $\langle color name \rangle$. These can also be used in the $\langle type 4 function \rangle$ argument.

`\pgfshadecolor tocmyk` `\pgfshadecolor tocmyk { $\langle color name \rangle$ }{ $\langle macro \rangle$ }`

This command is analogous to `\pgfshadecolor torgb` , but stores the colour's cyan/magenta/yellow/black components. Four macros suffixed with **cyan**, **magenta**, **yellow**, and **black** are also defined.

`\pgfshadecolor tgray` `\pgfshadecolor tgray { $\langle color name \rangle$ }{ $\langle macro \rangle$ }`

This command is analogous to `\pgfshadecolor torgb` , but stores the colour's gray component. Although it's not needed, for consistency a second macro suffixed with **gray** is also defined.

`\pgffuncshading rgbtocmyk` `\pgffuncshading rgbtocmyk`

Within the $\langle type 4 function \rangle$ argument of `\pgfdeclarefunctionalshading`, this command can be used to convert the top 3 elements on the stack from RGB to CMYK. In combination with the `\ifpgfshadingmodel cmyk` conditional this macro can be used to make functional shading declarations more portable across colour models.

`\pgffuncshading rgbtogray` `\pgffuncshading rgbtogray`

Within the $\langle type 4 function \rangle$ argument of `\pgfdeclarefunctionalshading`, this command can be used to convert the top 3 elements on the stack from RGB to grayscale. In combination with the `\ifpgfshadingmodel gray` conditional this macro can be used to make functional shading declarations more portable across colour models.

`\pgffuncshading cmyktorgb` `\pgffuncshading cmyktorgb`

Within the $\langle type 4 function \rangle$ argument of `\pgfdeclarefunctionalshading`, this command can be used to convert the top 4 elements on the stack from CMYK to RGB. In combination with the `\ifpgfshadingmodel rgb` conditional this macro can be used to make functional shading declarations more portable across colour models.

`\pgffuncshading cmyktogray` `\pgffuncshading cmyktogray`

Within the $\langle type 4 function \rangle$ argument of `\pgfdeclarefunctionalshading`, this command can be used to convert the top 4 elements on the stack from CMYK to grayscale. In combination with the `\ifpgfshadingmodel gray` conditional this macro can be used to make functional shading declarations more portable across colour models.

`\pgffuncshading graytorgb` `\pgffuncshading graytorgb`

Within the $\langle type 4 function \rangle$ argument of `\pgfdeclarefunctionalshading`, this command can be used to convert the top element on the stack from grayscale to RGB. In combination with the `\ifpgfshadingmodel rgb` conditional this macro can be used to make functional shading declarations more portable across colour models.

`\pgffuncshadinggraytocmyk`

`\pgffuncshadinggraytocmyk`

Within the *<type 4 function>* argument of `\pgfdeclarefunctionalshading`, this command can be used to convert the top element on the stack from grayscale to CMYK. In combination with the `\ifpgfshadingmodelcmyk` conditional this macro can be used to make functional shading declarations more portable across colour models.

`\ifpgfshadingmodelrgb`

`\ifpgfshadingmodelrgb`

Within the *<type 4 function>* argument of `\pgfdeclarefunctionalshading`, this command can be used to test if the `xcolor` colour model is `rgb` *at the time the shading is created*. This can be used to ensure that the data output in the *<type 4 function>* correctly matches the active colour model.

`\ifpgfshadingmodelcmyk`

`\ifpgfshadingmodelcmyk`

Within the *<type 4 function>* argument of `\pgfdeclarefunctionalshading`, this command can be used to test if the `xcolor` colour model is `cmyk` *at the time the shading is created*. This can be used to ensure that the data output in the *<type 4 function>* correctly matches the active colour model.

`\ifpgfshadingmodelgray`

`\ifpgfshadingmodelgray`

Within the *<type 4 function>* argument of `\pgfdeclarefunctionalshading`, this command can be used to test if the `xcolor` colour model is `gray` *at the time the shading is created*. This can be used to ensure that the data output in the *<type 4 function>* correctly matches the active colour model.

5 Implementation

5.1 Main Package

```
1 <*package>
2 \ProvidesPackage{pgf-cmykshadings}%
3 [2018/10/24
4   CMYK and grayscale shadings support for PGF (DCP)]
5 \RequirePackage{pgf}
```

Replace dependence on `\pgf@convertrgbstring` and `\pgf@rgbconv` with generic macros `\pgf@convertstring` and `\pgf@conv`.

```
6 \def\pgf@parsefunc#1{%
7   \edef\temp{#{#1}}%
8   \expandafter\pgf@convertstring\temp%
9   \edef\temp{#\pgf@conv}%
10  \expandafter\pgf@@parsefunc\temp}
```

Replace RGB parsing macros with new macros selected on the basis of the current colour space (`\pgf@shading@mode`).

```
11 \def\pgf@@parsefunc#1{%
12   \let\pgf@bounds=\pgfutil@empty%
13   \let\pgf@funcs=\pgfutil@empty%
14   \let\pgf@psfuncs=\pgfutil@empty%
15   \let\pgf@encode=\pgfutil@empty%
16   \let\pgf@sys@shading@ranges=\pgfutil@empty%
```

```

17 \pgf@sys@shading@range@num=0\relax%
18 \csname pgf@parsefirst\pgf@shading@model\endcsname[#1; ]%
19 \csname pgf@parselastdom\pgf@shading@model\endcsname[#1; ]%
20 \csname pgf@parsemid\pgf@shading@model\endcsname[#1; ]%
21 \ifx\pgf@bounds\pgfutil@empty%
22   \edef\pgf@pdfparseddomain{0 1}%
23   \edef\pgf@pdfparsedfunction{\pgf@singlefunc\space}%
24 \else%
25   \edef\pgf@pdfparseddomain{\pgf@doma\space\pgf@domb}%
26   \edef\pgf@pdfparsedfunction{%
27     << /FunctionType 3 /Domain [\pgf@doma\space\pgf@domb] /Functions
28     [\pgf@funcs\space] /Bounds [\pgf@bounds] /Encode [0 1 \pgf@encode]
29     >> }% <<
30 \fi%
31 \xdef\pgf@psfuncs{\pgf@psfuncs}%
32 }

```

Define RGB parsing macros.

```

33 \let\pgf@parsefirstrgb\pgf@parsefirst
34 \let\pgf@parselastdomrgb\pgf@parselastdom
35 \let\pgf@parsemidrgb\pgf@parsemid
36 \let\pgf@parserestrgb\pgf@parserest

```

Define new CMYK parsing macros.

```

37 \def\pgf@parsefirstcmyk[cmyk(#1)=(#2,#3,#4,#5)#6]{%
38   \pgfmathsetlength\pgf@x{#1}%
39   \edef\pgf@sys@shading@start@pos{\the\pgf@x}%
40   \pgf@sys@bp@correct\pgf@x%
41   \edef\pgf@doma{\pgf@sys@tonumber{\pgf@x}}%
42   \edef\pgf@prevx{\pgf@sys@tonumber{\pgf@x}}%
43   \pgf@getcmyktuplewithmixin{#2}{#3}{#4}{#5}%
44   \edef\pgf@sys@shading@start@cmyk{\pgf@sys@cmyk}%
45   \let\pgf@sys@prevcolor=\pgf@sys@shading@start@cmyk%
46   \let\pgf@sys@prevpos=\pgf@sys@shading@start@pos%
47   \edef\pgf@prevcolor{\pgf@cmyk}%
48   \edef\pgf@firstcolor{\pgf@cmyk}}
49 \def\pgf@parselastdomcmyk[cmyk(#1)=(#2,#3,#4,#5); {%
50   \pgfutil@ifnextchar){%
51     \pgfmathsetlength\pgf@x{#1}%
52     \edef\pgf@sys@shading@end@pos{\the\pgf@x}%
53     \pgf@max=\pgf@x\relax%
54     \pgf@sys@bp@correct\pgf@x%
55     \edef\pgf@domb{\pgf@sys@tonumber{\pgf@x}}%
56     \pgf@getcmyktuplewithmixin{#2}{#3}{#4}{#5}%
57     \edef\pgf@sys@shading@end@cmyk{\pgf@sys@cmyk}%
58     \pgfutil@gobble}{\pgf@parselastdomcmyk[]}}
59 \def\pgf@parsemidcmyk[cmyk(#1)=(#2,#3,#4,#5); {\pgf@parserestcmyk[]
60 \def\pgf@parserestcmyk[cmyk(#1)=(#2,#3,#4,#5); {%
61   \advance\pgf@sys@shading@range@num by1\relax%
62   \pgfutil@ifnextchar){%
63     \pgf@getcmyktuplewithmixin{#2}{#3}{#4}{#5}%
64     \edef\pgf@singlefunc{\space%
65     << /FunctionType 2 /Domain [0 1] /CO
66     [\pgf@prevcolor] /C1 [\pgf@cmyk] /N 1 >> }% <<
67     \edef\pgf@funcs{\pgf@funcs\space%

