

# The `bmpsize` package

Heiko Oberdiek\*

2016/05/16 v1.7

## Abstract

Package `bmpsize` analyzes bitmap images to extract size and resolution data. It adds this feature to the graphics package that now do not need separate bounding box files for bitmap images. Additionally the implementation for the inclusion of bitmap images in some drivers of package `graphicx` are rewritten to support options `viewport`, `trim` and `clip`.

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	Introduction	2
1.2	Bitmap image parsers	3
1.2.1	User interface	3
1.2.2	Hints	4
1.2.3	Test program	4
1.2.4	Interface for programmers	4
1.3	Improved bitmap inclusion	4
<b>2</b>	<b>Implementation</b>	<b>5</b>
2.1	Basic package <code>bmpsize-base</code>	5
2.2	Bitmap formats	12
2.2.1	<code>png</code>	12
2.2.2	<code>jpg</code>	13
2.2.3	<code>bmp</code>	22
2.2.4	<code>gif</code>	24
2.2.5	<code>tiff</code>	25
2.2.6	<code>pnm</code>	28
2.2.7	<code>pam</code>	32
2.2.8	<code>xpm</code>	39
2.2.9	<code>tga</code>	44
2.2.10	<code>pcx</code>	46
2.2.11	<code>msp</code>	47
2.2.12	<code>sgi</code>	48
2.3	Package <code>bmpsize</code>	49
2.4	Drivers	52
2.4.1	<code>dvips</code>	52
2.4.2	<code>dvipdfm</code> and <code>dvipdfmx</code>	54
2.5	Test program <code>bmpsize-test.tex</code>	57

---

\*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

<b>3</b>	<b>Installation</b>	<b>59</b>
3.1	Download	59
3.2	Bundle installation	60
3.3	Package installation	60
3.4	Refresh file name databases	60
3.5	Some details for the interested	60
<b>4</b>	<b>References</b>	<b>61</b>
4.1	URLs for bitmap format descriptions	61
4.1.1	JPEG	61
4.1.2	PNG	61
4.1.3	GIF	61
4.1.4	BMP	61
4.1.5	PCX	61
4.1.6	MSP	62
4.1.7	TIFF	62
4.1.8	TGA	62
4.1.9	SGI	62
4.1.10	WMF	62
4.1.11	XPM	62
<b>5</b>	<b>History</b>	<b>62</b>
[2006/08/24 v1.0]		62
[2007/02/18 v1.1]		62
[2007/04/11 v1.2]		62
[2007/05/01 v1.3]		63
[2007/11/11 v1.4]		63
[2008/08/11 v1.5]		63
[2009/09/04 v1.6]		63
[2016/05/16 v1.7]		63

# 1 Documentation

## 1.1 Introduction

The support of bitmap images in the  $\text{\TeX}$  world is quite poor.  $\text{\TeX}$  can read text files and thus parse the bounding box of EPS files, but it cannot read binary files. If  $\text{\TeX}$  reads a line, it removes spaces before the line end and normalizes the line end itself to get independent from the convention of the operating system.

The situation changed with pdf $\text{\TeX}$ . It is a  $\text{\TeX}$  compiler, where the output driver is already integrated. Images of type JPEG and PNG are supported directly and the size of the images are reported back to the  $\text{\TeX}$  language. Thus it is easy for package `graphics` to get the size of the images.

The problem remains for other drivers than pdf $\text{\TeX}$  in PDF mode. The size information must either be given manually by the bounding box options or an additional file is used for each image, where the size information is stored as EPS bounding box. Program `dvips` comes with the program `ebb` that create these `.bb` files. However it ignores the natural size of the image and uses a fixed resolution of 100 DPI.

Since pdf $\text{\TeX}$  1.30.0 there are some new primitives. Especially `\pdffiledump` is very helpful. It reads a file in binary mode and reports the selected area as hex dump. It works in both DVI and PDF mode of pdf $\text{\TeX}$ . Thus it is now possible

to read and parse bitmap files to get their size. This project uses this feature to implement parsers for many bitmap file types.

## 1.2 Bitmap image parsers

This project supports the following image types:

BMP, GIF, JPEG, MSP, PAM, PCX, PNG, PNM, SGI, TGA, TIFF, WMF, XPM

Consult the documentation of your T<sub>E</sub>X distribution and driver which types are supported by your driver. Sometimes automatically triggered conversions can be configured to extend the range of supported image types.

### 1.2.1 User interface

Package `bmpsize` hooks into package `graphics`. If an image is included and its size is not given, then `bmpsize` investigates the image. If it could be parsed as known bitmap file type, the size is reported back to package `graphics`.

The following options are added to the options of package `graphicx`:

**resolutionunit:** Specifies the unit of the options for setting the resolution. Default is `1in` that means the numbers are interpreted as dots per inch (DPI).

**defaultresolution:** Bitmap files do not always provide information about their resolution (density). If this information is not given, the values of this option are used to calculate the image size. Default: `72 !`

**resolution:** This option override the resolution given in the bitmap file.

**bmpsizefast:** Values are `true` and `false`. The option is enabled by default. Then mainly  $\epsilon$ -T<sub>E</sub>X's arithmetic is used to calculate the width and height. However the dimension dimensions are limited. Therefore overflow errors can happen. Disable then this option to use the arithmetic of package `fp`. It allows a larger range of numbers at the cost of speed.

Options `defaultresolution` and `resolution` expect two numbers, separated by a space. The first is taken as density for the horizontal x axis, the second for the vertical y axis. One of the numbers may be replaced by an exclamation mark. In this an aspect ratio is respected and the correct density for this axis automatically calculated. If one number is given, this number is used for both axes. Examples:

```
defaultresolution=72 !      % Default
resolution=100             % Simulates behaviour of program ebb
```

The options can be set in `\includegraphics` or using `\bmpsizesetup`. `\setkeys{Gin}` is equivalent to the latter case.

```
\bmpsizesetup{resolutionunit=1in, resolution=100}
\includegraphics[
  defaultresolution=72!,
  bmpsizefast=false
]{image}
```

### 1.2.2 Hints

- My version of `dvips.def` 1999/02/16 v3.0i defines rules for the supported bitmap extensions, but does not include them in the list of extensions that are tried if the file name is not given with an extension. In such a case, the list of extensions can be set by `\DeclareGraphicsExtensions`, see `grfguide`. The following code just extends the list:

```
\makeatletter
\g@addto@macro\Gin@extensions{.bmp,.pcx,.msp}
\makeatother
```

- My version of `dvipdfm.def` 1998/11/24 vx.x misses the graphics rule for PNG files. It can be added by:

```
\DeclareGraphicsRule{.png}{bmp}{.bb}{#1}
```

See the previous issue to add the extension `.png` to the list of extensions for package `graphics`.

### 1.2.3 Test program

There is a test program `bmpsize-test.tex`. Run it through `latex`, `pdflatex`, or `pdfTeX`. Then given image files are inspected and the result is printed.

### 1.2.4 Interface for programmers

The macro names of the parsers are `\bmpsize@read<type>`. Example: `\bmpsize@read@jpg` in case of JPEG.

A parser sets the switch `\ifbmpsize@ok` to true, if it could successfully parse the image file. The width and height are returned in `\bmpsize@width` and `\bmpsize@height`. If information about density is available, it is used to calculate width and height of the image, otherwise the values given by option `defaultresolution` is used. `resolution` overwrites the values in the image file.

## 1.3 Improved bitmap inclusion

Some drivers for package `graphics` define the graphics type `bmp` for bitmap images. The code in the standard drivers for `dvips`, `dvipdfm`, and `dvipdfmx` is very basic and misses essential features of the package `graphicx`. Therefore the code for bitmap inclusion is automatically rewritten by this package to add the following features:

- Support for viewport and trim.
- Support for clip.
- In case of `dvipdfm` and `dvipdfmx` the bitmap images are reused and not included again if they are used more than once.

However, there is a difference between `dvipdfm` and `dvipdfmx`, especially if images are reused. In the former case the reused box has width and height of 1bp, in the latter case its natural width. Thus the correct driver option must be given. `dvipdfm` and `dvipdfmx` are not equivalent.

Older versions of `dvipdfmx` uses a size of 1in. However I do want to distinguish between versions of the same program. Therefore the support of these older versions has stopped with version 1.6 of this package. Use version `dvipdfmx-20090708` or newer (some few versions before will probably also work, but I don't want to investigate this further).

## 2 Implementation

### 2.1 Basic package **bmpsize-base**

Identification.

```
1 ⟨*base⟩
2 \ProvidesPackage{bmpsize-base}%
3 [2016/05/16 v1.7 Basic part of bmpsize (H0)]%
```

Modules of package `fp` are used for calculations.

```
4 \RequirePackage{fp-basic}
5 \RequirePackage{fp-snap}
```

Package `fp` uses nested `\loop` structures. That breaks with the plain- $\text{\TeX}$  version of `\loop`. Therefore we use the  $\text{\LaTeX}$  variant.

```
\@bmpsize@plain@loop

6 \long\def\@bmpsize@plain@loop#1\repeat{%
7   \def\iterate{%
8     #1\relax
9     \expandafter\iterate\fi
10  }%
11  \iterate
12  \let\iterate\relax
13 }

14 \RequirePackage{pdftexcmds}[2007/11/11]

15 \newif\ifbmpsize@ok
16 \let\@bmpsize@ok\bmpsize@oktrue
17
18 \newif\if@bmpsize@bigendian
19 \newif\if@bmpsize@absnum
20 \newif\if@bmpsize@user@resolution
21 \newif\if@bmpsize@fast
22 \@bmpsize@fasttrue
23
24 \def\@bmpsize@init{%
25   \let\@bmpsize@org@plain@loop\loop
26   \let\loop\@bmpsize@plain@loop
27   \bmpsize@okfalse
28   \@bmpsize@bigendiantrue
29   \@bmpsize@absnumfalse
30   \let\bmpsize@pixelwidth\relax
31   \let\bmpsize@pixelheight\relax
32   \let\bmpsize@pixelx\relax
33   \let\bmpsize@pixely\relax
34   \let\bmpsize@unit\relax
35   \let\bmpsize@pixelxdenom\relax
36   \let\bmpsize@pixelydenom\relax
37   \let\bmpsize@orientation\relax
38 }
39
40 \def\@bmpsize@stop#1\@nil{}
41
42 \def\@bmpsize@loop#1{%
43   #1%
44   \@bmpsize@loop{#1}%
45 }
46 \def\@bmpsize@break#1\@bmpsize@loop#2{}
```

```

47
48 \def\@bysize@size#1#2#3{%
49   \edef#3{\pdf@filesize{#1}}%
50   \ifx#3\@empty
51     \expandafter\@bysize@stop
52   \fi
53   \ifnum#3<#2\relax
54     \expandafter\@bysize@stop
55   \fi
56 }
57
58 \def\@bysize@read#1#2#3{%
59   \edef\@bysize@buf{\pdf@filedump{#3}{#2}{#1}}%
60   \edef\@bysize@temp{%
61     \noexpand\@bysize@check@byte{#2}\@bysize@buf{}\}\noexpand\\%
62   }%
63   \@bysize@temp
64 }
65 \def\@bysize@fillbuf#1{%
66   \ifx\@bysize@buf\@empty
67     \expandafter\@firstofone
68   \else
69     \expandafter\@gobble
70   \fi
71   {%
72     \edef\@bysize@buf{%
73       \pdf@filedump{\bysize@offset}\bysize@fillbuflength}{#1}%
74     }%
75     \ifx\@bysize@buf\@empty
76       \expandafter\@bysize@stop
77     \fi
78     \edef\bysize@offset{\the\numexpr\bysize@offset+\bysize@fillbuflength}%
79   }%
80 }
81 \def\bysize@fillbuflength{10}
82
83 \def\@bysize@append#1#2#3{%
84   \edef#1{#2#3}%
85 }
86 \def\@bysize@pushback#1{%
87   \edef\@bysize@buf{#1\@bysize@buf}%
88 }
89
90 \def\@bysize@iswhite#1{%
91   \ifnum\pdf@strcmp{#1}{09}=\z@
92   \else
93     \ifnum\pdf@strcmp{#1}{0A}=\z@
94     \else
95       \ifnum\pdf@strcmp{#1}{0D}=\z@
96       \else
97         \ifnum\pdf@strcmp{#1}{20}=\z@
98         \else
99           1%
100         \fi
101       \fi
102     \fi
103   \fi
104   \space

```

```

105 }
106 \def\@bysize@isdigit#1{%
107   \ifnum\pdf@strcmp{#1}{30}<\z@
108     1%
109   \else
110     \ifnum\pdf@strcmp{#1}{39}>\z@
111       1%
112     \fi
113   \fi
114   \space
115 }
116
117 \def\@bysize@check@byte#1#2#3{%
118   \ifnum#1<\@ne
119     \csname fi\endcsname
120     \@bysize@cleanup@end
121   \else
122     \csname fi\endcsname
123     \ifx!#2#3!%
124       \csname fi\endcsname
125       \@bysize@stop
126     \else
127       \csname fi\endcsname
128       \expandafter\@bysize@check@byte\expandafter{\the\numexpr#1-1}%
129     }
130 \def\@bysize@cleanup@end#1\{\}
131
132 \def\@bysize@swap@maybe#1{%
133   \if@bysize@bigendian
134   \else
135     \edef#1{\expandafter\@bysize@@swap#1\@empty\@empty\@empty\@empty}%
136   \fi
137 }
138 \def\@bysize@@swap#1#2#3#4#5#6#7#8{%
139   #7#8#5#6#3#4#1#2%
140 }
141
142 \def\@bysize@skip@one{%
143   \edef\@bysize@buf{\expandafter\@gobbletwo\@bysize@buf}%
144 }
145 \def\@bysize@skip@two{%
146   \edef\@bysize@buf{\expandafter\@gobblefour\@bysize@buf}%
147 }
148 \def\@bysize@skip@four{%
149   \edef\@bysize@buf{%
150     \expandafter\expandafter\expandafter\@gobblefour\expandafter
151     \@gobblefour\@bysize@buf
152   }%
153 }
154
155 \def\@bysize@grab#1#2{%
156   \edef#1{\noexpand\@bysize@grab@byte#2=\@bysize@buf\noexpand\}%
157   \edef#1{#1}%
158 }
159 \def\@bysize@grab@byte#1=#2#3{%
160   #2#3%
161   \ifnum#1>\@ne
162     \expandafter\@bysize@grab@byte\the\numexpr#1-1\expandafter=%

```

```

163 \else
164 \expandafter\@bysize@cleanup@end
165 \fi
166 }
167
168 \def\@bysize@abs@maybe#1{%
169 \let\@bysize@temp\relax
170 \if@bysize@absnum
171 \ifnum"\expandafter\@car#1\@nil>7 %
172 \edef#1{\expandafter\@bysize@abs@byte#1\relax}%
173 \ifnum\pdf@strcmp{#1}{7FFFFFFF}=\z@
174 \let\@bysize@temp\@bysize@stop
175 \else
176 \def\@bysize@temp{\edef#1{\the\numexpr#1+1}}%
177 \fi
178 \fi
179 \fi
180 }
181 \def\@bysize@abs@byte#1{%
182 \ifx#1\relax
183 \else
184 \ifcase"0#1 %
185 F\or E\or D\or C\or B\or A\or 9\or 8\or
186 7\or 6\or 5\or 4\or 3\or 2\or 1\or 0%
187 \fi
188 \expandafter\@bysize@abs@byte
189 \fi
190 }
191
192 \def\@bysize@num@one#1{%
193 \@bysize@grab#11%
194 \@bysize@abs@maybe#1%
195 \edef#1{\number"#1}%
196 \@bysize@temp
197 \@bysize@skip@one
198 }
199 \def\@bysize@num@two#1{%
200 \@bysize@grab#12%
201 \@bysize@swap@maybe#1%
202 \@bysize@abs@maybe#1%
203 \edef#1{\number"#1}%
204 \@bysize@temp
205 \@bysize@skip@two
206 }
207 \def\@bysize@num@four#1{%
208 \@bysize@grab#14%
209 \@bysize@swap@maybe#1%
210 \@bysize@abs@maybe#1%
211 \ifnum\pdf@strcmp{#1}{7FFFFFFF}>\z@
212 \expandafter\@bysize@stop
213 \fi
214 \edef#1{\number"#1}%
215 \@bysize@temp
216 \@bysize@skip@four
217 }
218
219 \def\@bysize@div#1#2#3{% #1 := #2/#3
220 \FPdiv#1{#2}{#3}%

```



```

221 \@bysize@beautify#1%
222 }
223 \def\@bysize@beautify#1{%
224   \FPifint#1%
225     \edef#1{\expandafter\@bysize@trunc#1.\@nil}%
226   \else
227     \edef#1{\expandafter\@bysize@cleanup@frac#1.\@nil}%
228   \fi
229 }
230 \def\@bysize@trunc#1.#2\@nil{#1}
231 % #1 isn't an integer, thus we should have at least one
232 % necessary digit after the dot
233 \def\@bysize@cleanup@frac#1.#2#3.#4\@nil{%
234   #1.#2%
235   \ifx\@#3\%
236   \else
237     \@bysize@cleanup@fracdigits#3000000000\@nil
238   \fi
239 }
240 \def\@bysize@cleanup@fracdigits#1#2#3#4#5#6#7#8#9{%
241   \ifcase#9 %
242     \ifcase#8 %
243       \ifcase#7 %
244         \ifcase#6 %
245           \ifcase#5 %
246             \ifcase #4 %
247               \ifcase #3 %
248                 \ifcase #2 %
249                   \ifcase #1 %
250                     \else
251                       #1%
252                     \fi
253                   \else
254                     #1#2%
255                   \fi
256                 \else
257                   #1#2#3%
258                 \fi
259               \else
260                 #1#2#3#4%
261               \fi
262             \else
263               #1#2#3#4#5%
264             \fi
265           \else
266             #1#2#3#4#5#6%
267           \fi
268         \else
269           #1#2#3#4#5#6#7%
270         \fi
271       \else
272         #1#2#3#4#5#6#7#8%
273       \fi
274     \else
275       #1#2#3#4#5#6#7#8#9%
276     \fi
277   \@bysize@trunc.%
278 }

```

```

279
280 \def\@bysize@end{%
281   \ifbysize@ok
282     \ifx\bysize@pixelwidth\relax
283       \bysize@okfalse
284     \fi
285     \ifx\bysize@pixelheight\relax
286       \bysize@okfalse
287     \fi
288   \fi
289   \ifbysize@ok
290     \ifnum\bysize@pixelwidth>\z@
291     \else
292       \bysize@okfalse
293     \fi
294     \ifnum\bysize@pixelheight>\z@
295     \else
296       \bysize@okfalse
297     \fi
298   \fi
299   \ifbysize@ok
300     \ifcase 0%
301       \ifx\bysize@pixelx\relax 1 \fi
302       \ifx\bysize@pixely\relax 1 \fi
303       \ifnum\bysize@pixelx>\z@\else 1 \fi
304       \ifnum\bysize@pixely>\z@\else 1 \fi
305       \ifx\bysize@pixelxdenom\relax
306         \ifx\bysize@pixelydenom\relax\else 1 \fi
307       \else
308         \ifnum\bysize@pixelxdenom>\z@\else 1 \fi
309       \fi
310       \ifx\bysize@pixelydenom\relax
311       \else
312         \ifnum\bysize@pixelydenom>\z@\else 1 \fi
313       \fi
314     \else
315       \let\bysize@pixelx\relax
316       \let\bysize@pixely\relax
317       \let\bysize@unit\relax
318       \let\bysize@pixelxdenom\relax
319       \let\bysize@pixelydenom\relax
320     \fi
321     \ifx\bysize@pixelxdenom\relax
322     \else
323       \@bysize@div\bysize@pixelx\bysize@pixelx\bysize@pixelxdenom
324       \@bysize@div\bysize@pixely\bysize@pixely\bysize@pixelydenom
325       \let\bysize@pixelxdenom\relax
326       \let\bysize@pixelydenom\relax
327     \fi
328     \ifcase 0\ifx\bysize@unit\relax 1\fi
329       \ifbysize@user@resolution 1\fi
330       \relax
331       \let\bysize@calc@unit\bysize@unit
332       \let\bysize@calc@pixelx\bysize@pixelx
333       \let\bysize@calc@pixely\bysize@pixely
334     \else
335       \let\bysize@calc@unit\bysize@unit@default
336       \let\bysize@calc@pixelx\bysize@pixelx@default

```

```

337 \let\bmpsize@calc@pixely\bmpsize@pixely@default
338 \ifx\bmpsize@calc@pixely\Gin@exclamation
339 \ifx\bmpsize@pixelx\relax
340 \let\bmpsize@calc@pixely\bmpsize@calc@pixelx
341 \else
342 \FPdiv\bmpsize@calc@pixely\bmpsize@calc@pixelx\bmpsize@pixelx
343 \FPmul\bmpsize@calc@pixely\bmpsize@calc@pixely\bmpsize@pixely
344 \fi
345 \else
346 \ifx\bmpsize@calc@pixelx\Gin@exclamation
347 \ifx\bmpsize@pixelx\relax
348 \let\bmpsize@calc@pixelx\bmpsize@calc@pixely
349 \else
350 \FPdiv\bmpsize@calc@pixelx\bmpsize@calc@pixely\bmpsize@pixely
351 \FPmul\bmpsize@calc@pixelx\bmpsize@calc@pixelx\bmpsize@pixelx
352 \fi
353 \fi
354 \fi
355 \fi
356 \FPdiv\bmpsize@width\bmpsize@pixelwidth\bmpsize@calc@pixelx
357 \FPdiv\bmpsize@height\bmpsize@pixelheight\bmpsize@calc@pixely
358 % calculation of width and height in bp for package graphics
359 % 1in = 72bp = 72.27pt, 72/72.27 = 8/8.03, 1pt = 65536sp
360 \if@bmpsize@fast
361 \edef\bmpsize@width{%
362 \strip@pt\dimexpr.99626\dimexpr
363 \bmpsize@width\dimexpr\bmpsize@calc@unit
364 }%
365 \edef\bmpsize@height{%
366 \strip@pt\dimexpr.99626\dimexpr
367 \bmpsize@height\dimexpr\bmpsize@calc@unit
368 }%
369 \else
370 \edef\@bmpsize@temp{\number\dimexpr\bmpsize@calc@unit}%
371 \ifnum\@bmpsize@temp>100000 %
372 \FPmul\@bmpsize@temp\@bmpsize@temp{0.00001}%
373 \def\@bmpsize@corr{100000}%
374 \else
375 \let\@bmpsize@corr\relax
376 \fi
377 \FPmul\bmpsize@width\bmpsize@width\@bmpsize@temp
378 \FPmul\bmpsize@height\bmpsize@height\@bmpsize@temp
379 \FPmul\bmpsize@width\bmpsize@width{8}%
380 \FPmul\bmpsize@height\bmpsize@height{8}%
381 \FPdiv\bmpsize@width\bmpsize@width{8.03}%
382 \FPdiv\bmpsize@height\bmpsize@height{8.03}%
383 \FPdiv\bmpsize@width\bmpsize@width{65536}%
384 \FPdiv\bmpsize@height\bmpsize@height{65536}%
385 \ifx\@bmpsize@corr\relax
386 \else
387 \FPmul\bmpsize@width\bmpsize@width\@bmpsize@corr
388 \FPmul\bmpsize@height\bmpsize@height\@bmpsize@corr
389 \fi
390 \FPround\bmpsize@width\bmpsize@width{5}%
391 \FPround\bmpsize@height\bmpsize@height{5}%
392 \@bmpsize@beautify\bmpsize@width
393 \@bmpsize@beautify\bmpsize@height
394 \fi

```

```

395 \fi
396 \let\loop\@bysize@org@plain@loop
397 }
398 \def\bysize@unit@default{72.27pt}% more accurate than lin
399 \def\bysize@pixelx@default{72}
400 \let\bysize@pixely@default\Gin@exclamation
401
402 \def\bysize@types{png,jpg,bmp,gif,tiff,pnm,pam,xpm,tga,pcx,msp,sgi}
403 \base

```

## 2.2 Bitmap formats

### 2.2.1 png

```

begin_png
big-endian

read_24_0
grab_8_0->\temp
check_streq_\temp[0x89_0x0D_0x0A_0x1A_0x0A]
num_4_0->\length
grab_4_0->\temp
check_streq_\temp["IHDR"]
num_4_0->\pixelwidth
num_4_0->\pixelheight
ok
assign_numexpr(20_+\length)_->\offset
loop
  read_8_\offset
  num_4_0->\length
  grab_4_0->\temp
  if_streq_\temp["IDAT"]
  stop
  fi
  if_streq_\temp["pHYs"]
  read_9_numexpr(\offset_+8)
  num_4_0->\pixelx
  num_4_0->\pixely
  grab_1_0->\temp
  if_numeq_\temp_1
  assign_{100cm}_->\unit
  fi
  stop
  fi
  assign_numexpr(\offset_+12_+\length)_->\offset
repeat
end

```

\bysize@read@png

```

404 (*base)
405 \def\bysize@read@png#1{%
406 \@bysize@init
407 \@bysize@bigendiantrue
408 \@bysize@read{#1}{24}{0}%
409 \@bysize@grab\bysize@temp{8}%
410 \@bysize@skip@four
411 \@bysize@skip@four
412 \ifnum\pdf@strcmp{\bysize@temp}{89504E470D0A1A0A}=\z@

```

```

413 \else
414   \expandafter\@bysize@stop
415 \fi
416 \@bysize@num@four\bysize@length
417 \@bysize@grab\bysize@temp{4}%
418 \@bysize@skip@four
419 \ifnum\pdf@strcmp{\bysize@temp}{49484452}=\z@
420 \else
421   \expandafter\@bysize@stop
422 \fi
423 \@bysize@num@four\bysize@pixelwidth
424 \@bysize@num@four\bysize@pixelheight
425 \@bysize@ok
426 \edef\bysize@offset{\the\numexpr20+\bysize@length}%
427 \@bysize@loop{%
428   \@bysize@read{#1}{8}{\bysize@offset}%
429   \@bysize@num@four\bysize@length
430   \@bysize@grab\bysize@temp{4}%
431   \@bysize@skip@four
432   \ifnum\pdf@strcmp{\bysize@temp}{49444154}=\z@
433     \expandafter\@firstofone
434   \else
435     \expandafter\@gobble
436   \fi
437   {%
438     \@bysize@stop
439   }%
440   \ifnum\pdf@strcmp{\bysize@temp}{70485973}=\z@
441     \expandafter\@firstofone
442   \else
443     \expandafter\@gobble
444   \fi
445   {%
446     \@bysize@read{#1}{9}{\numexpr\bysize@offset+8\relax}%
447     \@bysize@num@four\bysize@pixelx
448     \@bysize@num@four\bysize@pixely
449     \@bysize@grab\bysize@temp{1}%
450     \@bysize@skip@one
451     \ifnum\bysize@temp=1\relax
452       \expandafter\@firstofone
453     \else
454       \expandafter\@gobble
455     \fi
456     {%
457       \def\bysize@unit{100cm}%
458     }%
459     \@bysize@stop
460   }%
461   \edef\bysize@offset{\the\numexpr\bysize@offset+12+\bysize@length}%
462 }%
463 \@bysize@stop
464 \nil
465 \@bysize@end
466 }%
467 \</base>

```

### 2.2.2 jpg

```

begin_jpg

read_3_0
grab_3_0->$temp%_S0I_and_0xFF
check_streq_$temp_[0xFF_0xD8_0xFF]
assign_{2}_->$offset
assign_{0}_->$exifdensity
loop
  read_4_$offset
  grab_1_0->$temp
  check_streq_$temp_[0xFF]
  num_1_0->$temp
  if_numeq_$temp_0xDA_%_SOS
  stop
fi
%_look_for_JFIF_APP0_segment
if_numeq_$temp_0xE0_%_APP0
  num_2_0->$length
  if_numeq_$exifdensity_0
  if_numge_$length_16_%_a_JFIF_segment_has_16_bytes_at_least
  read_12_numexpr($offset+_4)
  grab_5_0->$temp%_identifier
  if_streq_$temp_["JFIF"_0x0]
  check_numge_$length_16
  skip_2_%_version
  num_1_0->$temp%_units
  if_numeq_$temp_1
  assign_{72.27pt}_->$unit
  else
  if_numeq_$temp_2
  assign_{1cm}_->$unit
  fi
fi
num_2_0->$pixelx
num_2_0->$pixely
fi
fi
else
  if_numeq_$temp_0xE1_%_APP1
  %_look_for_Exif_APP1_segment
  num_2_0->$length
  if_numge_$length_20_%_identifier_(6)_+_Tiff_header_(8)_+_first_IFD_(>=6)
  read_20_numexpr($offset+_4)
  grab_6_0->$temp
  if_streq_$temp_["Exif"_0x0_0x0]
  assign_numexpr($offset+_10)_->$exifoffset
  %_read_TIFF_header
  grab_2_0->$temp
  if_streq_$temp_["II"]
  little-endian
  else
  check_streq_$temp_["MM"]
  %_big-endian
  fi
  num_2_0->$temp
  check_numeq_$temp_42
  num_4_0->$temp%_offset_of_first_IFD

```

```

#####check_numgt_$temp_0
#####%_read_first_IFD
#####assign_numexpr($temp+_$_exifoffset)->_$_off
#####read_2_$_off
#####num_2->_$_entries
#####assign_numexpr($_off+_2)->_$_off
#####loop
#####if_numeq_$_entries_0
#####break
#####fi
#####assign_numexpr($_entries-_1)->_$_entries
#####%_entry_format:
#####%_2_tag
#####%_2_field_type
#####%_4_count
#####%_4_value/offset
#####read_12_$_off
#####assign_numexpr($_off+_12)->_$_off
#####num_2->_$_tag
#####if_numeq_$_tag_296_$_ResolutionUnit
#####skip_6_$_type:_3_(short),_count:_1
#####num_2->_$_temp
#####ifcase_$_temp
#####or_1
#####clear_$_unit
#####or_2
#####assign_{72.27pt}->_$_unit
#####or_3
#####assign_{1cm}->_$_unit
#####else
#####clear_$_unit_$_unknown
#####fi
#####ifcase_$_temp
#####or_1
#####or_2
#####assign_{1}->_$_exifdensity
#####or_3
#####assign_{1}->_$_exifdensity
#####else
#####assign_$_exifdensity->_$_exifdensity
#####fi
#####fi
#####%_256_ImageWidth_(use_width_of_JPG_part)
#####%_257_ImageHeight_(use_height_of_JPG_part)
#####if_numeq_$_tag_274_$_Orientation
#####skip_6_$_type:_3_(short),_count:_1
#####num_2->_$_temp
#####if_numge_$_temp_0
#####if_numle_$_temp_8
#####assign_$_temp->_$_orientation
#####fi
#####fi
#####fi
#####if_numeq_$_tag_282_$_XResolution
#####skip_6
#####num_4->_$_temp
#####read_8_numexpr($temp+_$_exifoffset)
#####num_4->_$_pixelx

```

```

num_4->$temp
if_numeq_$temp_1
else
assign_numexpr($temp)->$pixelxdenom
%div_$pixelx_$temp->$pixelx
fi
fi
if_numeq_$tag_283_%YResolution
skip_6
num_4->$temp
read_8_numexpr($temp+_$exifoffset)
num_4->$pixely
num_4->$temp
if_numeq_$temp_1
else
assign_numexpr($temp)->$pixelydenom
%div_$pixely_$temp->$pixely
fi
fi
repeat
big-endian
fi
fi
else
assign_numexpr($temp-_0xC0)->$temp
ifcase_$temp_%SOF_0
or_%SOF_1
or_%SOF_2
or_%SOF_3
or_%DHT
assign_{-1}->$temp
or_%SOF_5
or_%SOF_6
or_%SOF_7
or_%JPG
assign_{-1}->$temp
or_%SOF_9
or_%SOF_10
or_%SOF_11
or_%DAC
assign_{-1}->$temp
or_%SOF_13
or_%SOF_14
or_%SOF_15
else
assign_{-1}->$temp
fi
if_numeq_$temp_-1
else
read_4_numexpr($offset+_5)
num_2->$pixelheight
num_2->$pixelwidth
if_numeq_$pixelheight_0
clear_$pixelheight
stop
fi
ok
stop