```



```

68     << /FunctionType 2 /Domain [\pgf@doma\space\pgf@domb] /CO
69     [\pgf@prevcolor] /C1 [\pgf@cmyk] /N 1 >> }% <<
70 \edef\pgf@psfuncs{\pgf@prevx\space
71 \pgf@cmyk\space \pgf@prevcolor\space pgfshade \pgf@psfuncs}%
72 \pgfmathsetlength\pgf@x{#1}%
73 \edef\pgf@sys@shading@ranges{\pgf@sys@shading@ranges{%
74 {\pgf@sys@prevpos}{\the\pgf@x}{\pgf@sys@prevcolor}{\pgf@sys@cmyk}}}%
75 \edef\pgf@sys@prevpos{\the\pgf@x}%
76 \let\pgf@sys@prevcolor=\pgf@sys@cmyk%
77 \pgfutil@gobble}%
78 \pgfmathsetlength\pgf@x{#1}%
79 \pgf@getcmyktuplewithmixin{#2}{#3}{#4}{#5}%
80 \edef\pgf@sys@shading@ranges{\pgf@sys@shading@ranges{%
81 {\pgf@sys@prevpos}{\the\pgf@x}{\pgf@sys@prevcolor}{\pgf@sys@cmyk}}}%
82 \edef\pgf@sys@prevpos{\the\pgf@x}%
83 \let\pgf@sys@prevcolor=\pgf@sys@cmyk%
84 \edef\pgf@psfuncs{\pgf@prevx\space \pgf@cmyk\space
85 \pgf@prevcolor\space pgfshade \pgf@psfuncs}%
86 \pgf@sys@bp@correct\pgf@x%
87 \edef\pgf@prevx{\pgf@sys@tonumber{\pgf@x}}%
88 \edef\pgf@bounds{\pgf@bounds\space\pgf@sys@tonumber{\pgf@x}}%
89 \edef\pgf@encode{\pgf@encode\space0 1}%
90 \edef\pgf@singlefunc{\space%
91 << /FunctionType 2 /Domain [0 1] /CO
92 [\pgf@prevcolor] /C1 [\pgf@cmyk] /N 1 >> }% <<
93 \edef\pgf@funcs{\pgf@funcs\space%
94 << /FunctionType 2 /Domain [\pgf@doma\space\pgf@domb] /CO
95 [\pgf@prevcolor] /C1 [\pgf@cmyk] /N 1 >> }% <<
96 \edef\pgf@prevcolor{\pgf@cmyk}%
97 \pgf@parserestcmyk[]}
98 \def\pgf@getcmyktuplewithmixin#1#2#3#4{%
99 \pgfutil@definecolor{pgfshadetemp}{cmyk}{#1,#2,#3,#4}%
100 \pgfutil@ifundefined{applycolormixins}{\applycolormixins{pgfshadetemp}}%
101 \pgfutil@extractcolorspec{pgfshadetemp}{\pgf@tempcolor}%
102 \expandafter\pgfutil@convertcolorspec\pgf@tempcolor{cmyk}{\pgf@cmykcolor}%
103 \expandafter\pgf@getcmyk@@\pgf@cmykcolor!}
104 \def\pgf@getcmyk@@#1,#2,#3,#4!{%
105 \def\pgf@cmyk{#1 #2 #3 #4}%
106 \def\pgf@sys@cmyk{{#1}{#2}{#3}{#4}}%
107 }

```

Define new grayscale parsing macros.

```

108 \def\pgf@parsefirstgray[gray(#1)=(#2)#3]{%
109 \pgfmathsetlength\pgf@x{#1}%
110 \edef\pgf@sys@shading@start@pos{\the\pgf@x}%
111 \pgf@sys@bp@correct\pgf@x%
112 \edef\pgf@doma{\pgf@sys@tonumber{\pgf@x}}%
113 \edef\pgf@prevx{\pgf@sys@tonumber{\pgf@x}}%
114 \pgf@getgraytuplewithmixin{#2}%
115 \edef\pgf@sys@shading@start@gray{\pgf@sys@gray}%
116 \let\pgf@sys@prevcolor=\pgf@sys@shading@start@gray%
117 \let\pgf@sys@prevpos=\pgf@sys@shading@start@pos%
118 \edef\pgf@prevcolor{\pgf@gray}%
119 \edef\pgf@firstcolor{\pgf@gray}}
120 \def\pgf@parselastdomgray[gray(#1)=(#2); {%

```

```

121 \pgfutil@ifnextchar}{%
122   \pgfmathsetlength\pgf@x{#1}%
123   \edef\pgf@sys@shading@end@pos{\the\pgf@x}%
124   \pgf@max=\pgf@x\relax%
125   \pgf@sys@bp@correct\pgf@x%
126   \edef\pgf@domb{\pgf@sys@tonumber{\pgf@x}}%
127   \pgf@getgraytuplewithmixin{#2}%
128   \edef\pgf@sys@shading@end@gray{\pgf@sys@gray}%
129   \pgfutil@gobble}{\pgf@parselastdomgray[]}
130 \def\pgf@parsemidgray[gray(#1)=(#2); {\pgf@parserestgray}]
131 \def\pgf@parserestgray[gray(#1)=(#2); {%
132   \advance\pgf@sys@shading@range@num by1\relax%
133   \pgfutil@ifnextchar}{%
134     \pgf@getgraytuplewithmixin{#2}%
135     \edef\pgf@singlefunc{\space%
136       << /FunctionType 2 /Domain [0 1] /CO
137       [\pgf@prevcolor] /C1 [\pgf@gray] /N 1 >> }% <<
138     \edef\pgf@funcs{\pgf@funcs\space%
139       << /FunctionType 2 /Domain [\pgf@domba\space\pgf@domb] /CO
140       [\pgf@prevcolor] /C1 [\pgf@gray] /N 1 >> }% <<
141     \edef\pgf@psfuncs{\pgf@prevx\space \pgf@gray\space
142       \pgf@prevcolor\space pgfshade \pgf@psfuncs}%
143     \pgfmathsetlength\pgf@x{#1}%
144     \edef\pgf@sys@shading@ranges{\pgf@sys@shading@ranges{%
145       {\pgf@sys@prevpos}{\the\pgf@x}{\pgf@sys@prevcolor}{\pgf@sys@gray}}}%
146     \edef\pgf@sys@prevpos{\the\pgf@x}%
147     \let\pgf@sys@prevcolor=\pgf@sys@gray%
148     \pgfutil@gobble}{%
149     \pgfmathsetlength\pgf@x{#1}%
150     \pgf@getgraytuplewithmixin{#2}%
151     \edef\pgf@sys@shading@ranges{\pgf@sys@shading@ranges{%
152       {\pgf@sys@prevpos}{\the\pgf@x}{\pgf@sys@prevcolor}{\pgf@sys@gray}}}%
153     \edef\pgf@sys@prevpos{\the\pgf@x}%
154     \let\pgf@sys@prevcolor=\pgf@sys@gray%
155     \edef\pgf@psfuncs{\pgf@prevx\space \pgf@gray\space
156       \pgf@prevcolor\space pgfshade \pgf@psfuncs}%
157     \pgf@sys@bp@correct\pgf@x%
158     \edef\pgf@prevx{\pgf@sys@tonumber{\pgf@x}}%
159     \edef\pgf@bounds{\pgf@bounds\space\pgf@sys@tonumber{\pgf@x}}%
160     \edef\pgf@encode{\pgf@encode\space0 1}%
161     \edef\pgf@singlefunc{\space%
162       << /FunctionType 2 /Domain [0 1] /CO
163       [\pgf@prevcolor] /C1 [\pgf@gray] /N 1 >> }% <<
164     \edef\pgf@funcs{\pgf@funcs\space%
165       << /FunctionType 2 /Domain [\pgf@domba\space\pgf@domb] /CO
166       [\pgf@prevcolor] /C1 [\pgf@gray] /N 1 >> }% <<
167     \edef\pgf@prevcolor{\pgf@gray}%
168     \pgf@parserestgray[]}
169 \def\pgf@getgraytuplewithmixin#1{%
170   \pgfutil@definecolor{pgfshadetemp}{gray}{#1}%
171   \pgfutil@ifundefined{applycolormixins}{\applycolormixins{pgfshadetemp}}%
172   \pgfutil@extractcolorspec{pgfshadetemp}{\pgf@tempcolor}%
173   \expandafter\pgfutil@convertcolorspec\pgf@tempcolor{gray}{\pgf@graycolor}%
174   \expandafter\pgf@getgray@@\pgf@graycolor!}

```

```

175 \def\pgf@getgray@#1!{%
176   \def\pgf@gray{#1}%
177   \def\pgf@sys@gray{#1}%
178 }