```



```

uuuuuu fi
uuuuuu num_2_ -> _ $length
uuuu fi
uu fi
uu assign_ numexpr($offset_+_ $length_+_2)_ -> _ $offset
repeat
end

```

\bmpsize@read@jpg

```

468 (*base)
469 \def\bmpsize@read@jpg#1{%
470   \@bmpsize@init
471   \@bmpsize@read{#1}{3}{0}%
472   \@bmpsize@grab\bmpsize@temp{3}%
473   \@bmpsize@skip@two
474   \@bmpsize@skip@one
475   \ifnum\pdf@strcmp{\bmpsize@temp}{FFD8FF}=\z@
476   \else
477     \expandafter\@bmpsize@stop
478   \fi
479   \def\bmpsize@offset{2}%
480   \def\bmpsize@exifdensity{0}%
481   \@bmpsize@loop{%
482     \@bmpsize@read{#1}{4}{\bmpsize@offset}%
483     \@bmpsize@grab\bmpsize@temp{1}%
484     \@bmpsize@skip@one
485     \ifnum\pdf@strcmp{\bmpsize@temp}{FF}=\z@
486     \else
487       \expandafter\@bmpsize@stop
488     \fi
489     \@bmpsize@num@one\bmpsize@temp
490     \ifnum\bmpsize@temp=218\relax
491       \expandafter\@firstofone
492     \else
493       \expandafter\@gobble
494     \fi
495     {%
496       \@bmpsize@stop
497     }%
498     \ifnum\bmpsize@temp=224\relax
499       \expandafter\@firstoftwo
500     \else
501       \expandafter\@secondoftwo
502     \fi
503     {%
504       \@bmpsize@num@two\bmpsize@length
505       \ifnum\bmpsize@exifdensity=0\relax
506         \expandafter\@firstofone
507       \else
508         \expandafter\@gobble
509       \fi
510     {%
511       \unless\ifnum\bmpsize@length<16\relax
512         \expandafter\@firstofone
513       \else
514         \expandafter\@gobble
515       \fi
516     {%

```

```

517 \bysize@read{#1}{12}{\numexpr\bysize@offset+4\relax}%
518 \bysize@grab\bysize@temp{5}%
519 \bysize@skip@four
520 \bysize@skip@one
521 \ifnum\pdf@strcmp{\bysize@temp}{4A46494600}=\z@
522 \expandafter\@firstofone
523 \else
524 \expandafter\@gobble
525 \fi
526 {%
527 \ifnum\bysize@length<16\relax
528 \expandafter\bysize@stop
529 \fi
530 \bysize@skip@two
531 \bysize@num@one\bysize@temp
532 \ifnum\bysize@temp=1\relax
533 \expandafter\@firstoftwo
534 \else
535 \expandafter\@secondoftwo
536 \fi
537 {%
538 \def\bysize@unit{72.27pt}%
539 }{%
540 \ifnum\bysize@temp=2\relax
541 \expandafter\@firstofone
542 \else
543 \expandafter\@gobble
544 \fi
545 {%
546 \def\bysize@unit{1cm}%
547 }%
548 }%
549 \bysize@num@two\bysize@pixelx
550 \bysize@num@two\bysize@pixely
551 }%
552 }%
553 }%
554 }{%
555 \ifnum\bysize@temp=225\relax
556 \expandafter\@firstoftwo
557 \else
558 \expandafter\@secondoftwo
559 \fi
560 {%
561 \bysize@num@two\bysize@length
562 \unless\ifnum\bysize@length<20\relax
563 \expandafter\@firstofone
564 \else
565 \expandafter\@gobble
566 \fi
567 {%
568 \bysize@read{#1}{20}{\numexpr\bysize@offset+4\relax}%
569 \bysize@grab\bysize@temp{6}%
570 \bysize@skip@four
571 \bysize@skip@two
572 \ifnum\pdf@strcmp{\bysize@temp}{457869660000}=\z@
573 \expandafter\@firstofone
574 \else

```

```

575         \expandafter\@gobble
576     \fi
577     {%
578         \edef\bmpsize@exifoffset{\the\numexpr\bmpsize@offset+10}%
579         \@bmpsize@grab\bmpsize@temp{2}%
580         \@bmpsize@skip@two
581         \ifnum\pdf@strcmp{\bmpsize@temp}{4949}=\z@
582             \expandafter\@firstoftwo
583         \else
584             \expandafter\@secondoftwo
585         \fi
586     {%
587         \@bmpsize@bigendianfalse
588     }{%
589         \ifnum\pdf@strcmp{\bmpsize@temp}{4D4D}=\z@
590             \else
591                 \expandafter\@bmpsize@stop
592             \fi
593     }%
594     \@bmpsize@num@two\bmpsize@temp
595     \ifnum\bmpsize@temp=42\relax
596     \else
597         \expandafter\@bmpsize@stop
598     \fi
599     \@bmpsize@num@four\bmpsize@temp
600     \ifnum\bmpsize@temp>0\relax
601     \else
602         \expandafter\@bmpsize@stop
603     \fi
604     \edef\bmpsize@off{\the\numexpr\bmpsize@temp+\bmpsize@exifoffset}%
605     \@bmpsize@read{#1}{2}{\bmpsize@off}%
606     \@bmpsize@num@two\bmpsize@entries
607     \edef\bmpsize@off{\the\numexpr\bmpsize@off+2}%
608     \@bmpsize@loop{%
609         \ifnum\bmpsize@entries=0\relax
610             \expandafter\@firstofone
611         \else
612             \expandafter\@gobble
613         \fi
614     {%
615         \@bmpsize@break
616     }%
617     \edef\bmpsize@entries{\the\numexpr\bmpsize@entries-1}%
618     \@bmpsize@read{#1}{12}{\bmpsize@off}%
619     \edef\bmpsize@off{\the\numexpr\bmpsize@off+12}%
620     \@bmpsize@num@two\bmpsize@tag
621     \ifnum\bmpsize@tag=296\relax
622         \expandafter\@firstofone
623     \else
624         \expandafter\@gobble
625     \fi
626     {%
627         \@bmpsize@skip@four
628         \@bmpsize@skip@two
629         \@bmpsize@num@two\bmpsize@temp
630         \ifcase\bmpsize@temp\relax
631         \or
632             \let\bmpsize@unit\relax

```

```

633         \or
634         \def\bmpsize@unit{72.27pt}%
635         \or
636         \def\bmpsize@unit{1cm}%
637         \else
638         \let\bmpsize@unit\relax
639         \fi
640         \ifcase\bmpsize@temp\relax
641         \or
642         \or
643         \def\bmpsize@exifdensity{1}%
644         \or
645         \def\bmpsize@exifdensity{1}%
646         \else
647         \let\bmpsize@exifdensity\bmpsize@exifdensity
648         \fi
649     }%
650     \ifnum\bmpsize@tag=274\relax
651         \expandafter\@firstofone
652     \else
653         \expandafter\@gobble
654     \fi
655     {%
656         \@bmpsize@skip@four
657         \@bmpsize@skip@two
658         \@bmpsize@num@two\bmpsize@temp
659         \unless\ifnum\bmpsize@temp<0\relax
660             \expandafter\@firstofone
661         \else
662             \expandafter\@gobble
663         \fi
664         {%
665             \unless\ifnum\bmpsize@temp>8\relax
666                 \expandafter\@firstofone
667             \else
668                 \expandafter\@gobble
669             \fi
670             {%
671                 \let\bmpsize@orientation\bmpsize@temp
672             }%
673         }%
674     }%
675     \ifnum\bmpsize@tag=282\relax
676         \expandafter\@firstofone
677     \else
678         \expandafter\@gobble
679     \fi
680     {%
681         \@bmpsize@skip@four
682         \@bmpsize@skip@two
683         \@bmpsize@num@four\bmpsize@temp
684         \@bmpsize@read{#1}{8}{\numexpr\bmpsize@temp+\bmpsize@exifoffset\relax}%
685         \@bmpsize@num@four\bmpsize@pixelx
686         \@bmpsize@num@four\bmpsize@temp
687         \ifnum\bmpsize@temp=1\relax
688             \expandafter\@gobble
689         \else
690             \expandafter\@firstofone

```

```

691         \fi
692     {%
693         \edef\bmpsize@pixelxdenom{\the\numexpr\bmpsize@temp}%
694     }%
695 }%
696 \ifnum\bmpsize@tag=283\relax
697     \expandafter\@firstofone
698 \else
699     \expandafter\@gobble
700 \fi
701 {%
702     \@bmpsize@skip@four
703     \@bmpsize@skip@two
704     \@bmpsize@num@four\bmpsize@temp
705     \@bmpsize@read{#1}{8}{\numexpr\bmpsize@temp+\bmpsize@exifoffset\relax}%
706     \@bmpsize@num@four\bmpsize@pixely
707     \@bmpsize@num@four\bmpsize@temp
708     \ifnum\bmpsize@temp=1\relax
709         \expandafter\@gobble
710     \else
711         \expandafter\@firstofone
712     \fi
713     {%
714         \edef\bmpsize@pixelydenom{\the\numexpr\bmpsize@temp}%
715     }%
716 }%
717 }%
718 \@bmpsize@bigendiantrue
719 }%
720 }%
721 }{%
722     \edef\bmpsize@temp{\the\numexpr\bmpsize@temp-192}%
723     \ifcase\bmpsize@temp\relax
724     \or
725     \or
726     \or
727     \or
728     \def\bmpsize@temp{-1}%
729     \or
730     \or
731     \or
732     \or
733     \def\bmpsize@temp{-1}%
734     \or
735     \or
736     \or
737     \or
738     \def\bmpsize@temp{-1}%
739     \or
740     \or
741     \or
742     \else
743         \def\bmpsize@temp{-1}%
744     \fi
745     \ifnum\bmpsize@temp=-1\relax
746         \expandafter\@gobble
747     \else
748         \expandafter\@firstofone

```

```

749     \fi
750     {%
751         \@bysize@read{#1}{4}{\numexpr\bysize@offset+5\relax}%
752         \@bysize@num@two\bysize@pixelheight
753         \@bysize@num@two\bysize@pixelwidth
754         \ifnum\bysize@pixelheight=0\relax
755             \expandafter\@firstofone
756         \else
757             \expandafter\@gobble
758         \fi
759         {%
760             \let\bysize@pixelheight\relax
761             \@bysize@stop
762         }%
763         \@bysize@ok
764         \@bysize@stop
765     }%
766     \@bysize@num@two\bysize@length
767 }%
768 }%
769 \edef\bysize@offset{\the\numexpr\bysize@offset+\bysize@length+2}%
770 }%
771 \@bysize@stop
772 \@nil
773 \@bysize@end
774 }%
775 \</base>

```

### 2.2.3 bmp

```

begin_bmp
little-endian

read_26_0
grab_2_>_temp
check_streq_temp["BM"]
skip_12
%_header_size_is_4_bytes_in_V3+,_unknown_for_V1,_V2,
%_known_header_sizes_fit_in_2_bytes
num_2_>_temp
if_num_eq_temp_12_%_V1
    skip_2
    num_2_>_pixelwidth
    num_2_>_pixelheight
    %_no_resolution_entries
    ok
    stop
fi
if_num_eq_temp_64_%_V2
    skip_2
    num_2_>_pixelwidth
    num_2_>_pixelheight
    %_missing_specification_for_resolution
    ok
    stop
fi
%_V3,_V4,_V5
skip_2

```

```

num_4_>_pixelwidth
absnum_4_>_pixelheight
ok
read_8_38
num_4_>_pixelx
num_4_>_pixely
assign_{100cm}_->_unit
end

```

\bmpsize@read@bmp

```

776 (*base)
777 \def\bmpsize@read@bmp#1{%
778   \@bmpsize@init
779   \@bmpsize@bigendianfalse
780   \@bmpsize@read{#1}{26}{0}%
781   \@bmpsize@grab\bmpsize@temp{2}%
782   \@bmpsize@skip@two
783   \ifnum\pdf@strcmp{\bmpsize@temp}{424D}=\z@
784   \else
785     \expandafter\@bmpsize@stop
786   \fi
787   \@bmpsize@skip@four
788   \@bmpsize@skip@four
789   \@bmpsize@skip@four
790   \@bmpsize@num@two\bmpsize@temp
791   \ifnum\bmpsize@temp=12\relax
792     \expandafter\@firstofone
793   \else
794     \expandafter\@gobble
795   \fi
796   {%
797     \@bmpsize@skip@two
798     \@bmpsize@num@two\bmpsize@pixelwidth
799     \@bmpsize@num@two\bmpsize@pixelheight
800     \@bmpsize@ok
801     \@bmpsize@stop
802   }%
803   \ifnum\bmpsize@temp=64\relax
804     \expandafter\@firstofone
805   \else
806     \expandafter\@gobble
807   \fi
808   {%
809     \@bmpsize@skip@two
810     \@bmpsize@num@two\bmpsize@pixelwidth
811     \@bmpsize@num@two\bmpsize@pixelheight
812     \@bmpsize@ok
813     \@bmpsize@stop
814   }%
815   \@bmpsize@skip@two
816   \@bmpsize@num@four\bmpsize@pixelwidth
817   \@bmpsize@absnumtrue
818   \@bmpsize@num@four\bmpsize@pixelheight
819   \@bmpsize@absnumfalse
820   \@bmpsize@ok
821   \@bmpsize@read{#1}{8}{38}%
822   \@bmpsize@num@four\bmpsize@pixelx
823   \@bmpsize@num@four\bmpsize@pixely

```

```

824 \def\bmpsize@unit{100cm}%
825 \@bmpsize@stop
826 \@nil
827 \@bmpsize@end
828 }%
829 </base>

```

## 2.2.4 gif

```

begin_gif
little-endian

```

```

%_Header
read_13_0
grab_3_0->$temp
check_streq_$temp["GIF"]
skip_3_0%_version

%_Logical_Screen_Descriptor
num_2_0->$pixelwidth
num_2_0->$pixelheight
skip_2_0
num_1_0->$temp%_Pixel_Aspect_Ratio
if_num_eq_$temp_0
else
  _assign_numexpr($temp+_15)_->$pixelx
  _assign_{64}_->$pixely
fi
ok
end

```

\bmpsize@read@gif

```

830 (*base)
831 \def\bmpsize@read@gif#1{%
832   \@bmpsize@init
833   \@bmpsize@bigendianfalse
834   \@bmpsize@read{#1}{13}{0}%
835   \@bmpsize@grab\bmpsize@temp{3}%
836   \@bmpsize@skip@two
837   \@bmpsize@skip@one
838   \ifnum\pdf@strcmp{\bmpsize@temp}{474946}=\z@
839   \else
840     \expandafter\@bmpsize@stop
841   \fi
842   \@bmpsize@skip@two
843   \@bmpsize@skip@one
844   \@bmpsize@num@two\bmpsize@pixelwidth
845   \@bmpsize@num@two\bmpsize@pixelheight
846   \@bmpsize@skip@two
847   \@bmpsize@num@one\bmpsize@temp
848   \ifnum\bmpsize@temp=0\relax
849     \expandafter\@gobble
850   \else
851     \expandafter\@firstofone
852   \fi
853   {%
854     \edef\bmpsize@pixelx{\the\numexpr\bmpsize@temp+15}%
855     \def\bmpsize@pixely{64}%

```



```

856 }%
857 \@bmpsize@ok
858 \@bmpsize@stop
859 \@nil
860 \@bmpsize@end
861 }%
862 </base>

```

## 2.2.5 tiff

```

begin_tiff
%_defaults
assign_{72.27pt}_->_$unit

%_Image_File_Header
read_8_0
grab_2_->_$temp
if_streq_$temp["II"]
    _little-endian
else
    _check_streq_$temp["MM"]
    _big-endian
fi
num_2_->_$temp
check_numeq_$temp_42
num_4_->_$offset_/_first_IFD_(Image_File_Directory)

%_First_IFD
read_2_$_offset
assign_numexpr($_offset+_2)_->_$offset
num_2_->_$entries
ok_/_must_rely_on_checks_at_the_end
loop
    _if_numeq_$entries_0
        _stop
    _fi
    _assign_numexpr($entries-_1)_->_$entries
    _/_entry_format:
        _/_2_tag
        _/_2_field_type
        _/_4_count
        _/_4_value/offset
        _read_12_$_offset
        _assign_numexpr($_offset+_12)_->_$offset
        _num_2_->_$tag_/_tag
        _if_numeq_$temp_296_/_ResolutionUnit
            _skip_6_/_type:_3_(short),_count:_1
            _num_2_->_$temp
            _ifcase_$temp
                _or_/_1
                    _clear_$unit
                _or_/_2
                    _assign_{72.27pt}_->_$unit
                _or_/_3
                    _assign_{1cm}_->_$unit
            _else
                _clear_$unit
            _fi

```

```

\fi
\ifnumeq\tag\256\%ImageWidth
\skip6
\num4->\$pixelwidth
\fi
\ifnumeq\tag\257\%ImageLength
\skip6
\num4->\$pixelheight
\fi
\ifnumeq\tag\282\%XResolution
\skip6
\num4->\$temp
\read8\$temp
\num4->\$pixelx
\num4->\$temp
\ifnumeq\$temp1
\else
\assignnumexpr(\$temp)->\$pixelxdenom
\div\$pixelx\$temp->\$pixelx
\fi
\fi
\ifnumeq\tag\283\%YResolution
\skip6
\num4->\$temp
\read8\$temp
\num4->\$pixely
\num4->\$temp
\ifnumeq\$temp1
\else
\assignnumexpr(\$temp)->\$pixelydenom
\div\$pixely\$temp->\$pixely
\fi
\fi
repeat
end

```

\bmpsize@read@tiff

```

863 (*base)
864 \def\bmpsize@read@tiff#1{%
865   \@bmpsize@init
866   \def\bmpsize@unit{72.27pt}%
867   \@bmpsize@read{#1}{8}{0}%
868   \@bmpsize@grab\bmpsize@temp{2}%
869   \@bmpsize@skip@two
870   \ifnum\pdf@strcmp{\bmpsize@temp}{4949}=\z@
871     \expandafter\@firstoftwo
872   \else
873     \expandafter\@secondoftwo
874   \fi
875   {%
876     \@bmpsize@bigendianfalse
877   }{%
878     \ifnum\pdf@strcmp{\bmpsize@temp}{4D4D}=\z@
879     \else
880       \expandafter\@bmpsize@stop
881     \fi
882     \@bmpsize@bigendiantrue
883   }%

```

```

884 \@bysize@num@two\bysize@temp
885 \ifnum\bysize@temp=42\relax
886 \else
887   \expandafter\@bysize@stop
888 \fi
889 \@bysize@num@four\bysize@offset
890 \@bysize@read{#1}{2}{\bysize@offset}%
891 \edef\bysize@offset{\the\numexpr\bysize@offset+2}%
892 \@bysize@num@two\bysize@entries
893 \@bysize@ok
894 \@bysize@loop{%
895   \ifnum\bysize@entries=0\relax
896     \expandafter\@firstofone
897   \else
898     \expandafter\@gobble
899   \fi
900   {%
901     \@bysize@stop
902   }%
903 \edef\bysize@entries{\the\numexpr\bysize@entries-1}%
904 \@bysize@read{#1}{12}{\bysize@offset}%
905 \edef\bysize@offset{\the\numexpr\bysize@offset+12}%
906 \@bysize@num@two\bysize@tag
907 \ifnum\bysize@temp=296\relax
908   \expandafter\@firstofone
909 \else
910   \expandafter\@gobble
911 \fi
912 {%
913   \@bysize@skip@four
914   \@bysize@skip@two
915   \@bysize@num@two\bysize@temp
916   \ifcase\bysize@temp\relax
917   \or
918     \let\bysize@unit\relax
919   \or
920     \def\bysize@unit{72.27pt}%
921   \or
922     \def\bysize@unit{1cm}%
923   \else
924     \let\bysize@unit\relax
925   \fi
926 }%
927 \ifnum\bysize@tag=256\relax
928   \expandafter\@firstofone
929 \else
930   \expandafter\@gobble
931 \fi
932 {%
933   \@bysize@skip@four
934   \@bysize@skip@two
935   \@bysize@num@four\bysize@pixelwidth
936 }%
937 \ifnum\bysize@tag=257\relax
938   \expandafter\@firstofone
939 \else
940   \expandafter\@gobble
941 \fi

```

```

942   {%
943     \@bysize@skip@four
944     \@bysize@skip@two
945     \@bysize@num@four\byysize@pixelheight
946   }%
947   \ifnum\byysize@tag=282\relax
948     \expandafter\@firstofone
949   \else
950     \expandafter\@gobble
951   \fi
952   {%
953     \@bysize@skip@four
954     \@bysize@skip@two
955     \@bysize@num@four\byysize@temp
956     \@bysize@read{#1}{8}{\byysize@temp}%
957     \@bysize@num@four\byysize@pixelx
958     \@bysize@num@four\byysize@temp
959     \ifnum\byysize@temp=1\relax
960       \expandafter\@gobble
961     \else
962       \expandafter\@firstofone
963     \fi
964     {%
965       \edef\byysize@pixelxdenom{\the\numexpr\byysize@temp}%
966     }%
967   }%
968   \ifnum\byysize@tag=283\relax
969     \expandafter\@firstofone
970   \else
971     \expandafter\@gobble
972   \fi
973   {%
974     \@bysize@skip@four
975     \@bysize@skip@two
976     \@bysize@num@four\byysize@temp
977     \@bysize@read{#1}{8}{\byysize@temp}%
978     \@bysize@num@four\byysize@pixely
979     \@bysize@num@four\byysize@temp
980     \ifnum\byysize@temp=1\relax
981       \expandafter\@gobble
982     \else
983       \expandafter\@firstofone
984     \fi
985     {%
986       \edef\byysize@pixelydenom{\the\numexpr\byysize@temp}%
987     }%
988   }%
989 }%
990 \@bysize@stop
991 \@nil
992 \@bysize@end
993 }%
994 /base>

```

## 2.2.6 pnm

```

begin_pnm
assign_{0}_{-}>_{offset}

```

```

read_3_offset
assign_{3}_->_offset
grab_1->_temp
check_streq_temp["P"]
grab_1->_temp
check_strge_temp["1"]
check_strle_temp["6"]
%ensure_one_whitespace
grab_1->_temp
if_iswhite_temp
else
  stop
fi
loop
  %skip_whitespace
  fillbuf
  grab_1->_temp
  if_iswhite_temp
  else
    if_streq_temp["#"]
    %ignore_comments
    loop
      fillbuf
      grab_1->_temp
      if_streq_temp[0x0A]
      break
    else
      if_streq_temp[0x0D]
      break
    fi
  fi
  repeat
    assign_{ }->_tempnum
  loop
    fillbuf
    grab_1->_temp
    if_isdigit_temp
    append_tempnum_temp->_tempnum
  else
    if_iswhite_temp
    break
  else
    stop
  fi
  repeat
    assign_unescapehex(tempnum)->_pixelwidth
  loop
    fillbuf
    grab_1->_temp
    if_iswhite_temp
  else

```

```

        pushback_ $temp
        break
    fi
    repeat
        assign_{} _->_ $tempnum
    loop
        fillbuf
        grab_1 _->_ $temp
        if_ isdigit_ $temp
        append_ $tempnum_ $temp _->_ $tempnum
    else
        if_ iswhite_ $temp
        break
    else
        stop
    fi
    repeat
        assign_ unescapehex($tempnum) _->_ $pixelheight
    ok
end

```

\bmpsize@read@pnm

```

995 (*base)
996 \def\bmpsize@read@pnm#1{%
997   \@bmpsize@init
998   \def\bmpsize@offset{0}%
999   \@bmpsize@read{#1}{3}{\bmpsize@offset}%
1000  \def\bmpsize@offset{3}%
1001  \@bmpsize@grab\bmpsize@temp{1}%
1002  \@bmpsize@skip@one
1003  \ifnum\pdf@strcmp{\bmpsize@temp}{50}=\z@
1004  \else
1005    \expandafter\@bmpsize@stop
1006  \fi
1007  \@bmpsize@grab\bmpsize@temp{1}%
1008  \@bmpsize@skip@one
1009  \ifnum\pdf@strcmp{\bmpsize@temp}{31}<\z@
1010    \expandafter\@bmpsize@stop
1011  \fi
1012  \ifnum\pdf@strcmp{\bmpsize@temp}{36}>\z@
1013    \expandafter\@bmpsize@stop
1014  \fi
1015  \@bmpsize@grab\bmpsize@temp{1}%
1016  \@bmpsize@skip@one
1017  \ifcase 0\@bmpsize@iswhite\bmpsize@temp
1018    \expandafter\@gobble
1019  \else
1020    \expandafter\@firstofone
1021  \fi
1022  {%
1023    \@bmpsize@stop
1024  }%
1025  \@bmpsize@loop{%
1026    \@bmpsize@fillbuf{#1}%
1027    \@bmpsize@grab\bmpsize@temp{1}%
1028    \@bmpsize@skip@one
1029    \ifcase 0\@bmpsize@iswhite\bmpsize@temp

```

```

1030     \expandafter\@gobble
1031 \else
1032     \expandafter\@firstofone
1033 \fi
1034 {%
1035     \ifnum\pdf@strcmp{\bysize@temp}{23}=\z@
1036         \expandafter\@firstoftwo
1037     \else
1038         \expandafter\@secondoftwo
1039 \fi
1040 {%
1041     \@bysize@loop{%
1042         \@bysize@fillbuf{#1}%
1043         \@bysize@grab\bysize@temp{1}%
1044         \@bysize@skip@one
1045         \ifnum\pdf@strcmp{\bysize@temp}{0A}=\z@
1046             \expandafter\@firstoftwo
1047         \else
1048             \expandafter\@secondoftwo
1049         \fi
1050         {%
1051             \@bysize@break
1052         }{%
1053             \ifnum\pdf@strcmp{\bysize@temp}{0D}=\z@
1054                 \expandafter\@firstofone
1055             \else
1056                 \expandafter\@gobble
1057             \fi
1058             {%
1059                 \@bysize@break
1060             }%
1061         }%
1062     }%
1063 }{%
1064     \@bysize@pushback\bysize@temp
1065     \@bysize@break
1066 }%
1067 }%
1068 }%
1069 \def\bysize@tempnum{%
1070 \@bysize@loop{%
1071     \@bysize@fillbuf{#1}%
1072     \@bysize@grab\bysize@temp{1}%
1073     \@bysize@skip@one
1074     \ifcase 0\@bysize@isdigit\bysize@temp
1075         \expandafter\@firstoftwo
1076     \else
1077         \expandafter\@secondoftwo
1078     \fi
1079     {%
1080         \@bysize@append\bysize@tempnum\bysize@tempnum\bysize@temp
1081     }{%
1082         \ifcase 0\@bysize@iswhite\bysize@temp
1083             \expandafter\@firstoftwo
1084         \else
1085             \expandafter\@secondoftwo
1086         \fi
1087         {%

```

```

1088      \@bysize@break
1089    }{%
1090      \@bysize@stop
1091    }%
1092  }%
1093 }%
1094 \edef\bysize@pixelwidth{\pdf@unescapehex{\bysize@tempnum}}%
1095 \@bysize@loop{%
1096   \@bysize@fillbuf{#1}%
1097   \@bysize@grab\bysize@temp{1}%
1098   \@bysize@skip@one
1099   \ifcase 0\@bysize@iswhite\bysize@temp
1100     \expandafter\@gobble
1101   \else
1102     \expandafter\@firstofone
1103   \fi
1104   {%
1105     \@bysize@pushback\bysize@temp
1106     \@bysize@break
1107   }%
1108 }%
1109 \def\bysize@tempnum{%
1110 \@bysize@loop{%
1111   \@bysize@fillbuf{#1}%
1112   \@bysize@grab\bysize@temp{1}%
1113   \@bysize@skip@one
1114   \ifcase 0\@bysize@isdigit\bysize@temp
1115     \expandafter\@firstoftwo
1116   \else
1117     \expandafter\@secondoftwo
1118   \fi
1119   {%
1120     \@bysize@append\bysize@tempnum\bysize@tempnum\bysize@temp
1121   }{%
1122     \ifcase 0\@bysize@iswhite\bysize@temp
1123       \expandafter\@firstoftwo
1124     \else
1125       \expandafter\@secondoftwo
1126     \fi
1127     {%
1128       \@bysize@break
1129     }{%
1130       \@bysize@stop
1131     }%
1132   }%
1133 }%
1134 \edef\bysize@pixelheight{\pdf@unescapehex{\bysize@tempnum}}%
1135 \@bysize@ok
1136 \@bysize@stop
1137 \@nil
1138 \@bysize@end
1139 }%
1140 </base>

```

## 2.2.7 pam

```

begin_pam
read_3_0

```



```

assign_3->_offset
assign_offset->_off
grab_3->_temp
check_streq_temp["P7"0x0A]
loop
  fillbuf
  grab_1->_temp
  if_iswhite_temp
  %%ignore_whitespace
  assign_numexpr($off+1)->_off
  else
    if_streq_temp["#"]
    %%ignore_comment_line
    assign_numexpr($off+1)->_off
    loop
      fillbuf
      grab_1->_temp
      assign_numexpr($off+1)->_off
      if_streq_temp[0x0A]
      break
    fi
  repeat
  else
    read_6_$off
    assign_numexpr($off+6)->_offset
    grab_5->_head
    if_streq_head["WIDTH"]
    assign_numexpr($off+5)->_off
    %%skip_whitespace
    loop
      fillbuf
      grab_1->_temp
      if_iswhite_temp
      assign_numexpr($off+1)->_off
      else
        if_isdigit_temp
        assign_numexpr($off+1)->_off
        break
      else
        %error
        stop
      fi
    repeat
    %read_number
    assign_temp->_tempnum
    loop
      fillbuf
      grab_1->_temp
      if_isdigit_temp
      assign_numexpr($off+1)->_off
      append_tempnum_temp->_tempnum
      else
        pushback_temp
        break
      fi
    repeat
    %skip_to_end_of_line

```

```

loop
fillbuf
grab_1->$temp
assign_numexpr($off+_1)->$off
if_streq$temp[0x0A]
break
fi
repeat
assign_unescapehex($tempnum)->$pixelwidth
else
grab_1->$temp
append_$head$temp->$head
if_streq$head["ENDHDR"]
%_last_header_line
ok
stop
else
if_streq$head["HEIGHT"]
assign_numexpr($off+_6)->$off
%_skip_white_space
loop
fillbuf
grab_1->$temp
if_iswhite$temp
assign_numexpr($off+_1)->$off
else
if_isdigit$temp
assign_numexpr($off+_1)->$off
break
else
%_error
stop
fi
fi
repeat
%_read_number
assign_$temp->$tempnum
loop
fillbuf
grab_1->$temp
if_isdigit$temp
assign_numexpr($off+_1)->$off
append_$tempnum$temp->$tempnum
else
pushback_$temp
break
fi
repeat
%_skip_to_end_of_line
loop
fillbuf
grab_1->$temp
assign_numexpr($off+_1)->$off
if_streq$temp[0x0A]
break
fi
repeat
assign_unescapehex($tempnum)->$pixelheight