```

Define new colour space agnostic colour specification parsing macros. This includes parsing CMYK colour specifications (i.e., `color(\langle position \rangle)=(\langle C \rangle,\langle M \rangle,\langle Y \rangle,\langle K \rangle)`).

```

179 \def\pgf@convertstring#1{%
180   \def\pgf@conv{}%
181   \pgf@convert#1}%
182 }
183 \def\pgf@convert{%
184   \pgfutil@ifnextchar{\pgfutil@gobble}%done!
185   {%
186     \pgfutil@ifnextchar;{\pgf@grabsemicolor}%
187     {%
188       \pgfutil@ifnextchar c{\pgf@gobblec}%
189       {%
190         \pgfutil@ifnextchar g{\pgf@grabgray}%
191         {%
192           \pgfutil@ifnextchar o{\pgf@grabcolor}%
193           {%
194             \pgfutil@ifnextchar m{\pgf@grabcmyk}%
195             {%
196               \pgfutil@ifnextchar r{\pgf@grabrgb}%
197               {\pgferror{Illformed shading
198                 specification}\pgf@convert}%
199             }%
200           }%
201         }%
202       }%
203     }%
204   }%
205 }
206 \def\pgf@grabsemicolor;{%
207   \edef\pgf@conv{\pgf@conv; }\pgf@convert}
208 \def\pgf@gobblec c{\pgf@convert}
209 \def\pgf@savecolor#1{%
210   \pgfutil@extractcolorspec{pgf@tempcol}{\pgf@tempcolor}%
211   \expandafter\pgfutil@convertcolorspec\pgf@tempcolor
212   {\pgf@shading@model}{\pgf@color}%
213   \expandafter\pgf@convget@\expandafter{\pgf@color}{#1}%
214 }
215 \def\pgf@grabrgb rgb(#1)=(#2,#3,#4){%
216   \pgfutil@definecolor{pgf@tempcol}{rgb}{#2,#3,#4}%
217   \pgf@savecolor{#1}%
218 }
219 \def\pgf@grabcmyk myk(#1)=(#2,#3,#4,#5){%
220   \pgfutil@definecolor{pgf@tempcol}{cmyk}{#2,#3,#4,#5}%
221   \pgf@savecolor{#1}%
222 }
223 \def\pgf@grabgray gray(#1)=(#2){%
224   \pgfutil@definecolor{pgf@tempcol}{gray}{#2}%
225   \pgf@savecolor{#1}%

```

```

226 }
227 \def\pgf@grabcolor color(#1)=(#2){%
228   \pgfutil@colorlet{pgf@tempcol}{#2}%
229   \pgf@savcolor{#1}%
230 }
231 \def\pgf@convget@#1#2{%
232   \edef\pgf@conv{\pgf@conv \pgf@shading@model(#2)=(#1)}\pgf@convert}

```

New macros to convert CMYK colours to a format suitable for use in the *(type 4 function)* argument of `\pgfdeclarefunctionalshading`.

```

233 \newdimen\pgf@xd
234 \def\pgfshadecolortocmyk#1#2{%
235   \pgfutil@colorlet{pgf@tempcol}{#1}%
236   \pgfutil@extractcolorspec{pgf@tempcol}{\pgf@tempcolor}%
237   \expandafter\pgfutil@convertcolorspec\pgf@tempcolor{cmyk}{\pgf@cmykcolor}%
238   \expandafter\pgfshading@cmyk\pgf@cmykcolor\relax%
239   \edef#2{\pgf@sys@tonumber{\pgf@xa}\space\pgf@sys@tonumber{\pgf@xb}\space
240     \pgf@sys@tonumber{\pgf@xc}\space\pgf@sys@tonumber{\pgf@xd}\space}%
241   \c@pgf@counta\escapechar%
242   \escapechar-1\relax%
243   \expandafter\edef\csname\string#2cyan\endcsname{%
244     \pgf@sys@tonumber{\pgf@xa}\space}%
245   \expandafter\edef\csname\string#2magenta\endcsname{%
246     \pgf@sys@tonumber{\pgf@xb}\space}%
247   \expandafter\edef\csname\string#2yellow\endcsname{%
248     \pgf@sys@tonumber{\pgf@xc}\space}%
249   \expandafter\edef\csname\string#2black\endcsname{%
250     \pgf@sys@tonumber{\pgf@xd}\space}%
251   \escapechar\c@pgf@counta
252 }
253 \def\pgfshading@cmyk#1,#2,#3,#4\relax{%
254   \pgf@xa=#1pt%
255   \pgf@xb=#2pt%
256   \pgf@xc=#3pt%
257   \pgf@xd=#4pt%
258 }

```

New macros to convert grayscale colours to a format suitable for use in the *(type 4 function)* argument of `\pgfdeclarefunctionalshading`.

```

259 \def\pgfshadecolortogray#1#2{%
260   \pgfutil@colorlet{pgf@tempcol}{#1}%
261   \pgfutil@extractcolorspec{pgf@tempcol}{\pgf@tempcolor}%
262   \expandafter\pgfutil@convertcolorspec\pgf@tempcolor{gray}{\pgf@graycolor}%
263   \expandafter\pgfshading@gray\pgf@graycolor\relax
264   \edef#2{\pgf@sys@tonumber{\pgf@xa}\space}%
265   \c@pgf@counta\escapechar
266   \escapechar-1\relax
267   \expandafter\edef\csname\string#2gray\endcsname{%
268     \pgf@sys@tonumber{\pgf@xa}\space}%
269   \escapechar\c@pgf@counta
270 }
271 \def\pgfshading@gray#1\relax{%
272   \pgf@xa=#1pt%
273 }

```

Ensure colour model is set up based on the current xcolor colour model when declaring shadings.

```

274 \def\pgfdeclarehorizontalshading{%
275   \pgf@setup@model
276   \pgfutil@ifnextchar[%
277     \pgf@declarehorizontalshading{\pgf@declarehorizontalshading[]}]
278 \def\pgfdeclareverticalshading{%
279   \pgf@setup@model
280   \pgfutil@ifnextchar[%
281     \pgf@declareverticalshading{\pgf@declareverticalshading[]}]
282 \def\pgfdeclareradialshading{%
283   \pgf@setup@model
284   \pgfutil@ifnextchar[%
285     \pgf@declareradialshading{\pgf@declareradialshading[]}]
286 \def\pgfdeclarefunctionalshading{%
287   \pgf@setup@model
288   \pgfutil@ifnextchar[%
289     \pgf@declarefunctionalshading{\pgf@declarefunctionalshading[]}]

```

Ensure colour model is set up based on the current xcolor colour model when using shadings.

```

290 \def\pgfuseshading#1{%
291   \edef\pgf@shadingname{@pgfshading#1}%
292   \edef\pgf@shadingsavedmodel{@pgfshading#1@model}%
293   \pgf@tryextensions{\pgf@shadingname}{\pgf@alternateextension}%
294   \expandafter\pgfutil@ifundefined\expandafter{\pgf@shadingname}%
295   {\pgferror{Undefined shading "#1"}}%
296   {%
297     {%
298       \pgf@setup@model
299       \pgfutil@globalcolorsfalse
300       \def\pgf@shade@adds{}%
301       \pgfutil@ifundefined{pgf@deps\pgf@shadingname}%
302       {}%
303       {%
304         \edef\@list{\csname pgf@deps\pgf@shadingname\endcsname}%
305         \pgfutil@for\@temp:=\@list\do{%
306           {%
307             \pgfutil@ifundefined{applycolormixins}%
308             {}{\applycolormixins{\@temp}}%
309             \pgfutil@extractcolorspec{\@temp}{\pgf@tempcolor}%
310             \expandafter\pgfutil@ifundefined\expandafter{%
311               \pgf@shadingsavedmodel}%
312               {\expandafter\pgfutil@convertcolorspec\pgf@tempcolor{%
313                 \pgf@shading@model}{\pgf@color}}%
314                 {\expandafter\pgfutil@convertcolorspec\pgf@tempcolor{%
315                   \pgf@shadingsavedmodel}{\pgf@color}}%
316                 \xdef\pgf@shade@adds{\pgf@shade@adds,\pgf@color}%
317               }%
318             }%
319           }%
320           \expandafter\pgf@strip@shadename\pgf@shadingname!!%
321           \pgfutil@ifundefined{@pgfshading\pgf@basename\pgf@shade@adds!}%
322           {%

```

```

323     {%
324         \expandafter\def\expandafter\@temp\expandafter{%
325             \csname pgf@func\pgf@shadingname\endcsname}%
326         \edef\@args{\pgf@basename\pgf@shade@adds}}%
327     \expandafter\expandafter\expandafter\def
328     \expandafter\expandafter\expandafter\@args
329     \expandafter\expandafter\expandafter{%
330         \csname pgf@args\pgf@shadingname\endcsname}%
331     \expandafter\expandafter\expandafter\@temp
332     \expandafter\@args\@args
333     }%
334 }%
335 {}%
336 \pgf@invoke shading{%
337     \csname @pgfshading\pgf@basename\pgf@shade@adds!\endcsname}%
338 }%
339 }%
340 }

```

Conditionals for use in the *<type 4 function>* argument of `\pgfdeclarefunctionalshading` to test for the currently active xcolor colour model.

```

341 \newif\ifpgfshadingmodelrgb
342 \newif\ifpgfshadingmodelcmyk
343 \newif\ifpgfshadingmodelgray