```



```

1179         \expandafter \@firstofone
1180     \else
1181         \expandafter \@gobble
1182     \fi
1183     {%
1184         \@bysize@break
1185     }%
1186 }%
1187 {%
1188     \@bysize@read{#1}{6}{\@bysize@off}%
1189     \edef\bysize@offset{\the\numexpr\bysize@off+6}%
1190     \@bysize@grab\bysize@head{5}%
1191     \@bysize@skip@four
1192     \@bysize@skip@one
1193     \ifnum\pdf@strcmp{\bysize@head}{5749445448}=\z@
1194         \expandafter \@firstoftwo
1195     \else
1196         \expandafter \@secondoftwo
1197     \fi
1198     {%
1199         \edef\bysize@off{\the\numexpr\bysize@off+5}%
1200         \@bysize@loop{%
1201             \@bysize@fillbuf{#1}%
1202             \@bysize@grab\bysize@temp{1}%
1203             \@bysize@skip@one
1204             \ifcase 0\@bysize@iswhite\bysize@temp
1205                 \expandafter \@firstoftwo
1206             \else
1207                 \expandafter \@secondoftwo
1208             \fi
1209             {%
1210                 \edef\bysize@off{\the\numexpr\bysize@off+1}%
1211             }{%
1212                 \ifcase 0\@bysize@isdigit\bysize@temp
1213                     \expandafter \@firstoftwo
1214                 \else
1215                     \expandafter \@secondoftwo
1216                 \fi
1217                 {%
1218                     \edef\bysize@off{\the\numexpr\bysize@off+1}%
1219                     \@bysize@break
1220                 }{%
1221                     \@bysize@stop
1222                 }%
1223             }%
1224         }%
1225         \let\bysize@tempnum\bysize@temp
1226         \@bysize@loop{%
1227             \@bysize@fillbuf{#1}%
1228             \@bysize@grab\bysize@temp{1}%
1229             \@bysize@skip@one
1230             \ifcase 0\@bysize@isdigit\bysize@temp
1231                 \expandafter \@firstoftwo
1232             \else
1233                 \expandafter \@secondoftwo
1234             \fi
1235             {%
1236                 \edef\bysize@off{\the\numexpr\bysize@off+1}%

```

```

1237         \@bysize@append\bysize@tempnum\bysize@tempnum\bysize@temp
1238     }{%
1239         \@bysize@pushback\bysize@temp
1240         \@bysize@break
1241     }%
1242 }%
1243 \@bysize@loop{%
1244     \@bysize@fillbuf{#1}%
1245     \@bysize@grab\bysize@temp{1}%
1246     \@bysize@skip@one
1247     \edef\bysize@off{\the\numexpr\bysize@off+1}%
1248     \ifnum\pdf@strcmp{\bysize@temp}{0A}=\z@
1249         \expandafter\@firstofone
1250     \else
1251         \expandafter\@gobble
1252     \fi
1253     {%
1254         \@bysize@break
1255     }%
1256 }%
1257 \edef\bysize@pixelwidth{\pdf@unescapehex{\bysize@tempnum}}%
1258 }{%
1259     \@bysize@grab\bysize@temp{1}%
1260     \@bysize@skip@one
1261     \@bysize@append\bysize@head\bysize@head\bysize@temp
1262     \ifnum\pdf@strcmp{\bysize@head}{454E44484452}=\z@
1263         \expandafter\@firstoftwo
1264     \else
1265         \expandafter\@secondoftwo
1266     \fi
1267     {%
1268         \@bysize@ok
1269         \@bysize@stop
1270     }{%
1271         \ifnum\pdf@strcmp{\bysize@head}{484549474854}=\z@
1272             \expandafter\@firstoftwo
1273         \else
1274             \expandafter\@secondoftwo
1275         \fi
1276         {%
1277             \edef\bysize@off{\the\numexpr\bysize@off+6}%
1278             \@bysize@loop{%
1279                 \@bysize@fillbuf{#1}%
1280                 \@bysize@grab\bysize@temp{1}%
1281                 \@bysize@skip@one
1282                 \ifcase 0\@bysize@iswhite\bysize@temp
1283                     \expandafter\@firstoftwo
1284                 \else
1285                     \expandafter\@secondoftwo
1286                 \fi
1287                 {%
1288                     \edef\bysize@off{\the\numexpr\bysize@off+1}%
1289                 }{%
1290                     \ifcase 0\@bysize@isdigit\bysize@temp
1291                         \expandafter\@firstoftwo
1292                     \else
1293                         \expandafter\@secondoftwo
1294                     \fi

```

```

1295         {%
1296         \edef\bmpsize@off{\the\numexpr\bmpsize@off+1}%
1297         \@bmpsize@break
1298         }{%
1299         \@bmpsize@stop
1300         }%
1301     }%
1302 }%
1303 \let\bmpsize@tempnum\bmpsize@temp
1304 \@bmpsize@loop{%
1305     \@bmpsize@fillbuf{#1}%
1306     \@bmpsize@grab\bmpsize@temp{1}%
1307     \@bmpsize@skip@one
1308     \ifcase 0\@bmpsize@isdigit\bmpsize@temp
1309         \expandafter\@firstoftwo
1310     \else
1311         \expandafter\@secondoftwo
1312     \fi
1313     {%
1314         \edef\bmpsize@off{\the\numexpr\bmpsize@off+1}%
1315         \@bmpsize@append\bmpsize@tempnum\bmpsize@tempnum\bmpsize@temp
1316     }{%
1317         \@bmpsize@pushback\bmpsize@temp
1318         \@bmpsize@break
1319     }%
1320 }%
1321 \@bmpsize@loop{%
1322     \@bmpsize@fillbuf{#1}%
1323     \@bmpsize@grab\bmpsize@temp{1}%
1324     \@bmpsize@skip@one
1325     \edef\bmpsize@off{\the\numexpr\bmpsize@off+1}%
1326     \ifnum\pdf@strcmp{\bmpsize@temp}{0A}=\z@
1327         \expandafter\@firstofone
1328     \else
1329         \expandafter\@gobble
1330     \fi
1331     {%
1332         \@bmpsize@break
1333     }%
1334 }%
1335 \edef\bmpsize@pixelheight{\pdf@unescapehex{\bmpsize@tempnum}}%
1336 }{%
1337     \@bmpsize@pushback\bmpsize@head
1338     \@bmpsize@loop{%
1339         \@bmpsize@fillbuf{#1}%
1340         \@bmpsize@grab\bmpsize@temp{1}%
1341         \@bmpsize@skip@one
1342         \edef\bmpsize@off{\the\numexpr\bmpsize@off+1}%
1343         \ifnum\pdf@strcmp{\bmpsize@temp}{0A}=\z@
1344             \expandafter\@firstofone
1345         \else
1346             \expandafter\@gobble
1347         \fi
1348         {%
1349             \@bmpsize@break
1350         }%
1351     }%
1352 }%

```

```

1353         }%
1354     }%
1355 }%
1356 }%
1357 }%
1358 \@bmpsize@stop
1359 \@nil
1360 \@bmpsize@end
1361 }%
1362 </base>

```

## 2.2.8 xpm

```

begin_xpm
read_9_0
grab_9->$temp
assign_{9}_->$offset
check_streq_$temp["/*XPM*/"]
loop
  fillbuf
  grab_1->$temp
  if_streq_$temp[0x22]_%_
  break
fi
if_streq_$temp["/"]
  fillbuf
  grab_1->$temp
  if_streq_$temp["*"]
    %_look_for_end_of_C_comment
    loop
      fillbuf
      grab_1->$temp
      if_streq_$temp["*"]
        loop
          fillbuf
          grab_1->$temp
          if_streq_$temp["/"]
            break
          fi
          if_streq_$temp["*"]
            else
              break
            fi
          repeat
            if_streq_$temp["/"]
              break
            fi
          fi
        repeat
      fi
    fi
  repeat
    %_width
    assign_{}_->$tempnum
    loop
      fillbuf
      grab_1->$temp
      if_iswhite_$temp

```

```

    else
        if isdigit temp
            append tempnum temp-> tempnum
            break
        else
            stop
    fi
    fi
    repeat
    loop
    fillbuf
    grab 1-> temp
    if isdigit temp
        append tempnum temp-> tempnum
    else
        if iswhite temp
            break
        else
            stop
    fi
    fi
    repeat
    assign unescapehex(tempnum)-> pixelwidth
    % height
    assign {}-> tempnum
    loop
    fillbuf
    grab 1-> temp
    if iswhite temp
    else
        if isdigit temp
            append tempnum temp-> tempnum
            break
        else
            stop
    fi
    fi
    repeat
    loop
    fillbuf
    grab 1-> temp
    if isdigit temp
        append tempnum temp-> tempnum
    else
        if iswhite temp
            break
        else
            stop
    fi
    fi
    repeat
    assign unescapehex(tempnum)-> pixelheight
    ok
end

```

\bmpsize@read@xpm

```

1363 (*base)
1364 \def\bmpsize@read@xpm#1{%

```



```

1365 \@bysize@init
1366 \@bysize@read{#1}{9}{0}%
1367 \@bysize@grab\bysize@temp{9}%
1368 \@bysize@skip@four
1369 \@bysize@skip@four
1370 \@bysize@skip@one
1371 \def\bysize@offset{9}%
1372 \ifnum\pdf@strcmp{\bysize@temp}{2F2A2058504D202A2F}=\z@
1373 \else
1374   \expandafter\bysize@stop
1375 \fi
1376 \@bysize@loop{%
1377   \@bysize@fillbuf{#1}%
1378   \@bysize@grab\bysize@temp{1}%
1379   \@bysize@skip@one
1380   \ifnum\pdf@strcmp{\bysize@temp}{22}=\z@
1381     \expandafter\@firstofone
1382   \else
1383     \expandafter\@gobble
1384   \fi
1385   {%
1386     \@bysize@break
1387   }%
1388   \ifnum\pdf@strcmp{\bysize@temp}{2F}=\z@
1389     \expandafter\@firstofone
1390   \else
1391     \expandafter\@gobble
1392   \fi
1393   {%
1394     \@bysize@fillbuf{#1}%
1395     \@bysize@grab\bysize@temp{1}%
1396     \@bysize@skip@one
1397     \ifnum\pdf@strcmp{\bysize@temp}{2A}=\z@
1398       \expandafter\@firstofone
1399     \else
1400       \expandafter\@gobble
1401     \fi
1402     {%
1403       \@bysize@loop{%
1404         \@bysize@fillbuf{#1}%
1405         \@bysize@grab\bysize@temp{1}%
1406         \@bysize@skip@one
1407         \ifnum\pdf@strcmp{\bysize@temp}{2A}=\z@
1408           \expandafter\@firstofone
1409         \else
1410           \expandafter\@gobble
1411         \fi
1412         {%
1413           \@bysize@loop{%
1414             \@bysize@fillbuf{#1}%
1415             \@bysize@grab\bysize@temp{1}%
1416             \@bysize@skip@one
1417             \ifnum\pdf@strcmp{\bysize@temp}{2F}=\z@
1418               \expandafter\@firstofone
1419             \else
1420               \expandafter\@gobble
1421             \fi
1422             {%

```

```

1423         \@bysize@break
1424     }%
1425     \ifnum\pdf@strcmp{\bysize@temp}{2A}=\z@
1426         \expandafter\@gobble
1427     \else
1428         \expandafter\@firstofone
1429     \fi
1430     {%
1431         \@bysize@break
1432     }%
1433 }%
1434 \ifnum\pdf@strcmp{\bysize@temp}{2F}=\z@
1435     \expandafter\@firstofone
1436 \else
1437     \expandafter\@gobble
1438 \fi
1439 {%
1440     \@bysize@break
1441 }%
1442 }%
1443 }%
1444 }%
1445 }%
1446 }%
1447 \def\bysize@tempnum{%
1448 \@bysize@loop{%
1449     \@bysize@fillbuf{#1}%
1450     \@bysize@grab\bysize@temp{1}%
1451     \@bysize@skip@one
1452     \ifcase 0\@bysize@iswhite\bysize@temp
1453         \expandafter\@gobble
1454     \else
1455         \expandafter\@firstofone
1456     \fi
1457     {%
1458         \ifcase 0\@bysize@isdigit\bysize@temp
1459             \expandafter\@firstoftwo
1460         \else
1461             \expandafter\@secondoftwo
1462         \fi
1463         {%
1464             \@bysize@append\bysize@tempnum\bysize@tempnum\bysize@temp
1465             \@bysize@break
1466         }{%
1467             \@bysize@stop
1468         }%
1469     }%
1470 }%
1471 \@bysize@loop{%
1472     \@bysize@fillbuf{#1}%
1473     \@bysize@grab\bysize@temp{1}%
1474     \@bysize@skip@one
1475     \ifcase 0\@bysize@isdigit\bysize@temp
1476         \expandafter\@firstoftwo
1477     \else
1478         \expandafter\@secondoftwo
1479     \fi
1480     {%

```

```

1481     \@bysize@append\bysize@tempnum\bysize@tempnum\bysize@temp
1482 }{%
1483     \ifcase 0\@bysize@iswhite\bysize@temp
1484         \expandafter\@firstoftwo
1485     \else
1486         \expandafter\@secondoftwo
1487     \fi
1488     {%
1489         \@bysize@break
1490     }{%
1491         \@bysize@stop
1492     }%
1493 }%
1494 }%
1495 \edef\bysize@pixelwidth{\pdf@unescapehex{\bysize@tempnum}}%
1496 \def\bysize@tempnum{}%
1497 \@bysize@loop{%
1498     \@bysize@fillbuf{#1}%
1499     \@bysize@grab\bysize@temp{1}%
1500     \@bysize@skip@one
1501     \ifcase 0\@bysize@iswhite\bysize@temp
1502         \expandafter\@gobble
1503     \else
1504         \expandafter\@firstofone
1505     \fi
1506     {%
1507         \ifcase 0\@bysize@isdigit\bysize@temp
1508             \expandafter\@firstoftwo
1509         \else
1510             \expandafter\@secondoftwo
1511         \fi
1512         {%
1513             \@bysize@append\bysize@tempnum\bysize@tempnum\bysize@temp
1514             \@bysize@break
1515         }{%
1516             \@bysize@stop
1517         }%
1518     }%
1519 }%
1520 \@bysize@loop{%
1521     \@bysize@fillbuf{#1}%
1522     \@bysize@grab\bysize@temp{1}%
1523     \@bysize@skip@one
1524     \ifcase 0\@bysize@isdigit\bysize@temp
1525         \expandafter\@firstoftwo
1526     \else
1527         \expandafter\@secondoftwo
1528     \fi
1529     {%
1530         \@bysize@append\bysize@tempnum\bysize@tempnum\bysize@temp
1531     }{%
1532         \ifcase 0\@bysize@iswhite\bysize@temp
1533             \expandafter\@firstoftwo
1534         \else
1535             \expandafter\@secondoftwo
1536         \fi
1537         {%
1538             \@bysize@break

```

```

1539     }{%
1540         \@bysize@stop
1541     }%
1542 }%
1543 }%
1544 \edef\bysize@pixelheight{\pdf@unescapehex{\bysize@tempnum}}%
1545 \@bysize@ok
1546 \@bysize@stop
1547 \@nil
1548 \@bysize@end
1549 }%
1550 />base>

```

## 2.2.9 tga

```

begin_tga
little-endian
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%id_length(1byte)
read_16_1
grab_1->_temp%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%color_map_type(1byte),_values:_0,_1
if_streq_temp[0x00]
else
    if_streq_temp[0x01]
    else
        stop
    fi
fi
skip_10%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%image_type(1byte)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%color_map_specification(5bytes)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%x_origin(2bytes)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%y_origin(2bytes)
num_2->_pixelwidth%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%image_width
num_2->_pixelheight%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%image_height
ok
%_TGA_File_Footer
size_26->_temp
read_26_numexpr(temp-_26)
num_4->_offset%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%the_extension_area_offset
skip_4%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%the_developer_directory_offset
grab_18->_temp%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%the_signature,_"",_0x00
if_streq_temp["TRUEVISION-XFILE."_0x00]
else
    stop
fi
if_numeq_offset_0
stop%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%no_extension_area
fi
read_4_numexpr(offset+_474)_pixel_aspect_ratio(4bytes)
num_2->_pixelx%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%pixel_ratio_numerator(pixel_width)
num_2->_pixely%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%pixel_ratio_denominator(pixel_height)
if_numeq_pixely_0%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%no_pixel_aspect_ratio
clear_pixelx
clear_pixely
fi
end

```

\bysize@read@tga

```
1551 (*base>
```

```

1552 \def\bmpsize@read@tga#1{%
1553   \@bmpsize@init
1554   \@bmpsize@bigendianfalse
1555   \@bmpsize@read{#1}{16}{1}%
1556   \@bmpsize@grab\bmpsize@temp{1}%
1557   \@bmpsize@skip@one
1558   \ifnum\pdf@strcmp{\bmpsize@temp}{00}=\z@
1559     \expandafter\@gobble
1560   \else
1561     \expandafter\@firstofone
1562   \fi
1563   {%
1564     \ifnum\pdf@strcmp{\bmpsize@temp}{01}=\z@
1565       \expandafter\@gobble
1566     \else
1567       \expandafter\@firstofone
1568     \fi
1569     {%
1570       \@bmpsize@stop
1571     }%
1572   }%
1573   \@bmpsize@skip@four
1574   \@bmpsize@skip@four
1575   \@bmpsize@skip@two
1576   \@bmpsize@num@two\bmpsize@pixelwidth
1577   \@bmpsize@num@two\bmpsize@pixelheight
1578   \@bmpsize@ok
1579   \@bmpsize@size{#1}{26}\bmpsize@temp \@bmpsize@read{#1}{26}{\numexpr\bmpsize@temp-
26\relax}%
1580   \@bmpsize@num@four\bmpsize@offset
1581   \@bmpsize@skip@four
1582   \@bmpsize@grab\bmpsize@temp{18}%
1583   \@bmpsize@skip@four
1584   \@bmpsize@skip@four
1585   \@bmpsize@skip@four
1586   \@bmpsize@skip@four
1587   \@bmpsize@skip@two
1588   \ifnum\pdf@strcmp{\bmpsize@temp}{54525545564953494F4E2D5846494C452E00}=\z@
1589     \expandafter\@gobble
1590   \else
1591     \expandafter\@firstofone
1592   \fi
1593   {%
1594     \@bmpsize@stop
1595   }%
1596   \ifnum\bmpsize@offset=0\relax
1597     \expandafter\@firstofone
1598   \else
1599     \expandafter\@gobble
1600   \fi
1601   {%
1602     \@bmpsize@stop
1603   }%
1604   \@bmpsize@read{#1}{4}{\numexpr\bmpsize@offset+474\relax}%
1605   \@bmpsize@num@two\bmpsize@pixelx
1606   \@bmpsize@num@two\bmpsize@pixely
1607   \ifnum\bmpsize@pixely=0\relax
1608     \expandafter\@firstofone

```

```

1609 \else
1610 \expandafter\@gobble
1611 \fi
1612 {%
1613 \let\bmpsize@pixelx\relax
1614 \let\bmpsize@pixely\relax
1615 }%
1616 \@bmpsize@stop
1617 \@nil
1618 \@bmpsize@end
1619 }%
1620 </base>

```

## 2.2.10 pcx

```

begin_pcx
little-endian
read_16_0
grab_1_>_temp%manufacturer
check_streq_temp[0x0A]
skip_1%version
num_1_>_temp%encoding
check_num_eq_temp_1
skip_1%bits_per_pixel
num_2_>_temp%pixelwidth%_x_min
num_2_>_temp%pixelheight%_y_min
num_2_>_temp%_x_max
assign_numexpr(temp-_temp%pixelwidth+_1)_>_temp%pixelwidth
num_2_>_temp%_y_max
assign_numexpr(temp-_temp%pixelheight+_1)_>_temp%pixelheight
check_num_gt_temp%pixelwidth_0
check_num_gt_temp%pixelheight_0
ok
num_2_>_temp%horizontal_resolution_in_DPI
num_2_>_temp%vertical_resolution_in_DPI
assign_{72.27pt}_>_temp%unit
end

```

\bmpsize@read@pcx

```

1621 <(*base>
1622 \def\bmpsize@read@pcx#1{%
1623 \@bmpsize@init
1624 \@bmpsize@bigendianfalse
1625 \@bmpsize@read{#1}{16}{0}%
1626 \@bmpsize@grab\bmpsize@temp{1}%
1627 \@bmpsize@skip@one
1628 \ifnum\pdfstrcmp{\bmpsize@temp}{0A}=\z@
1629 \else
1630 \expandafter\@bmpsize@stop
1631 \fi
1632 \@bmpsize@skip@one
1633 \@bmpsize@num@one\bmpsize@temp
1634 \ifnum\bmpsize@temp=1\relax
1635 \else
1636 \expandafter\@bmpsize@stop
1637 \fi
1638 \@bmpsize@skip@one
1639 \@bmpsize@num@two\bmpsize@pixelwidth

```

```

1640 \@bysize@num@two\bysize@pixelheight
1641 \@bysize@num@two\bysize@temp
1642 \edef\bysize@pixelwidth{\the\numexpr\bysize@temp-\bysize@pixelwidth+1}%
1643 \@bysize@num@two\bysize@temp
1644 \edef\bysize@pixelheight{\the\numexpr\bysize@temp-\bysize@pixelheight+1}%
1645 \ifnum\bysize@pixelwidth>0\relax
1646 \else
1647 \expandafter\@bysize@stop
1648 \fi
1649 \ifnum\bysize@pixelheight>0\relax
1650 \else
1651 \expandafter\@bysize@stop
1652 \fi
1653 \@bysize@ok
1654 \@bysize@num@two\bysize@pixelx
1655 \@bysize@num@two\bysize@pixely
1656 \def\bysize@unit{72.27pt}%
1657 \@bysize@stop
1658 \@nil
1659 \@bysize@end
1660 }%
1661 \base

```

### 2.2.11 msp

```

begin_msp
little-endian

read_16_0

%_header_4
grab_4->_temp
if_streq_temp["DanM"]
else
  _check_streq_temp["LinS"]
fi
num_2->_pixelwidth
num_2->_pixelheight
ok
num_2->_pixelx_%x_asp
num_2->_pixely_%y_asp
assign_{72.27pt}_->_unit_%guessing
if_num_eq_pixelx_0
  _num_2->_pixelx_%x_asp_prn
  _num_2->_pixely_%y_asp_prn
fi
%_num_2_%width_prn
%_num_2_%height_prn
end

```

\bysize@read@msp

```

1662 (*base)
1663 \def\bysize@read@msp#1{%
1664 \@bysize@init
1665 \@bysize@bigendianfalse
1666 \@bysize@read{#1}{16}{0}%
1667 \@bysize@grab\bysize@temp{4}%
1668 \@bysize@skip@four

```

```

1669 \ifnum\pdf@strcmp{\bysize@temp}{44616E4D}=\z@
1670 \expandafter\@gobble
1671 \else
1672 \expandafter\@firstofone
1673 \fi
1674 {%
1675 \ifnum\pdf@strcmp{\bysize@temp}{4C696E53}=\z@
1676 \else
1677 \expandafter\@bysize@stop
1678 \fi
1679 }%
1680 \@bysize@num@two\bysize@pixelwidth
1681 \@bysize@num@two\bysize@pixelheight
1682 \@bysize@ok
1683 \@bysize@num@two\bysize@pixelx
1684 \@bysize@num@two\bysize@pixely
1685 \def\bysize@unit{72.27pt}%
1686 \ifnum\bysize@pixelx=0\relax
1687 \expandafter\@firstofone
1688 \else
1689 \expandafter\@gobble
1690 \fi
1691 {%
1692 \@bysize@num@two\bysize@pixelx
1693 \@bysize@num@two\bysize@pixely
1694 }%
1695 \@bysize@stop
1696 \@nil
1697 \@bysize@end
1698 }%
1699 \</base>

```

### 2.2.12 sgi

```

begin_sgi
big-endian
read_10_0
grab_2_>_temp
check_streq_temp[0x01_0xDA]_magic:_474_decimal
grab_1_>_temp_ storage:_0_0r_1
check_numge_temp_0
check_numle_temp_1
skip_2_ bpc,_dimension
num_2_>_pixelwidth
num_2_>_pixelheight
ok
end

```

\bysize@read@sgi

```

1700 (*base)
1701 \def\bysize@read@sgi#1{%
1702 \@bysize@init
1703 \@bysize@bigendiantrue
1704 \@bysize@read{#1}{10}{0}%
1705 \@bysize@grab\bysize@temp{2}%
1706 \@bysize@skip@two
1707 \ifnum\pdf@strcmp{\bysize@temp}{01DA}=\z@
1708 \else

```



```

1709 \expandafter\@bysize@stop
1710 \fi
1711 \@bysize@grab\bysize@temp{1}%
1712 \@bysize@skip@one
1713 \ifnum\bysize@temp<0\relax
1714 \expandafter\@bysize@stop
1715 \fi
1716 \ifnum\bysize@temp>1\relax
1717 \expandafter\@bysize@stop
1718 \fi
1719 \@bysize@skip@two
1720 \@bysize@num@two\bysize@pixelwidth
1721 \@bysize@num@two\bysize@pixelheight
1722 \@bysize@ok
1723 \@bysize@stop
1724 \@nil
1725 \@bysize@end
1726 }%
1727 \base

```

## 2.3 Package bysize

```

1728 (*package)
1729 \ProvidesPackage{bysize}%
1730 [2016/05/16 v1.7 Extract size/resolution from bitmap files (HO)]%
1731 \RequirePackage{ifpdf}
1732 \ifpdf
1733 \PackageInfo{bysize}{Superseded by pdfTeX in PDF mode}%
1734 \expandafter\endinput
1735 \fi
1736 \RequirePackage{pdftexcmds}[2007/11/11]
1737 \begingroup\expandafter\expandafter\expandafter\endgroup
1738 \expandafter\ifx\csname pdf@filedump\endcsname\relax
1739 \PackageError{bysize}{%
1740 You need pdfTeX 1.30.0 or newer%
1741 }{Package loading is aborted.}%
1742 \expandafter\endinput
1743 \fi
1744
1745 \RequirePackage{infwarerr}[2007/09/09]
1746 \RequirePackage{graphics}

```

In case of plain T<sub>E</sub>X options are not executed and \KV@err and \KV@errx are undefined.

```

1747 \RequirePackage{keyval}\relax
1748 \expandafter\ifx\csname KV@errx\endcsname\relax
1749 \def\KV@errx#1{%
1750 \@PackageError{keyval}{#1}\@ehc
1751 }%
1752 \fi
1753 \expandafter\ifx\csname KV@err\endcsname\relax
1754 \let\KV@err\KV@errx
1755 \fi
1756 \RequirePackage{bysize-base}
1757
1758 \InputIfFileExists{bysize-\Gin@driver}{-}{-}
1759
1760 \define@key{Gin}{bysizefast}[true]{%

```

```

1761 \expandafter\ifx\csname if#1\expandafter\endcsname\csname iftrue\endcsname
1762 \@bysize@fasttrue
1763 \else
1764 \@bysize@fastfalse
1765 \fi
1766 }
1767 \define@key{Gin}{resolutionunit}{%
1768 \def\bysize@unit@default{#1}%
1769 }
1770 \beginngroup
1771 \def\x#1{\endgroup
1772 \define@key{Gin}{resolution}{%
1773 \@bysize@read@resolution\@bysize@user@resolutiontrue##1#1#1\@nil
1774 }%
1775 \define@key{Gin}{defaultresolution}{%
1776 \@bysize@read@resolution\@bysize@user@resolutionfalse##1#1#1\@nil
1777 }%
1778 }%
1779 \x{ }
1780 \def\bysize@read@resolution#1#2 #3 #4\@nil{%
1781 \ifcase 0\ifx\#2\1\fi
1782 \ifnum\pdf@strcmp{#2}{\Gin@exclamation}=\z@
1783 \ifx\#3\1\fi
1784 \ifnum\pdf@strcmp{#3}{\Gin@exclamation}=\z@
1785 1%
1786 \fi
1787 \fi
1788 \ifcase\pdf@strcmp{#2}{\Gin@exclamation}\relax
1789 \let\bysize@pixelx@default\Gin@exclamation
1790 \else
1791 \edef\bysize@pixelx@default{#2}%
1792 \fi
1793 \ifcase\pdf@strcmp{#3}{\Gin@exclamation}\relax
1794 \let\bysize@pixely@default\Gin@exclamation
1795 \else
1796 \ifx\#3\%
1797 \let\bysize@pixely@default\bysize@pixelx@default
1798 \else
1799 \edef\bysize@pixely@default{#3}%
1800 \fi
1801 \fi
1802 #1%
1803 \else
1804 \PackageError{bysize}{%
1805 Wrong syntax for key (default)resolution%
1806 }{%
1807 See package documentation for correct syntax.%
1808 }%
1809 \fi
1810 }
1811 \newcommand*{\bysize@setup}{\setkeys{Gin}}
1812
1813 \let\bysize@org@setfile\Gin@setfile
1814 \def\Gin@setfile#1#2#3{%
1815 \ifcase\pdf@strcmp{#1}{bmp}\relax
1816 \expandafter\@firstofone
1817 \else
1818 \expandafter\@gobble

```

```

1819 \fi
1820 {%
1821   \bysize@okfalse
1822   \edef\bysize@ext{\ifx\Gin@ext\relax\Gin@eext\else\Gin@ext\fi}%
1823   \edef\bysize@file{\Gin@base\bysize@ext}%
1824   \edef\bysize@temp{\bysize@ext}%
1825   \ifundefined\bysize@read@\@bysize@temp}{%
1826     \ifundefined\bysize@map@\@bysize@temp}{}%
1827     \expandafter\let\expandafter\bysize@temp
1828     \csname bysize@map@\@bysize@temp\endcsname
1829   }%
1830 }{%
1831   \ifundefined\bysize@read@\@bysize@temp}{%
1832   }{%
1833     \csname bysize@read@\@bysize@temp\endcsname\bysize@file
1834   }%
1835   \ifbysize@ok
1836   \else
1837     \@for\@bysize@temp:=\bysize@types\do{%
1838       \ifbysize@ok
1839       \else
1840         \csname bysize@read@\@bysize@temp\endcsname\bysize@file
1841       \fi
1842     }%
1843   \fi
1844   \ifbysize@ok
1845     \ifGin@bbox
1846       \ifundefined\Gin@vllx}{%
1847         \@PackageWarning{bysize}{Explicit bounding box is ignored}%
1848       }{%
1849         \ifx\Gin@viewport@code\relax
1850           \def\Gin@ollx{0}%
1851           \let\Gin@olly\Gin@ollx
1852           \let\Gin@ourx\bysize@width
1853           \let\Gin@oury\bysize@height
1854           \let\Gin@vllx\Gin@ollx
1855           \let\Gin@vllly\Gin@olly
1856           \let\Gin@vurx\Gin@ourx
1857           \let\Gin@vury\Gin@oury
1858           \let\Gin@viewport@code\Gin@viewport
1859           \@PackageWarning{bysize}{%
1860             Explicit bounding box replaced by\MessageBreak
1861             viewport setting%
1862           }%
1863         \else
1864           \@PackageWarning{bysize}{Explicit bounding box is ignored}%
1865         \fi
1866       }%
1867     \fi
1868     \def\Gin@llx{0}%
1869     \def\Gin@lly{0}%
1870     \let\Gin@urx\bysize@width
1871     \let\Gin@ury\bysize@height
1872     \Gin@bboxtrue
1873   \else
1874     \@PackageInfo{bysize}{Unknown image type of \bysize@file}%
1875   \fi
1876 }%