```

Shading colour space property set up based on the currently active xcolor colour model.

```

344 \def\pgf@setup@model{%
345     \pgfshadingmodelrgbtrue
346     \pgfshadingmodelcmykfalse
347     \pgfshadingmodelgrayfalse
348     \XC@sdef\pgf@mod@test{\XC@tgt@mod{natural}}%
349     \def\pgf@shading@functional@range{0 1 0 1 0 1}%
350     \def\pgf@shading@device{/DeviceRGB}%
351     \def\pgf@shading@ps@device{setrgbcolor}%
352     \def\pgf@shading@model{rgb}%
353     \ifx\pgf@mod@test\XC@mod@natural
354         \ifpgfcmykshadingdefault
355             \def\pgf@shading@functional@range{0 1 0 1 0 1 0 1}%
356             \def\pgf@shading@device{/DeviceCMYK}%
357             \def\pgf@shading@ps@device{setcmykcolor}%
358             \def\pgf@shading@model{cmyk}%
359             \pgfshadingmodelrgbfalse
360             \pgfshadingmodelcmyktrue
361         \else
362             \def\pgf@shading@functional@range{0 1 0 1 0 1}%
363             \def\pgf@shading@device{/DeviceRGB}%
364             \def\pgf@shading@ps@device{setrgbcolor}%
365             \def\pgf@shading@model{rgb}%
366         \fi
367     \fi
368     \ifx\pgf@mod@test\XC@mod@cmyk
369         \def\pgf@shading@functional@range{0 1 0 1 0 1 0 1}%
370         \def\pgf@shading@device{/DeviceCMYK}%
371         \def\pgf@shading@ps@device{setcmykcolor}%
372         \def\pgf@shading@model{cmyk}%

```

```

373     \pgfshadingmodelrgbfalse
374     \pgfshadingmodelcmyktrue
375 \fi
376 \ifx\pgf@mod@test\XC@mod@gray
377   \def\pgf@shading@functional@range{0 1}%
378   \def\pgf@shading@device{/DeviceGray}%
379   \def\pgf@shading@ps@device{setgray}%
380   \def\pgf@shading@model{gray}%
381   \pgfshadingmodelrgbfalse
382   \pgfshadingmodelgraytrue
383 \fi
384 \edef\pgf@sys@driver@dvisvgm{pgfsys-dvisvgm.def}%
385 \ifx\pgfsysdriver\pgf@sys@driver@dvisvgm
386   \def\pgf@shading@model{rgb}%
387 \fi
388 \edef\pgf@sys@driver@texforht{pgfsys-tex4ht.def}%
389 \ifx\pgfsysdriver\pgf@sys@driver@texforht
390   \def\pgf@shading@model{rgb}%
391 \fi
392 }

```

Converters for use in the *<type 4 function>* argument of `\pgfdeclarefunctionalshading`. These macros use the same algorithms as `xcolor`.

```

393 \def\pgffuncshadingrgbtocmyk{%
394   1.0 exch sub 3 1 roll
395   1.0 exch sub 3 1 roll
396   1.0 exch sub 3 1 roll
397   3 copy
398   2 copy gt { exch } if pop
399   2 copy gt { exch } if pop
400   dup 3 1 roll sub
401   0.0 2 copy lt { exch } if pop
402   1.0 2 copy gt { exch } if pop
403   4 1 roll
404   dup 3 1 roll sub
405   0.0 2 copy lt { exch } if pop
406   1.0 2 copy gt { exch } if pop
407   4 1 roll
408   dup 3 1 roll sub
409   0.0 2 copy lt { exch } if pop
410   1.0 2 copy gt { exch } if pop
411   4 1 roll
412 }
413 \def\pgffuncshadingrgbtogray{%
414   0.11 mul exch 0.59 mul add exch 0.3 mul add
415 }
416 \def\pgffuncshadingcmyktorgb{%
417   % covert to CMY
418   dup 3 1 roll add
419   1.0 2 copy gt { exch } if pop
420   4 1 roll
421   dup 3 1 roll add
422   1.0 2 copy gt { exch } if pop
423   4 1 roll

```

```

424   add
425   1.0 2 copy gt { exch } if pop
426   3 1 roll
427   % covert to RGB
428   1.0 exch sub
429   3 1 roll
430   1.0 exch sub
431   3 1 roll
432   1.0 exch sub
433   3 1 roll
434 }
435 \def\pgffuncshadingcmyktogray{%
436   exch 0.11 mul add exch 0.59 mul add exch 0.3 mul add
437   1.0 2 copy gt { exch } if pop
438   1.0 exch sub
439 }
440 \def\pgffuncshadinggraytorgb{%
441   dup dup
442 }
443 \def\pgffuncshadinggraytocmyk{%
444   0.0 0.0 0.0
445   4 3 roll
446 }

```

Load the correct driver file.

```

447 \def\pgfutilgetcmykshadingsdriver{%
448   \expandafter\pgfutil@getcmykshadingsdriver\pgfsysdriver[%
449 }
450 \def\pgfutil@getcmykshadingsdriver pgfsys-#1[{%
451   \edef\pgfsyscmykshadingsdriver{pgfsys-cmykshadings-#1}%
452 }
453 \pgfutilgetcmykshadingsdriver
454 \pgfutil@InputIfFileExists{\pgfsyscmykshadingsdriver}{}{

```

Style options to use CMYK shadings by default or not when the selected xcolor colour model is natural.

```

455 \newif\ifpgfcmykshadingdefault
456 \DeclareOption{cmyk}{%
457   \pgfcmykshadingdefaulttrue
458 }
459 \DeclareOption{rgb}{%
460   \pgfcmykshadingdefaultfalse
461 }
462 \ExecuteOptions{cmyk}
463 \ProcessOptions\relax
464 </package>

```

5.2 Drivers

pdftex driver

```

465 <*pdftex-driver>
466 \ProvidesFile{pgfsys-cmykshadings-pdftex.def}%
467   [2018/10/24
468   CMYK and grayscale shadings support for PGF pdftex driver (DCP)]

```



```

469 \def\pgfsys@horishading#1#2#3{%
470   {%
471     \pgf@parsefunc{#3}%
472     \pgfmathparse{#2}%
473     \setbox\pgfutil@tempboxa=\hbox to\pgf@max{%
474       \vbox to\pgfmathresult pt{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
475     \pgf@process{\pgfpoint{\pgf@max}{#2}}%
476     \immediate\pdfxform resources {%
477       /Shading << /Sh << /ShadingType 2
478       /ColorSpace \pgf@shading@device\space
479       /Domain [\pgf@pdfparseddomain]
480       /Coords [\pgf@doma\space0 \pgf@domb\space0]
481       /Function \pgf@pdfparsedfunction
482       /Extend [false false] >> >>}\pgfutil@tempboxa% <<
483     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
484       \leavevmode\noexpand\pdfrefxform\the\pdflastxform}%
485     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
486       \pgf@shading@model}%
487   }%
488 }
489 \def\pgfsys@vertshading#1#2#3{%
490   {%
491     \pgf@parsefunc{#3}%
492     \pgfmathparse{#2}%
493     \setbox\pgfutil@tempboxa=\hbox to\pgfmathresult pt{%
494       \vbox to\pgf@max{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
495     \pgf@process{\pgfpoint{#2}{\pgf@max}}%
496     \immediate\pdfxform resources {%
497       /Shading << /Sh << /ShadingType 2
498       /ColorSpace \pgf@shading@device\space
499       /Domain [\pgf@pdfparseddomain]
500       /Coords [0 \pgf@doma\space0 \pgf@domb]
501       /Function \pgf@pdfparsedfunction
502       /Extend [false false] >> >>}\pgfutil@tempboxa% <<
503     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
504       \leavevmode\noexpand\pdfrefxform\the\pdflastxform}%
505     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
506       \pgf@shading@model}%
507   }%
508 }
509 \def\pgfsys@radialshading#1#2#3{%
510   {%
511     \pgf@parsefunc{#3}%
512     \setbox\pgfutil@tempboxa=\hbox to2\pgf@max{%
513       \vbox to2\pgf@max{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
514     \pgf@process{#2}%
515     \pgf@xa=\pgf@x
516     \pgf@ya=\pgf@y
517     \pgf@process{\pgfpoint{\pgf@max}{\pgf@max}}%
518     \advance\pgf@xa by \pgf@x
519     \advance\pgf@ya by \pgf@y
520     \pgf@sys@bp@correct{\pgf@x}%
521     \pgf@sys@bp@correct{\pgf@y}%
522     \pgf@sys@bp@correct{\pgf@xa}%

```

```

523 \pgf@sys@bp@correct{\pgf@ya}%
524 \immediate\pdfxform resources {%
525 /Shading << /Sh << /ShadingType 3
526 /ColorSpace \pgf@shading@device\space
527 /Domain [\pgf@pdfparseddomain]
528 /Coords [\pgf@sys@tonumber{\pgf@xa}
529 \pgf@sys@tonumber{\pgf@ya}
530 \pgf@doma\space
531 \pgf@sys@tonumber{\pgf@x}
532 \pgf@sys@tonumber{\pgf@y}
533 \pgf@domb]
534 /Function \pgf@pdfparsedfunction
535 /Extend [true false] >> >>}\pgfutil@tempboxa% <<
536 \expandafter\xdef\csname @pgfshading#1!\endcsname{%
537 \leavevmode\noexpand\pdfrefxform\the\pdflastxform}%
538 \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
539 \pgf@shading@model}%
540 }%
541 }
542 \def\pgfsys@functionalshading#1#2#3#4{%
543 {%
544 \pgf@process{#2}%
545 \pgf@xa=\pgf@x
546 \pgf@ya=\pgf@y
547 \pgf@process{#3}%
548 \pgf@xb=\pgf@x
549 \pgf@yb=\pgf@y
550 \advance\pgf@x by-\pgf@xa
551 \advance\pgf@y by-\pgf@ya
552 \setbox\pgfutil@tempboxa=\hbox to\pgf@x{%
553 \vbox to\pgf@y{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
554 \pgf@sys@bp@correct{\pgf@xa}%
555 \pgf@sys@bp@correct{\pgf@ya}%
556 \pgf@sys@bp@correct{\pgf@xb}%
557 \pgf@sys@bp@correct{\pgf@yb}%
558 \pgf@xc=-\pgf@xa
559 \pgf@yc=-\pgf@ya
560 % Now build the function
561 \pdfobj
562 stream
563 attr
564 {
565 /FunctionType 4
566 /Domain [\pgf@sys@tonumber{\pgf@xa}\space
567 \pgf@sys@tonumber{\pgf@xb}\space
568 \pgf@sys@tonumber{\pgf@ya}\space
569 \pgf@sys@tonumber{\pgf@yb}]
570 /Range [\pgf@shading@functional@range]
571 }
572 {{#4}}%
573 \edef\pgf@temp@num{\the\pdflastobj}%
574 \pdfxform resources {%
575 /Shading << /Sh << /ShadingType 1
576 /ColorSpace \pgf@shading@device\space