```

```

1877 \bysize@org@setfile{#1}{#2}{#3}%
1878 }
1879 \newcommand*{\bysize@ext@type}[1]{%
1880 \@namedef{bysize@map@#1}%
1881 }
1882 \bysize@ext@type{.jpg}{jpg}
1883 \bysize@ext@type{.jpe}{jpg}
1884 \bysize@ext@type{.jfif}{jpg}
1885 \bysize@ext@type{.jpeg}{jpg}
1886 \bysize@ext@type{.tif}{tiff}
1887 \bysize@ext@type{.tiff}{tiff}
1888 \bysize@ext@type{.pcx}{pcx}
1889 \bysize@ext@type{.msp}{msp}
1890 \bysize@ext@type{.bmp}{bmp}
1891 \bysize@ext@type{.png}{png}
1892 \bysize@ext@type{.pnm}{pnm}
1893 \bysize@ext@type{.pbm}{pnm}
1894 \bysize@ext@type{.pgm}{pnm}
1895 \bysize@ext@type{.ppm}{pnm}
1896 \bysize@ext@type{.pam}{pam}
1897 \bysize@ext@type{.xpm}{xpm}
1898 \bysize@ext@type{.gif}{gif}
1899 \bysize@ext@type{.tga}{tga}
1900 \bysize@ext@type{.sgi}{sgi}
1901 </package>

```

## 2.4 Drivers

### 2.4.1 dvips

Identification.

```

1902 (*dvips)
1903 \ProvidesFile{bysize-dvips.def}%
1904 [2016/05/16 v1.7 Graphics bitmap driver for dvips (H0)]%

```

Ensure correct catcodes.

```

1905 \expandafter\edef\csname @bysize@driver@catcodes\endcsname{%
1906 \catcode44 \the\catcode44 % ,
1907 \catcode58 \the\catcode58 % :
1908 \catcode60 \the\catcode60 % <
1909 \catcode61 \the\catcode61 % =
1910 \catcode62 \the\catcode62 % >
1911 \catcode64 \the\catcode64 % @
1912 }
1913 \catcode64 11 %
1914 \@makeother\,
1915 \@makeother\:
1916 \@makeother\<
1917 \@makeother\=
1918 \@makeother\>

```

\Gininclude@bmp Added features: support for viewport/trim and clip.

```

1919 \def\Gininclude@bmp#1{%
1920 \message{<#1>}%
1921 \raise\Gin@req@height
1922 \hbox to\Gin@req@width{%

```

Clipping support.

```

1923 \ifGin@clip
1924 \vbox to\z@{%

```

```

1925     \special{ps:gsave currentpoint}%
1926     \kern\Gin@req@height
1927     \hbox to\z@{%
1928         \kern\Gin@req@width
1929         \special{ps:%
1930             currentpoint %
1931             newpath %
1932             3 index 3 index moveto %
1933             1 index 3 index lineto %
1934             2 copy lineto %
1935             exch pop exch pop %
1936             lineto %
1937             closepath %
1938             clip %
1939         }%
1940         \hss
1941     }%
1942     \vss
1943 }%
1944 \fi

```

Support for viewport/trim. The original bounding box is ‘0 0 width height’. If package `bmpsize` is used and the image has been recognized, then the original width and height are known (`\bmpsize@width`, `\bmpsize@height`). Otherwise we try the saved values `\Gin@ourx` and `\Gin@oury`. This guessing will fail, if options `viewport` and `trim` are used both or several times. This is a deficiency of package `graphicx`. One of options `viewport` and `trim` should be used at most once.

```

1945     \@ifundefined{Gin@ollx}{%
1946         \dimen@ \z@
1947     }{%
1948         \ifx\Gin@scalex\Gin@exclamation
1949             \let\Gin@scalex\Gin@scaley
1950         \fi
1951         \ifx\Gin@scaley\Gin@exclamation
1952             \let\Gin@scaley\Gin@scalex
1953         \fi
1954         \@ifundefined{bmpsize@width}{%
1955             \let\bmpsize@width\Gin@ourx
1956             \let\bmpsize@height\Gin@oury
1957         }{%
1958             \dimen@=\Gin@llx bp\relax
1959             \dimen@=\Gin@scalex\dimen@
1960             \kern-\dimen@
1961             \advance\Gin@req@width\dimen@
1962             \dimen@=\bmpsize@width bp\relax
1963             \advance\dimen@ by -\Gin@urx bp\relax
1964             \dimen@=\Gin@scalex\dimen@
1965             \advance\Gin@req@width\dimen@
1966             \dimen@=\Gin@lly bp\relax
1967             \dimen@=\Gin@scaley\dimen@
1968             \advance\Gin@req@height\dimen@
1969             \dimen@=\bmpsize@height bp\relax
1970             \advance\dimen@ by -\Gin@ury bp\relax
1971             \dimen@=\Gin@scaley\dimen@
1972             \advance\Gin@req@height\dimen@
1973         }%
1974         \ifdim\dimen@=\z@
1975     \else

```

```

1976      \vbox to\z@\bgroup
1977      \kern-\dimen@
1978      \fi
The special for the image.
1979      \special{em:graph #1,\the\Gin@req@width,\the\Gin@req@height}%
1980      \ifdim\dimen@=\z@
1981      \else
1982          \vss
1983          \egroup
1984      \fi
1985      \ifGin@clip
1986          \special{ps::grestore}%
1987      \fi
1988      \hss
1989      }%
1990 }

1991 \@bysize@driver@catcodes
1992 </dvips>

```

## 2.4.2 dvipdfm and dvipdfmx

Identification.

```

1993 <*dvipdfm>
1994 \ProvidesFile{bysize-dvipdfm.def}%
1995 [2016/05/16 v1.7 Graphics bitmap driver for dvipdfm (H0)]%
1996 </dvipdfm>
1997 <*dvipdfmx>
1998 \ProvidesFile{bysize-dvipdfmx.def}%
1999 [2016/05/16 v1.7 Graphics bitmap driver for dvipdfmx (H0)]%
2000 </dvipdfmx>
2001 <*dvipdfm j dvipdfmx>

```

Ensure correct catcodes.

```

2002 \expandafter\edef\csname @bysize@driver@catcodes\endcsname{%
2003   \catcode44 \the\catcode44 % ,
2004   \catcode46 \the\catcode46 % .
2005   \catcode58 \the\catcode58 % :
2006   \catcode60 \the\catcode60 % <
2007   \catcode61 \the\catcode61 % =
2008   \catcode62 \the\catcode62 % >
2009   \catcode64 \the\catcode64 % @
2010 }
2011 \catcode64 11 %
2012 \@makeother\,
2013 \@makeother\.
2014 \@makeother\:
2015 \@makeother\<
2016 \@makeother\=
2017 \@makeother\>

```

Counter resource to generate unique names for xform objects.

```

2018 \@ifundefined{@bysize@count}{%
2019   \csname newcount\endcsname @bysize@count
2020   \@bysize@count=\z@
2021 }{}

```

The file name is given as PDF string in the image special. If we have pdf<sub>T</sub>EX with `\pdfescapestring` we use it.

`\@bysize@pdfescapestring`

```

2022 \begingroup\expandafter\expandafter\expandafter\endgroup
2023 \expandafter\ifx\csname pdf@escapestring\endcsname\relax
2024   \def\@bysize@pdfescapestring#1{#1}%
2025 \else
2026   \let\@bysize@pdfescapestring\pdf@escapestring
2027 \fi

```

The size of reused images of dvipdfm 0.13.2c is 1bp. It is the default size of an image object in user space. Thus the reused image must be scaled to the requested width and height. The factor is just the conversion from pt to bp (72/72.27).

`\bysize@dvipdfm@factor`

```

2028 (dvipdfm)\def\bysize@dvipdfm@factor{.99626}

```

Unhappily dvipdfmx behaves differently. It remembers the size assuming a resolution of 100 dots per inch and additionally scales the reused image to this size. Thus the scaling factor also depends on the pixel sizes of the image:

- width:  $(72 / 72.27) * (100 / 72) / \text{pixelwidth} = 100 / 72.27 / \text{pixelwidth}$
- height:  $100 / 72.27 / \text{pixelheight}$

Recent versions however use the natural size of the reused image. Thus the factor is the difference between the requested size and the natural size.

`\Gininclude@bmp` Added features: support for viewport/trim, clip, and image reuse.

```

2029 \def\Gininclude@bmp#1{%
2030   \message{<#1>}%

```

Clip support is achieved by putting the image inside a xform object. These xform objects are automatically clipped when they are used.

```

2031   \ifGin@clip
2032     \global\advance\@bysize@count\@ne
2033     \edef\@bysize@clip@name{@CLIP@the\@bysize@count}%
2034     \special{%
2035       pdf:bxobj \@bysize@clip@name\space
2036       width \the\Gin@req@width\space
2037       height \the\Gin@req@height
2038     }%
2039   \fi

```

Support for viewport/trim.

```

2040   \hbox to \z@{%
2041     \@ifundefined{Gin@ollx}{%
2042       \dimen@z@
2043     }{%
2044       \ifx\Gin@scalex\Gin@exclamation
2045         \let\Gin@scalex\Gin@scaley
2046       \fi
2047       \ifx\Gin@scaley\Gin@exclamation
2048         \let\Gin@scaley\Gin@scalex
2049       \fi
2050       \@ifundefined{bysize@width}{%
2051         \let\bysize@width\Gin@ourx
2052         \let\bysize@height\Gin@oury
2053       }{}%
2054       \dimen@=\Gin@llx bp\relax
2055       \dimen@=\Gin@scalex\dimen@

```

```

2056 \kern-\dimen@
2057 \advance\Gin@req@width\dimen@
2058 \dimen@=\bysize@width bp\relax
2059 \advance\dimen@ by -\Gin@urx bp\relax
2060 \dimen@=\Gin@scalex\dimen@
2061 \advance\Gin@req@width\dimen@
2062 \dimen@=\bysize@height bp\relax
2063 \advance\dimen@ by -\Gin@ury bp\relax
2064 \dimen@=\Gin@scaley\dimen@
2065 \advance\Gin@req@height\dimen@
2066 \dimen@=\Gin@lly bp\relax
2067 \dimen@=\Gin@scaley\dimen@
2068 \advance\Gin@req@height\dimen@
2069 }%
2070 \ifdim\dimen@=\z@
2071 \else
2072 \vbox to\z@\bgroup
2073 \kern\dimen@
2074 \fi

```

Reuse support, dvipdfm just remember the image. The requested sizes, clipping, ...do not matter. In case of dvipdfmx we also must remember the natural size.

```

2075 \edef\@bysize@temp{\IMG@\@bysize@pdfescapestring{#1}}%
2076 \ifundefined{\@bysize@temp}{%
2077 \global\advance\@bysize@count\@ne
2078 (*dvipdfm)
2079 \expandafter\xdef\csname\@bysize@temp\endcsname{%
2080 \the\@bysize@count
2081 }%
2082 /dvipdfm)
2083 (*dvipdfmx)
2084 \expandafter\ifx\csname bysize@pixelwidth\endcsname\relax
2085 \else
2086 \expandafter\xdef\csname\@bysize@temp\endcsname{%
2087 \the\@bysize@count:\bysize@width:\bysize@height
2088 }%
2089 \fi
2090 /dvipdfmx)
2091 \special{%
2092 pdf:image @IMG\the\@bysize@count\space
2093 width \the\Gin@req@width\space
2094 height \the\Gin@req@height\space
2095 depth 0pt (\@bysize@pdfescapestring{#1})%
2096 }%
2097 }{%
2098 (*dvipdfm)
2099 \special{%
2100 pdf:bt %
2101 xscale \strip@pt\dimexpr
2102 \bysize@dvipdfm@factor\Gin@req@width\relax\space
2103 yscale \strip@pt\dimexpr
2104 \bysize@dvipdfm@factor\Gin@req@height\relax
2105 }%
2106 \special{pdf:uobj @IMG\csname\@bysize@temp\endcsname}%
2107 \special{pdf:et}%
2108 /dvipdfm)
2109 (*dvipdfmx)
2110 \expandafter\expandafter\expandafter\@bysize@extract
2111 \csname\@bysize@temp\endcsname\@nil

```



```

2112 \edef\@bysize@xscale{\strip@pt\Gin@req@width}%
2113 \edef\@bysize@temp{\strip@pt\dimexpr\@bysize@width bp}%
2114 \@bysize@div\@bysize@xscale\@bysize@xscale\@bysize@temp
2115 \edef\@bysize@yscale{\strip@pt\Gin@req@height}%
2116 \edef\@bysize@temp{\strip@pt\dimexpr\@bysize@height bp}%
2117 \@bysize@div\@bysize@yscale\@bysize@yscale\@bysize@temp
2118 \special{%
2119 pdf:bt %
2120 xscale \@bysize@xscale\space
2121 yscale \@bysize@yscale
2122 }%
2123 \special{pdf:uxobj @IMG\@bysize@imgnum}%
2124 \special{pdf:et}%
2125 </dvipdfmx>
2126 }%
2127 \ifdim\dimen@=\z@
2128 \else
2129 \vss
2130 \egroup
2131 \fi
2132 \hss
2133 }%
2134 \ifGin@clip
2135 \special{pdf:exobj}%
2136 \special{pdf:uxobj \@bysize@clip@name}%
2137 \fi
2138 }

2139 <*/dvipdfmx>
2140 \def\@bysize@extract#1:#2:#3\@nil{%
2141 \def\@bysize@imgnum{#1}%
2142 \def\@bysize@width{#2}%
2143 \def\@bysize@height{#3}%
2144 }
2145 </dvipdfmx>

2146 \@bysize@driver@catcodes
2147 </dvipdfm j dvipdfmx>

```

## 2.5 Test program bysize-test.tex

```

2148 <*test>
2149 \expandafter\ifx\csname NeedsTeXFormat\endcsname\relax
2150 \input miniltx\relax
2151 \fi
2152 \begingroup\expandafter\expandafter\expandafter\endgroup
2153 \expandafter\ifx\csname pdfoutput\endcsname\relax
2154 \else
2155 \pdfoutput=0 %
2156 \fi
2157 \RequirePackage{bsize}
2158
2159 \endlinechar=-1
2160 \catcode`\@=11
2161 \def\msg#{\immediate\write16}
2162
2163 \def\init{%
2164 \msg{}%
2165 \msg{File name menu}%

```

```

2166 \msg{=====}%
2167 \msg{* Option menu: use `opt' as file name}%
2168 \msg{* Quit program: <return>}%
2169 \msg{}%
2170 \message{Image file name = }%
2171 \read-1 to \imagename
2172 \ifx\imagename\@empty
2173   \expandafter\@firstoftwo
2174 \else
2175   \expandafter\@secondoftwo
2176 \fi
2177 {%
2178   \csname @@end\endcsname
2179 \end
2180 }{%
2181   \ifnum\pdf@strcmp{\imagename}{opt}=\z@
2182     \expandafter\optionmenu
2183   \else
2184     \startimg
2185     \expandafter\init
2186   \fi
2187 }%
2188 }
2189 \def\optionmenu{%
2190   \msg{}%
2191   \msg{Option menu}%
2192   \msg{=====}%
2193   \msg{Current setting:}%
2194   \msg{* bmpsizefast = \if@bmpsize@fast true\else false\fi}%
2195   \msg{* \if@bmpsize@user@resolution\else default\fi resolution = %
2196     \bmpsize@pixelx@default
2197     \space
2198     \bmpsize@pixely@default
2199   }%
2200   \msg{* \if@bmpsize@user@resolution default\fi resolution: not set}%
2201   \msg{* resolutionunit = \bmpsize@unit@default}%
2202   \msg{* Quit option menu: <return>}%
2203   \msg{}%
2204   \message{Options = }%
2205   \read-1 to \options
2206   \ifx\options\empty
2207     \expandafter\init
2208   \else
2209     \edef\@bmpsize@temp{%
2210       \noexpand\setkeys{Gin}{\options}%
2211     }%
2212     \@bmpsize@temp
2213     \expandafter\optionmenu
2214   \fi
2215 }
2216
2217 \def\startimg{%
2218   \let\@found\@empty
2219   \msg{}%
2220   \msg{* File [\imagename]}%
2221   \@for\@type:=\bmpsize@types\do{%
2222     \ifx\@found\@empty
2223       \csname bmpsize@read@\@type\endcsname\imagename

```

```

2224 \ifbysize@ok
2225 \let\@found\@type
2226 \msg{\space\space Type: \@type}%
2227 \msg{\space\space Pixel width: \bysize@pixelwidth\space px}%
2228 \msg{\space\space Pixel height: \bysize@pixelheight\space px}%
2229 \ifx\bysize@pixelx\relax
2230 \else
2231 \ifx\bysize@unit\relax
2232 \let\@unit@spec\@empty
2233 \def\@ratio@name{Ratio }%
2234 \else
2235 \def\@unit@spec{\space dots per \bysize@unit}%
2236 \def\@ratio@name{Density }%
2237 \fi
2238 \msg{\space\space \@ratio@name x: \bysize@pixelx\@unit@spec}%
2239 \msg{\space\space \@ratio@name y: \bysize@pixely\@unit@spec}%
2240 \fi
2241 \msg{\space\space Width: \bysize@width\space bp}%
2242 \msg{\space\space Height: \bysize@height\space bp}%
2243 \ifx\bysize@orientation\relax
2244 \else
2245 \msg{\space\space Orientation: \bysize@orientation}%
2246 \fi
2247 \fi
2248 \fi
2249 }%
2250 \ifx\@found\@empty
2251 \edef\@file@date{\pdf@filemoddate{\imagenamex}}%
2252 \ifx\@file@date\@empty
2253 \msg{\space\space --> File not found <--}%
2254 \else
2255 \msg{\space\space --> Unknown image type <--}%
2256 \fi
2257 \fi
2258 }
2259
2260 \ifx\noinit!\else\expandafter\init\fi
2261 \end{test}

```

## 3 Installation

### 3.1 Download

**Package.** This package is available on CTAN<sup>1</sup>:

[CTAN:macros/latex/contrib/oberdiek/bysize.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/bysize.pdf](#) Documentation.