```

```

577 /Matrix [1 0 0 1 \pgf@sys@tonumber{\pgf@xc}\space
578 \pgf@sys@tonumber{\pgf@yc}]
579 /Domain [\pgf@sys@tonumber{\pgf@xa}\space
580 \pgf@sys@tonumber{\pgf@xb}\space
581 \pgf@sys@tonumber{\pgf@ya}\space
582 \pgf@sys@tonumber{\pgf@yb}]
583 /Function \pgf@temp@num\space 0 R
584 >> >>}\pgfutil@tempboxa% <<
585 \expandafter\xdef\csname @pgfshading#1!\endcsname{%
586 \leavevmode%
587 \noexpand\pdfrefxform\the\pdflastxform%
588 \noexpand\pdfrefobj\pgf@temp@num%
589 }%
590 \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
591 \pgf@shading@model}%
592 }%
593 }
594 </pdfTeX-driver>

```

xetex driver

```

595 <*\xetex-driver>
596 \ProvidesFile{pgfsys-cmykshadings-xetex.def}%
597 [2018/10/24
598 CMYK and grayscale shadings support for PGF xetex driver (DCP)]
599 \input pgfsys-cmykshadings-dvipdfmx.def
600 </xetex-driver>

```

luatex driver

```

601 <*\luatex-driver>
602 \ProvidesFile{pgfsys-cmykshadings-luatex.def}%
603 [2018/10/24
604 CMYK and grayscale shadings support for PGF luatex driver (DCP)]
605 \def\pgfsys@horishading#1#2#3{%
606 {%
607 \pgf@parsefunc{#3}%
608 \pgfmathparse{#2}%
609 \setbox\pgfutil@tempboxa=\hbox to\pgf@max{%
610 \vbox to\pgfmathresult pt{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
611 \pgf@process{\pgfpoint{\pgf@max}{#2}}%
612 \immediate\saveboxresource resources {%
613 /Shading << /Sh << /ShadingType 2
614 /ColorSpace \pgf@shading@device\space
615 /Domain [\pgf@pdfparseddomain]
616 /Coords [\pgf@doma\space0 \pgf@domb\space0]
617 /Function \pgf@pdfparsedfunction
618 /Extend [false] >> >>}\pgfutil@tempboxa% <<
619 \expandafter\xdef\csname @pgfshading#1!\endcsname{%
620 \leavevmode\noexpand\useboxresource\the\lastsavedboxresourceindex}%
621 \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
622 \pgf@shading@model}%
623 }%

```

```

624 }
625 \def\pgfsys@vertshading#1#2#3{%
626   {%
627     \pgf@parsefunc{#3}%
628     \pgfmathparse{#2}%
629     \setbox\pgfutil@tempboxa=\hbox to\pgfmathresult pt{%
630       \vbox to\pgf@max{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
631     \pgf@process{\pgfpoint{#2}{\pgf@max}}%
632     \immediate\saveboxresource resources {%
633       /Shading << /Sh << /ShadingType 2
634       /ColorSpace \pgf@shading@device\space
635       /Domain [\pgf@pdfparseddomain]
636       /Coords [0 \pgf@doma\space0 \pgf@domb]
637       /Function \pgf@pdfparsedfunction
638       /Extend [false false] >> >>}\pgfutil@tempboxa% <<
639     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
640       \leavevmode\noexpand\useboxresource\the\lastsavedboxresourceindex}%
641     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
642       \pgf@shading@model}%
643   }%
644 }
645 \def\pgfsys@radialshading#1#2#3{%
646   {%
647     \pgf@parsefunc{#3}%
648     \setbox\pgfutil@tempboxa=\hbox to2\pgf@max{%
649       \vbox to2\pgf@max{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
650     \pgf@process{#2}%
651     \pgf@xa=\pgf@x
652     \pgf@ya=\pgf@y
653     \pgf@process{\pgfpoint{\pgf@max}{\pgf@max}}%
654     \advance\pgf@xa by \pgf@x
655     \advance\pgf@ya by \pgf@y
656     \pgf@sys@bp@correct{\pgf@x}%
657     \pgf@sys@bp@correct{\pgf@y}%
658     \pgf@sys@bp@correct{\pgf@xa}%
659     \pgf@sys@bp@correct{\pgf@ya}%
660     \immediate\saveboxresource resources {%
661       /Shading << /Sh << /ShadingType 3
662       /ColorSpace \pgf@shading@device\space
663       /Domain [\pgf@pdfparseddomain]
664       /Coords [\pgf@sys@tonumber{\pgf@xa}
665         \pgf@sys@tonumber{\pgf@ya}
666         \pgf@doma\space
667         \pgf@sys@tonumber{\pgf@x}
668         \pgf@sys@tonumber{\pgf@y}
669         \pgf@domb]
670     /Function \pgf@pdfparsedfunction
671     /Extend [true false] >> >>}\pgfutil@tempboxa% <<
672     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
673       \leavevmode\noexpand\useboxresource\the\lastsavedboxresourceindex}%
674     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
675       \pgf@shading@model}%
676   }%
677 }

```

```

678 \def\pgfsys@functionals shading#1#2#3#4{%
679   {%
680     \pgf@process{#2}%
681     \pgf@xa=\pgf@x
682     \pgf@ya=\pgf@y
683     \pgf@process{#3}%
684     \pgf@xb=\pgf@x
685     \pgf@yb=\pgf@y
686     \advance\pgf@x by-\pgf@xa
687     \advance\pgf@y by-\pgf@ya
688     \setbox\pgfutil@tempboxa=\hbox to\pgf@x{%
689       \vbox to\pgf@y{\vfil\pgfsys@invoke{/Sh sh}}\hfil}%
690     \pgf@sys@bp@correct{\pgf@xa}%
691     \pgf@sys@bp@correct{\pgf@ya}%
692     \pgf@sys@bp@correct{\pgf@xb}%
693     \pgf@sys@bp@correct{\pgf@yb}%
694     \pgf@xc=-\pgf@xa
695     \pgf@yc=-\pgf@ya
696     % Now build the function
697     \pdfextension obj
698     stream
699     attr
700     {
701       /FunctionType 4
702       /Domain [ \pgf@sys@tonumber{\pgf@xa}\space
703         \pgf@sys@tonumber{\pgf@xb}\space
704         \pgf@sys@tonumber{\pgf@ya}\space
705         \pgf@sys@tonumber{\pgf@yb} ]
706       /Range [ \pgf@shading@functional@range ]
707     }
708     {{#4}}%
709     \edef\pgf@temp@num{\the\numexpr\pdffeedback lastobj\relax}%
710     \saveboxresource resources {%
711       /Shading << /Sh << /ShadingType 1
712       /ColorSpace \pgf@shading@device\space
713       /Matrix [ 1 0 0 1 \pgf@sys@tonumber{\pgf@xc}\space
714         \pgf@sys@tonumber{\pgf@yc} ]
715       /Domain [ \pgf@sys@tonumber{\pgf@xa}\space
716         \pgf@sys@tonumber{\pgf@xb}\space
717         \pgf@sys@tonumber{\pgf@ya}\space
718         \pgf@sys@tonumber{\pgf@yb} ]
719       /Function \pgf@temp@num\space 0 R
720       >> >> \pgfutil@tempboxa% <<
721     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
722       \leavevmode%
723       \noexpand\useboxresource\the\lastsavedboxresourceindex%
724       \noexpand\pdfextension refobj \pgf@temp@num%
725     }%
726     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
727       \pgf@shading@model}%
728   }%
729 }
730 </luatex-driver>

```

dvipdfmx driver

```
731 (*dvipdfmx-driver)
732 \ProvidesFile{pgfsys-cmykshadings-dvipdfmx.def}%
733 [2018/10/24
734   CMYK and grayscale shadings support for PGF dvipdfmx driver (DCP)]
735 \def\pgfsys@horishading#1#2#3{%
736   {%
737     \pgf@parsefunc{#3}%
738     \pgfmathparse{#2}%
739     \pgf@process{\pgfpoint{\pgf@max}{#2}}%
740     \edef\@tempa{\noexpand\pgfutil@insertatbegincurrentpagefrombox{%
741       \special{pdf:bxobj @pgfshade\the\pgfsys@objnum\space
742         width \the\pgf@max\space height \pgfmathresult pt}%
743       \special{pdf:put @resources
744         <<
745           /Shading << /Sh << /ShadingType 2
746           /ColorSpace \pgf@shading@device\space
747           /Domain [\pgf@pdfparseddomain]
748           /Coords [\pgf@doma\space0 \pgf@domb\space0]
749           /Function \pgf@pdfparsedfunction
750           /Extend [false false] >> >>
751         >>}%
752     \pgfsys@invoke{/Sh sh}%
753     \special{pdf:exobj}}\@tempa% <<
754     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
755       \hbox to\the\pgf@max{\vbox to\pgfmathresult pt{%
756         \vfil\special{pdf:uxobj @pgfshade\the\pgfsys@objnum}}\hfil}}%
757     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
758       \pgf@shading@model}%
759   }%
760   \global\advance\pgfsys@objnum\@ne%
761 }
762 \def\pgfsys@vertshading#1#2#3{%
763   {%
764     \pgf@parsefunc{#3}%
765     \pgfmathparse{#2}%
766     \pgf@process{\pgfpoint{\pgf@max}{#2}}%
767     \edef\@tempa{\noexpand\pgfutil@insertatbegincurrentpagefrombox{%
768       \special{pdf:bxobj @pgfshade\the\pgfsys@objnum\space
769         width \pgfmathresult pt\space height \the\pgf@max}%
770       \special{pdf:put @resources
771         <<
772           /Shading << /Sh << /ShadingType 2
773           /ColorSpace \pgf@shading@device\space
774           /Domain [\pgf@pdfparseddomain]
775           /Coords [0 \pgf@doma\space0 \pgf@domb]
776           /Function \pgf@pdfparsedfunction
777           /Extend [false false] >> >>
778         >>}%
779     \pgfsys@invoke{/Sh sh}%
780     \special{pdf:exobj}}\@tempa% <<
781     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
782       \hbox to\pgfmathresult pt{\vbox to\the\pgf@max{%
```

```

783     \vfil\special{pdf:uxobj @pgfshade\the\pgfsys@objnum}}\hfil}}%
784     \expandafter\xdef\cname @pgfshading#1@model!\endcsname{%
785     \pgf@shading@model}%
786   }%
787   \global\advance\pgfsys@objnum\@ne
788 }
789 \def\pgfsys@radialshading#1#2#3{%
790   {%
791     \pgf@parsefunc{#3}%
792     \pgf@process{#2}%
793     \pgf@xa=\pgf@x
794     \pgf@ya=\pgf@y
795     \pgf@process{\pgfpoint{\pgf@max}{\pgf@max}}%
796     \advance\pgf@xa by \pgf@x
797     \advance\pgf@ya by \pgf@y
798     \pgf@sys@bp@correct{\pgf@x}%
799     \pgf@sys@bp@correct{\pgf@y}%
800     \pgf@sys@bp@correct{\pgf@xa}%
801     \pgf@sys@bp@correct{\pgf@ya}%
802     \pgfutil@tempdima=2\pgf@max
803     \edef\@tempa{\noexpand\pgfutil@insertatbegincurrentpagefrombox{%
804       \special{pdf:bxobj @pgfshade\the\pgfsys@objnum\space
805         width \the\pgfutil@tempdima\space height \the\pgfutil@tempdima}%
806       \special{pdf:put @resources
807         <<
808         /Shading << /Sh << /ShadingType 3
809         /ColorSpace \pgf@shading@device\space
810         /Domain [\pgf@pdfparseddomain]
811         /Coords [\pgf@sys@tonumber{\pgf@xa}
812           \pgf@sys@tonumber{\pgf@ya}
813           \pgf@doma\space \pgf@sys@tonumber{\pgf@x}
814           \pgf@sys@tonumber{\pgf@y}
815           \pgf@domb]
816         /Function \pgf@pdfparsedfunction
817         /Extend [true false] >> >>
818         >>}}%
819     \pgfsys@invoke{/Sh sh}%
820     \special{pdf:exobj}}\@tempa% <<
821     \expandafter\xdef\cname @pgfshading#1!\endcsname{%
822     \hbox to\the\pgfutil@tempdima{\vbox to\the\pgfutil@tempdima{%
823       \vfil\special{pdf:uxobj @pgfshade\the\pgfsys@objnum}}\hfil}}%
824     \expandafter\xdef\cname @pgfshading#1@model!\endcsname{%
825     \pgf@shading@model}%
826   }%
827   \global\advance\pgfsys@objnum\@ne
828 }
829 \def\pgfsys@functionals shading#1#2#3#4{%
830   {%
831     \pgf@process{#2}%
832     \pgf@xa=\pgf@x
833     \pgf@ya=\pgf@y
834     \pgf@process{#3}%
835     \pgf@xb=\pgf@x
836     \pgf@yb=\pgf@y

```

```

837 \advance\pgf@x by-\pgf@xa%
838 \advance\pgf@y by-\pgf@ya%
839 \pgf@sys@bp@correct{\pgf@xa}%
840 \pgf@sys@bp@correct{\pgf@ya}%
841 \pgf@sys@bp@correct{\pgf@xb}%
842 \pgf@sys@bp@correct{\pgf@yb}%
843 \pgf@xc=-\pgf@xa
844 \pgf@yc=-\pgf@ya
845 % Now build the function
846 \edef\@tempa{\noexpand\pgfutil@insertatbegincurrentpagefrombox{%
847 \special{pdf:stream @pgfstream\the\pgf@sys@objnum\space({#4})
848 <</FunctionType 4 /Domain [\pgf@sys@tonumber{\pgf@xa}\space
849 \pgf@sys@tonumber{\pgf@xb}\space
850 \pgf@sys@tonumber{\pgf@ya}\space
851 \pgf@sys@tonumber{\pgf@yb}]
852 /Range [\pgf@shading@functional@range]>>}}}\@tempa%
853 \edef\@tempa{\noexpand\pgfutil@insertatbegincurrentpagefrombox{%
854 \special{pdf:bxobj @pgfshade\the\pgf@sys@objnum\space
855 width \the\pgf@x\space height \the\pgf@y}%
856 \special{pdf:put @resources <</Shading <</Sh <</ShadingType 1
857 /ColorSpace \pgf@shading@device\space
858 /Matrix [1 0 0 1 \pgf@sys@tonumber{\pgf@xc}\space
859 \pgf@sys@tonumber{\pgf@yc}]
860 /Domain [\pgf@sys@tonumber{\pgf@xa}\space
861 \pgf@sys@tonumber{\pgf@xb}\space
862 \pgf@sys@tonumber{\pgf@ya}\space
863 \pgf@sys@tonumber{\pgf@yb}]
864 /Function @pgfstream\the\pgf@sys@objnum>> >> >>}}%
865 \pgf@sys@invoke{/Sh sh}%
866 \special{pdf:exobj}}}\@tempa% <<
867 \expandafter\xdef\csname @pgfshading#1!\endcsname{%
868 \leavevmode\hbox to\the\pgf@x{\vbox to\the\pgf@y{%
869 \vfil\special{pdf:uxobj @pgfshade\the\pgf@sys@objnum}}\hfil}}}%
870 \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
871 \pgf@shading@model}%
872 }%
873 \global\advance\pgf@sys@objnum\@ne
874 }
875 </dvipdfm-driver>

```

dvipdfm driver

```

876 <*dvipdfm-driver>
877 \ProvidesFile{pgf@sys-cmykshadings-dvipdfm.def}%
878 [2018/10/24
879 CMYK and grayscale shadings support for PGF dvipdfm driver (DCP)]
880 \def\pgf@sys@horishading#1#2#3{%
881 {%
882 \pgf@parsefunc{#3}%
883 \pgf@process{\pgfpoint{\pgf@max}{#2}}%
884 \edef\@temp{\noexpand\pgfutil@insertatbegincurrentpage{%
885 \special{pdf: beginxobj @pgfshade#1 width \the\pgf@max\space
886 height \the\pgf@y}}}\@temp
887 \edef\@temp{\noexpand\pgfutil@insertatbegincurrentpage{%

```



```

888     \special{pdf: put @resources <<
889     /Shading << /Sh << /ShadingType 2
890     /ColorSpace \pgf@shading@device\space
891     /Domain [\pgf@pdfparseddomain]
892     /Coords [\pgf@doma\space0 \pgf@domb\space0]
893     /Function \pgf@pdfparsedfunction
894     /Extend [false false] >> >> >>}}\@temp% <<
895     \pgfutil@insertatbegincurrentpage{\special{pdf: content /Sh sh}}%
896     \special{pdf: endxobj}}%
897     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
898     \hbox to\the\pgf@max{\vbox to#2{%
899     \vfil\special{pdf: usexobj @pgfshade#1}}\hfil}}%
900     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
901     \pgf@shading@model}%
902     }%
903 }
904 \def\pgfsys@vertshading#1#2#3{%
905     {%
906     \pgf@parsefunc{#3}%
907     \pgf@process{\pgfpoint{\pgf@max}{#2}}%
908     \edef\@temp{\noexpand\pgfutil@insertatbegincurrentpage{%
909     \special{pdf: beginxobj @pgfshade#1 width \the\pgf@y\space
910     height \the\pgf@max\space}}}\@temp
911     \edef\@temp{\noexpand\pgfutil@insertatbegincurrentpage{%
912     \special{pdf: put @resources <<
913     /Shading << /Sh << /ShadingType 2
914     /ColorSpace \pgf@shading@device\space
915     /Domain [\pgf@pdfparseddomain]
916     /Coords [0 \pgf@doma\space0 \pgf@domb]
917     /Function \pgf@pdfparsedfunction
918     /Extend [false false] >> >> >>}}}\@temp% <<
919     \pgfutil@insertatbegincurrentpage{\special{pdf: content /Sh sh}}%
920     \special{pdf: endxobj}}%
921     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
922     \hbox to#2{\vbox to\the\pgf@max{\vfil\special{
923     pdf: usexobj @pgfshade#1}}\hfil}}%
924     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
925     \pgf@shading@model}%
926     }%
927 }
928 \def\pgfsys@radialshading#1#2#3{%
929     {%
930     \pgf@parsefunc{#3}%
931     \pgf@process{#2}%
932     \pgf@xa=\pgf@x
933     \pgf@ya=\pgf@y
934     \pgf@process{\pgfpoint{\pgf@max}{\pgf@max}}%
935     \advance\pgf@xa by \pgf@x\relax
936     \advance\pgf@ya by \pgf@y\relax
937     \pgf@sys@bp@correct{\pgf@x}%
938     \pgf@sys@bp@correct{\pgf@y}%
939     \pgf@sys@bp@correct{\pgf@xa}%
940     \pgf@sys@bp@correct{\pgf@ya}%
941     \pgfutil@tempdima=2\pgf@max\relax

```

```

942 \edef\@temp{\noexpand\pgfutil@insertatbegincurrentpage{%
943 \special{pdf: beginxobj @pgfshade#1 width
944 \the\pgfutil@tempdima\space height \the\pgfutil@tempdima}}}\@temp%
945 \edef\@temp{\noexpand\pgfutil@insertatbegincurrentpage{%
946 \special{pdf: put @resources <<
947 /Shading << /Sh << /ShadingType 3
948 /ColorSpace \pgf@shading@device\space
949 /Domain [\pgf@pdfparseddomain]
950 /Coords [\pgf@sys@tonumber{\pgf@xa} \pgf@sys@tonumber{\pgf@ya}
951 \pgf@doma\space \pgf@sys@tonumber{\pgf@x} \pgf@sys@tonumber{\pgf@y}
952 \pgf@domb]
953 /Function \pgf@pdfparsedfunction
954 /Extend [true false] >> >> >>}}}\@temp% <<
955 \pgfutil@insertatbegincurrentpage{\special{pdf: content /Sh sh}}%
956 \special{pdf: endxobj}}}%
957 \expandafter\xdef\csname @pgfshading#1!\endcsname{%
958 \hbox to\the\pgfutil@tempdima{\vbox to\the\pgfutil@tempdima{%
959 \vfil\special{pdf: usexobj @pgfshade#1}}\hfil}}}%
960 \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
961 \pgf@shading@model}%
962 }%
963 }%
964 </dvipdfm-driver>

```

dvips driver

```

965 <*dvips-driver>
966 \ProvidesFile{pgfsys-cmykshadings-dvips.def}%
967 [2018/10/24
968 CMYK and grayscale shadings support for PGF dvips driver (DCP)]
969 \input pgfsys-cmykshadings-common-postscript.def
970 </dvips-driver>

```

textures driver

```

971 <*textures-driver>
972 \ProvidesFile{pgfsys-cmykshadings-textures.def}%
973 [2018/10/24
974 CMYK and grayscale shadings support for PGF textures driver (DCP)]
975 \input pgfsys-cmykshadings-common-postscript.def
976 </textures-driver>

```

vtex driver

```

977 <*vtex-driver>
978 \ProvidesFile{pgfsys-cmykshadings-vtex.def}%
979 [2018/10/24
980 CMYK and grayscale shadings support for PGF vtex driver (DCP)]
981 \input pgfsys-cmykshadings-common-postscript.def
982 </vtex-driver>

```

PostScript® driver common code

```

983 <*common-ps-driver>
984 \ProvidesFile{pgfsys-cmykshadings-common-postscript.def}%
985 [2018/10/24
986 CMYK and grayscale shadings support for PGF PostScript driver (DCP)]
    TEX shading macros.
987 \def\pgfsys@horishading#1#2#3{%
988   {%
989     \pgf@parsefunc{#3}%
990     \pgfmathsetlength\pgf@x{#2}%
991     \pgf@xa=\pgf@x
992     \pgf@sys@bp@correct{\pgf@x}%
993     \pgf@y=\pgf@max
994     \pgf@sys@bp@correct{\pgf@y}%
995     \expandafter\xdef\csname @pgfshading#1!\endcsname{\hbox to \the\pgf@max{%
996       \noexpand\vrule width0pt height\the\pgf@xa
997       \noexpand\pgfsys@beginpurepicture
998         \noexpand\pgfsys@rect{0pt}{0pt}{\the\pgf@max}{\the\pgf@xa}%
999         \noexpand\pgfsys@clipnext
1000         \noexpand\pgfsys@discardpath
1001         \noexpand\pgfsys@invoke{\pgf@domb\space \pgf@sys@tonumber{\pgf@x}
1002           pgfH\pgf@shading@model\space \pgf@psfuncs\space pop}%
1003         \hss
1004         \noexpand\pgfsys@endpurepicture}}}%
1005     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
1006       \pgf@shading@model}%
1007   }%
1008 }
1009 \def\pgfsys@vertshading#1#2#3{%
1010   {%
1011     \pgf@parsefunc{#3}%
1012     \pgfmathsetlength\pgf@x{#2}%
1013     \pgf@xa=\pgf@x
1014     \pgf@sys@bp@correct{\pgf@x}%
1015     \pgf@y=\pgf@max
1016     \pgf@sys@bp@correct{\pgf@y}%
1017     \expandafter\xdef\csname @pgfshading#1!\endcsname{\hbox to\the\pgf@xa{%
1018       \noexpand\vrule width0pt height\the\pgf@max
1019       \noexpand\pgfsys@beginpurepicture
1020         \noexpand\pgfsys@rect{0pt}{0pt}{\the\pgf@xa}{\the\pgf@max}%
1021         \noexpand\pgfsys@clipnext
1022         \noexpand\pgfsys@discardpath
1023         \noexpand\pgfsys@invoke{\pgf@domb\space \pgf@sys@tonumber{\pgf@x}
1024           pgfV\pgf@shading@model\space \pgf@psfuncs\space pop}%
1025         \hss
1026         \noexpand\pgfsys@endpurepicture}}}%
1027     \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
1028       \pgf@shading@model}%
1029   }%
1030 }
1031 \def\pgfsys@radialshading#1#2#3{%
1032   {%
1033     \pgf@parsefunc{#3}%
1034     \pgf@process{#2}%

```

```

1035 \pgf@sys@bp@correct{\pgf@x}%
1036 \pgf@sys@bp@correct{\pgf@y}%
1037 \pgf@xa=2\pgf@max
1038 \pgf@sys@bp@correct{\pgf@max}%
1039 \advance\pgf@x by \pgf@max
1040 \advance\pgf@y by \pgf@max
1041 \expandafter\xdef\csname @pgfshading#1!\endcsname{\hbox to \the\pgf@xa{%
1042   \noexpand\vrule width0pt height\the\pgf@xa
1043   \noexpand\pgfsys@beginpurepicture
1044   \noexpand\pgfsys@invoke{%
1045     \pgf@domb\space \pgf@sys@tonumber{\pgf@y}
1046     \pgf@sys@tonumber{\pgf@x}
1047     \pgf@sys@tonumber{\pgf@max} pgfR1\pgf@shading@model\space
1048     \pgf@psfuncs\space \pgf@firstcolor\space \pgf@doma\space
1049     pgfR2\pgf@shading@model}%
1050     \hss
1051     \noexpand\pgfsys@endpurepicture}}%
1052   \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
1053     \pgf@shading@model}%
1054   }%
1055 }
1056 \def\pgfsys@functionalshading#1#2#3#4{%
1057   {%
1058     \pgf@process{#2}%
1059     \pgf@xa=\pgf@x
1060     \pgf@ya=\pgf@y
1061     \pgf@process{#3}%
1062     \pgf@xb=\pgf@x
1063     \pgf@yb=\pgf@y
1064     \advance\pgf@x by-\pgf@xa
1065     \advance\pgf@y by-\pgf@ya
1066     \pgf@sys@bp@correct{\pgf@xa}%
1067     \pgf@sys@bp@correct{\pgf@ya}%
1068     \pgf@sys@bp@correct{\pgf@xb}%
1069     \pgf@sys@bp@correct{\pgf@yb}%
1070     \pgf@xc=-\pgf@xa
1071     \pgf@yc=-\pgf@ya
1072     \expandafter\xdef\csname @pgfshading#1!\endcsname{%
1073       \hbox to\the\pgf@x{\vbox to\the\pgf@y{\vfil
1074         \noexpand\pgfsys@beginpurepicture
1075         \noexpand\pgfsys@invoke{%
1076           \pgf@sys@tonumber{\pgf@xc} \pgf@sys@tonumber{\pgf@yc} translate
1077           1.1 setlinewidth [] 0 setdash 0 setlinecap
1078           /pgfproc {#4} bind def
1079           \pgf@sys@tonumber{\pgf@ya} 1 \pgf@sys@tonumber{\pgf@yb}
1080           {
1081             \pgf@sys@tonumber{\pgf@xa} 1 \pgf@sys@tonumber{\pgf@xb}
1082             { 1 index 2 copy pgfproc \pgf@shading@ps@device\space
1083               moveto 1.1 0 rlineto stroke }
1084             for
1085             pop
1086           }
1087           for
1088           }%

```

```

1089         \noexpand\pgfsys@endpurepicture
1090     }\hfil}%
1091 }%
1092 \expandafter\xdef\csname @pgfshading#1@model!\endcsname{%
1093     \pgf@shading@model}%
1094 }%
1095 }

PostScript® support code.
1096 \g@addto@macro\pgfsys@atbegindocument{%

Define RGB PostScript® shading functions.
1097 \pgf@sys@postscript@header{/pgfHrgb { pgfH } bind def}%
1098 \pgf@sys@postscript@header{/pgfVrgb { pgfV } bind def}%
1099 \pgf@sys@postscript@header{/pgfR1rgb { pgfR1 } bind def}%
1100 \pgf@sys@postscript@header{/pgfR2rgb { pgfR2 } bind def}%

Define CMYK PostScript® shading functions.
1101 \pgf@sys@postscript@header{/pgfHcmyk{
1102     /pgfheight exch def 0.75 setlinewidth [] 0 setdash
1103     /pgfshade {pgfAcmyk} def /pgfdir { dup 0 moveto
1104         dup 6 index lineto } bind def} bind def}%
1105 \pgf@sys@postscript@header{/pgfVcmyk{
1106     /pgfheight exch def 0.75 setlinewidth [] 0 setdash
1107     /pgfshade {pgfAcmyk} def /pgfdir { dup 0 exch moveto dup 6 index
1108         exch lineto } bind def} bind def}%
1109 \pgf@sys@postscript@header{/pgfAcmyk{
1110     /pgfdiff 10 index round cvi 10 index round cvi sub 2 mul 1 add def
1111     3 index 8 index sub pgfdiff div % put cyan-step on stack
1112     3 index 8 index sub pgfdiff div % put magenta-step on stack
1113     3 index 8 index sub pgfdiff div % put yellow-step on stack
1114     3 index 8 index sub pgfdiff div % put black-step on stack
1115     pgfheight 12 index 12 index 12 index 18 index
1116     pgfdiff {
1117         4 index 4 index 4 index 4 index setcmykcolor % Set color
1118         pgfdir
1119         stroke
1120         5 -1 roll 9 index add % cyan += inccyan
1121         5 -1 roll 8 index add % magenta += incmagenta
1122         5 -1 roll 7 index add % yellow += incyellow
1123         5 -1 roll 6 index add % black += incblack
1124         5 -1 roll .5 sub % x += 0.5
1125     } repeat
1126     mark 19 1 roll cleartomark exch pop % leave only start x on stack
1127 }bind def }%
1128 \pgf@sys@postscript@header{/pgfR1cmyk{
1129     newpath dup dup dup 0 360 arc clip newpath
1130     dup /pgfendx exch def
1131     /pgfendy exch def
1132     0.875 setlinewidth
1133     [] 0 setdash
1134     /pgfshade {pgfRcmyk} def
1135     /pgfstartx exch def
1136     /pgfstarty exch def
1137     /pgfdiffx pgfendx pgfstartx sub def
1138     /pgfdiffy pgfendy pgfstarty sub def

```

```

1139     dup /pgfdomb exch def
1140   }bind def }%
1141 \pgf@sys@postscript@header{/pgfR2cmyk{
1142   newpath 0.5 add pgfcircx pgfcircy 3 2 roll 0 360 arc
1143   setcmykcolor fill pop}bind def }%
1144 \pgf@sys@postscript@header{/pgfRcmyk{
1145   /pgfdiff 10 index round cvi 10 index round cvi sub 4 mul 1 add def
1146   /pgfcircx pgfstartx 11 index pgfdiffx pgfdomb div mul add def
1147   /pgfcircy pgfstarty 11 index pgfdiffy pgfdomb div mul add def
1148   /pgfcircxe pgfstartx 10 index pgfdiffx pgfdomb div mul add def
1149   /pgfcircye pgfstarty 10 index pgfdiffy pgfdomb div mul add def
1150   /pgfxstep pgfcircxe pgfcircx sub pgfdiff div def
1151   /pgfystep pgfcircye pgfcircy sub pgfdiff div def
1152   3 index 8 index sub pgfdiff div % put cyan-step on stack
1153   3 index 8 index sub pgfdiff div % put magenta-step on stack
1154   3 index 8 index sub pgfdiff div % put yellow-step on stack
1155   3 index 8 index sub pgfdiff div % put black-step on stack
1156   11 index 11 index 11 index 11 index 17 index
1157   pgfdiff {
1158     4 index 4 index 4 index 4 index setcmykcolor % Set color
1159     pgfcircx pgfcircy 2 index 0 360 arc closepath
1160     stroke
1161     5 -1 roll 8 index add % cyan += inccyan
1162     5 -1 roll 7 index add % magenta += incmagenta
1163     5 -1 roll 6 index add % yellow += incyellow
1164     5 -1 roll 5 index add % black += incblack
1165     5 -1 roll .25 sub % x += 0.25
1166     /pgfcircx pgfcircx pgfxstep add def
1167     /pgfcircy pgfcircy pgfystep add def
1168   } repeat
1169   mark 18 1 roll cleartomark exch pop % leave only start x on stack
1170 }bind def}%

```

Define grayscale PostScript® shading functions.

```

1171 \pgf@sys@postscript@header{/pgfHgray{
1172   /pgfheight exch def 0.75 setlinewidth [] 0 setdash
1173   /pgfshade {pgfAgray} def /pgfdir { dup 0 moveto
1174     dup 3 index lineto } bind def} bind def}%
1175 \pgf@sys@postscript@header{/pgfVgray{
1176   /pgfheight exch def 0.75 setlinewidth [] 0 setdash
1177   /pgfshade {pgfAgray} def /pgfdir { dup 0 exch moveto dup 3 index
1178     exch lineto } bind def} bind def}%
1179 \pgf@sys@postscript@header{/pgfAgray{
1180   /pgfdiff 4 index round cvi 4 index round cvi sub 2 mul 1 add def
1181   dup 2 index sub pgfdiff div % put gray-step on stack
1182   pgfheight 3 index 6 index
1183   pgfdiff {
1184     1 index setgray % Set color
1185     pgfdir
1186     stroke
1187     exch 3 index add % gray += incgray
1188     exch .5 sub % x += 0.5
1189   } repeat
1190   mark 7 1 roll cleartomark exch pop % leave only start x on stack
1191 }bind def }%

```

```

1192 \pgf@sys@postscript@header{/pgfR1gray{
1193   newpath dup dup dup 0 360 arc clip newpath
1194   dup /pgfendx exch def
1195   /pgfendy exch def
1196   0.875 setlinewidth
1197   [] 0 setdash
1198   /pgfshade {pgfRgray} def
1199   /pgfstartx exch def
1200   /pgfstarty exch def
1201   /pgfdiffx pgfendx pgfstartx sub def
1202   /pgfdiffy pgfendy pgfstarty sub def
1203   dup /pgfdomb exch def
1204   }bind def }%
1205 \pgf@sys@postscript@header{/pgfR2gray{
1206   newpath 0.5 add pgfcircx pgfcircy 3 2 roll 0 360 arc
1207   setgray fill pop}bind def }%
1208 \pgf@sys@postscript@header{/pgfRgray{
1209   /pgfdiff 4 index round cvi 4 index round cvi sub 4 mul 1 add def
1210   /pgfcircx pgfstartx 5 index pgfdiffx pgfdomb div mul add def
1211   /pgfcircy pgfstarty 5 index pgfdiffy pgfdomb div mul add def
1212   /pgfcircxe pgfstartx 4 index pgfdiffx pgfdomb div mul add def
1213   /pgfcircye pgfstarty 4 index pgfdiffy pgfdomb div mul add def
1214   /pgfxstep pgfcircxe pgfcircx sub pgfdiff div def
1215   /pgfystep pgfcircye pgfcircy sub pgfdiff div def
1216   dup 2 index sub pgfdiff div % put gray-step on stack
1217   2 index 5 index
1218   pgfdiff {
1219     1 index setgray % Set color
1220     pgfcircx pgfcircy 2 index 0 360 arc closepath
1221     stroke
1222     exch 2 index add % gray += incgray
1223     exch .25 sub % x += 0.25
1224     /pgfcircx pgfcircx pgfxstep add def
1225     /pgfcircy pgfcircy pgfystep add def
1226   } repeat
1227   mark 6 1 roll cleartomark exch pop % leave only start x on stack
1228   }bind def}%
1229 }
1230 </common-ps-driver>

```

Change History

v1.0		Support dvisvgm driver	15	
	General: First public release	1	Support PostScript® drivers 26	
v1.1		Support tex4ht driver	15	
	General: Fix typo for rgb option . . .	16	v1.1a	
	Support dvipdfm driver	24	General: Fix missing percent sign . . .	11