**Bundle.** All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

TDS refers to the standard “A Directory Structure for T<sub>E</sub>X Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

---

<sup>1</sup>[CTAN:pkg/bysize](#)

## 3.2 Bundle installation

**Unpacking.** Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

**Script installation.** Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

## 3.3 Package installation

**Unpacking.** The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain `TEX`:

```
tex bmpsize.dtx
```

**TDS.** Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

<code>bmpsize.sty</code>	→ <code>tex/latex/oberdiek/bmpsize.sty</code>
<code>bmpsize-base.sty</code>	→ <code>tex/latex/oberdiek/bmpsize-base.sty</code>
<code>bmpsize-test.tex</code>	→ <code>tex/latex/oberdiek/bmpsize-test.tex</code>
<code>bmpsize-dvips.def</code>	→ <code>tex/latex/oberdiek/bmpsize-dvips.def</code>
<code>bmpsize-dvipdfm.def</code>	→ <code>tex/latex/oberdiek/bmpsize-dvipdfm.def</code>
<code>bmpsize-dvipdfmx.def</code>	→ <code>tex/latex/oberdiek/bmpsize-dvipdfmx.def</code>
<code>bmpsize.pdf</code>	→ <code>doc/latex/oberdiek/bmpsize.pdf</code>
<code>bmpsize.dtx</code>	→ <code>source/latex/oberdiek/bmpsize.dtx</code>

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

## 3.4 Refresh file name databases

If your `TEX` distribution (`teTEX`, `mikTEX`, ...) relies on file name databases, you must refresh these. For example, `teTEX` users run `texhash` or `mktextlsr`.

## 3.5 Some details for the interested

**Unpacking with  $\LaTeX$ .** The `.dtx` chooses its action depending on the format:

**plain `TEX`:** Run `docstrip` and extract the files.

**$\LaTeX$ :** Generate the documentation.

If you insist on using  $\LaTeX$  for `docstrip` (really, `docstrip` does not need  $\LaTeX$ ), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{bmpsize.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

**Generating the documentation.** You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfL<sup>A</sup>T<sub>E</sub>X:

```
pdflatex_bmpsize.dtx
makeindex_s_gind.ist_bmpsize.idx
pdflatex_bmpsize.dtx
makeindex_s_gind.ist_bmpsize.idx
pdflatex_bmpsize.dtx
```

## 4 References

- [1] D.P. Carlisle, The L<sup>A</sup>T<sub>E</sub>X Project: *Packages in the ‘graphics’ bundle*, 2005/11/14; [CTAN:macros/latex/required/graphics/grfguide.pdf](http://www.ctan.org/ctan/graphics/grfguide.pdf).

### 4.1 URLs for bitmap format descriptions

#### 4.1.1 JPEG

- <https://www.w3.org/Graphics/JPEG/jfif3.pdf>
- <http://exif.org/Exif2-2.PDF>

#### 4.1.2 PNG

- <https://en.wikipedia.org/wiki/PNG>
- <https://www.w3.org/TR/PNG/>

#### 4.1.3 GIF

- <https://www.w3.org/Graphics/GIF/spec-gif89a.txt>

#### 4.1.4 BMP

- [https://en.wikipedia.org/wiki/Windows\\_bitmap](https://en.wikipedia.org/wiki/Windows_bitmap)
- <https://docs.microsoft.com/en-us/windows/win32/gdi/bitmap-storage>
- <https://docs.microsoft.com/en-us/windows/win32/api/wingdi/ns-wingdi-bitmapfileheader>

#### 4.1.5 PCX

- <https://en.wikipedia.org/wiki/PCX>
- <https://de.wikipedia.org/wiki/PCX>
- <http://www.qzx.com/pc-gpe/pcx.txt>

#### 4.1.6 MSP

- [https://en.wikipedia.org/wiki/Microsoft\\_Paint](https://en.wikipedia.org/wiki/Microsoft_Paint)
- Sources of dvips.

#### 4.1.7 TIFF

- <https://en.wikipedia.org/wiki/TIFF>
- <https://www.adobe.io/content/dam/udp/en/open/standards/tiff/TIFF6.pdf>

#### 4.1.8 TGA

- [https://de.wikipedia.org/wiki/Targa\\_Image\\_File](https://de.wikipedia.org/wiki/Targa_Image_File)
- [https://en.wikipedia.org/wiki/Truevision\\_TGA](https://en.wikipedia.org/wiki/Truevision_TGA)
- <http://www.dca.fee.unicamp.br/~martino/disciplinas/ea978/tgaffs.pdf>

#### 4.1.9 SGI

- [https://en.wikipedia.org/wiki/Silicon\\_Graphics\\_Image](https://en.wikipedia.org/wiki/Silicon_Graphics_Image)
- <ftp://ftp.sgi.com/graphics/SGIIMAGESPEC>

#### 4.1.10 WMF

- <http://www.fileformat.info/format/wmf/>

#### 4.1.11 XPM

- [https://en.wikipedia.org/wiki/XPM\\_%28image\\_format%29](https://en.wikipedia.org/wiki/XPM_%28image_format%29)
- <https://de.wikipedia.org/wiki/Xpm>
- <http://koala.ilog.fr/ftp/pub/xpm/xpm-README.html>

## 5 History

[2006/08/24 v1.0]

- First version.

[2007/02/18 v1.1]

- 1in replaced by 72.27pt, because TeX is inaccurate if 1in is given.

[2007/04/11 v1.2]

- Line ends sanitized.

### [2007/05/01 v1.3]

- Uses package `infwarerr`.
- Image reuse algorithm fixed for `dvipdfmx`.
- Some support for Exif's orientation tag.

### [2007/11/11 v1.4]

- Use of package `pdftexcmds` for Lua<sub>TEX</sub> support.
- Fix of bug of package `keyval`: `\KV@err` and `\KV@errx` are used, but undefined if loaded by plain  $\TeX$ .

### [2008/08/11 v1.5]

- Code is not changed.
- Update of URLs.

### [2009/09/04 v1.6]

- Fixes for reusing objects with `dvipdfmx-20090708`. Older versions of `dvipdfmx` are no longer supported.

### [2016/05/16 v1.7]

- Documentation updates.

## 6 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; plain numbers refer to the code lines where the entry is used.

Symbols	
<code>\,</code> .....	1914, 2012
<code>\.</code> .....	2013
<code>\:</code> .....	1915, 2014
<code>\&lt;</code> .....	1916, 2015
<code>\=</code> .....	1917, 2016
<code>\&gt;</code> .....	1918, 2017
<code>\@</code> .....	2160
<code>\@PackageError</code> .....	1750
<code>\@PackageWarning</code> ...	1847, 1859, 1864
<code>\@bmpsizer@swap</code> .....	135, 138
<code>\@bmpsizer@abs@byte</code> ...	172, 181, 188
<code>\@bmpsizer@abs@maybe</code>	168, 194, 202, 210
<code>\@bmpsizer@absnumfalse</code> .....	29, 819
<code>\@bmpsizer@absnumtrue</code> .....	817
<code>\@bmpsizer@append</code> .....	83, 1080, 1120, 1237, 1261, 1315, 1464, 1481, 1513, 1530
<code>\@bmpsizer@beautify</code>	221, 223, 392, 393
<code>\@bmpsizer@bigendianfalse</code> .....	587, 779, 833, 876, 1554, 1624, 1665
<code>\@bmpsizer@bigendiantrue</code> .....	28, 407, 718, 882, 1703
<code>\@bmpsizer@break</code> ....	46, 615, 1051, 1059, 1065, 1088, 1106, 1128, 1184, 1219, 1240, 1254, 1297, 1318, 1332, 1349, 1386, 1423, 1431, 1440, 1465, 1489, 1514, 1538
<code>\@bmpsizer@buf</code> .....	59, 61, 66, 72, 75, 87, 143, 146, 149, 151, 156
<code>\@bmpsizer@check@byte</code> ...	61, 117, 128
<code>\@bmpsizer@cleanup@end</code> ..	120, 130, 164
<code>\@bmpsizer@cleanup@frac</code> .....	227, 233
<code>\@bmpsizer@cleanup@fracdigits</code>	237, 240
<code>\@bmpsizer@clip@name</code>	2033, 2035, 2136
<code>\@bmpsizer@corr</code> .	373, 375, 385, 387, 388
<code>\@bmpsizer@count</code> .....	2019, 2020, 2032, 2033, 2077, 2080, 2087, 2092
<code>\@bmpsizer@div</code>	219, 323, 324, 2114, 2117
<code>\@bmpsizer@driver@catcodes</code>	1991, 2146
<code>\@bmpsizer@end</code> .....	280, 465, 773, 827, 860, 992, 1138,

```

1360, 1548, 1618, 1659, 1697, 1725
\@bysize@extract ..... 2110, 2140
\@bysize@fastfalse ..... 1764
\@bysize@fasttrue ..... 22, 1762
\@bysize@fillbuf ..... 65, 1026,
1042, 1071, 1096, 1111, 1155,
1174, 1201, 1227, 1244, 1279,
1305, 1322, 1339, 1377, 1394,
1404, 1414, 1449, 1472, 1498, 1521
\@bysize@grab ..... 155, 193, 200,
208, 409, 417, 430, 449, 472,
483, 518, 569, 579, 781, 835,
868, 1001, 1007, 1015, 1027,
1043, 1072, 1097, 1112, 1147,
1156, 1175, 1190, 1202, 1228,
1245, 1259, 1280, 1306, 1323,
1340, 1367, 1378, 1395, 1405,
1415, 1450, 1473, 1499, 1522,
1556, 1582, 1626, 1667, 1705, 1711
\@bysize@grab@byte ... 156, 159, 162
\@bysize@height ..... 2116, 2143
\@bysize@imgnum ..... 2123, 2141
\@bysize@init ..... 24,
406, 470, 778, 832, 865, 997,
1143, 1365, 1553, 1623, 1664, 1702
\@bysize@isdigit .....
.. 106, 1074, 1114, 1212, 1230,
1290, 1308, 1458, 1475, 1507, 1524
\@bysize@iswhite ..... 90, 1017,
1029, 1082, 1099, 1122, 1158,
1204, 1282, 1452, 1483, 1501, 1532
\@bysize@loop .....
42, 44, 46, 427, 481, 608, 894,
1025, 1041, 1070, 1095, 1110,
1154, 1173, 1200, 1226, 1243,
1278, 1304, 1321, 1338, 1376,
1403, 1413, 1448, 1471, 1497, 1520
\@bysize@num@four . 207, 416, 423,
424, 429, 447, 448, 599, 683,
685, 686, 704, 706, 707, 816,
818, 822, 823, 889, 935, 945,
955, 957, 958, 976, 978, 979, 1580
\@bysize@num@one .....
..... 192, 489, 531, 847, 1633
\@bysize@num@two ..... 199, 504,
549, 550, 561, 594, 606, 620,
629, 658, 752, 753, 766, 790,
798, 799, 810, 811, 844, 845,
884, 892, 906, 915, 1576, 1577,
1605, 1606, 1639, 1640, 1641,
1643, 1654, 1655, 1680, 1681,
1683, 1684, 1692, 1693, 1720, 1721
\@bysize@ok ..... 16, 425, 763,
800, 812, 820, 857, 893, 1135,
1268, 1545, 1578, 1653, 1682, 1722
\@bysize@org@plain@loop .... 25, 396
\@bysize@org@setfile .... 1813, 1877
\@bysize@pdfescapestring .....
..... 2022, 2075, 2095
\@bysize@plain@loop ..... 6, 26
\@bysize@pushback .....
. 86, 1064, 1105, 1239, 1317, 1337
\@bysize@read ..... 58,
408, 428, 446, 471, 482, 517,
568, 605, 618, 684, 705, 751,
780, 821, 834, 867, 890, 904,
956, 977, 999, 1144, 1188, 1366,
1555, 1579, 1604, 1625, 1666, 1704
\@bysize@read@resolution .....
..... 1773, 1776, 1780
\@bysize@size ..... 48, 1579
\@bysize@skip@four ... 148, 216,
410, 411, 418, 431, 519, 570,
627, 656, 681, 702, 787, 788,
789, 913, 933, 943, 953, 974,
1191, 1368, 1369, 1573, 1574,
1581, 1583, 1584, 1585, 1586, 1668
\@bysize@skip@one .....
..... 142, 197, 450, 474, 484,
520, 837, 843, 1002, 1008, 1016,
1028, 1044, 1073, 1098, 1113,
1149, 1157, 1176, 1192, 1203,
1229, 1246, 1260, 1281, 1307,
1324, 1341, 1370, 1379, 1396,
1406, 1416, 1451, 1474, 1500,
1523, 1557, 1627, 1632, 1638, 1712
\@bysize@skip@two .....
..... 145, 205, 473, 530,
571, 580, 628, 657, 682, 703,
782, 797, 809, 815, 836, 842,
846, 869, 914, 934, 944, 954,
975, 1148, 1575, 1587, 1706, 1719
\@bysize@stop ..... 40, 51, 54,
76, 125, 174, 212, 414, 421, 438,
459, 463, 477, 487, 496, 528,
591, 597, 602, 761, 764, 771,
785, 801, 813, 825, 840, 858,
880, 887, 901, 990, 1005, 1010,
1013, 1023, 1090, 1130, 1136,
1152, 1221, 1269, 1299, 1358,
1374, 1467, 1491, 1516, 1540,
1546, 1570, 1594, 1602, 1616,
1630, 1636, 1647, 1651, 1657,
1677, 1695, 1709, 1714, 1717, 1723
\@bysize@swap@maybe .. 132, 201, 209
\@bysize@temp .....
60, 63, 169, 174, 176, 196, 204,
215, 370, 371, 372, 377, 378,
1824, 1825, 1826, 1827, 1828,
1831, 1833, 1837, 1840, 2075,
2076, 2079, 2086, 2106, 2111,
2113, 2114, 2116, 2117, 2209, 2212
\@bysize@trunc ..... 225, 230, 277
\@bysize@user@resolutionfalse 1776
\@bysize@user@resolutiontrue . 1773

```



\@bysize@width	2113, 2142	\@type	2221, 2223, 2225, 2226
\@bysize@xscale	2112, 2114, 2120	\@unit@spec	2232, 2235, 2238, 2239
\@bysize@yscale	2115, 2117, 2121	\\	61, 130, 156, 235, 1781, 1783, 1796
\@car	171		
\@ehc	1750		
\@empty	50, 66, 75, 135, 2172, 2218, 2222, 2232, 2250, 2252	<b>A</b>	
\@file@date	2251, 2252	\advance	1961, 1963, 1965, 1968, 1970, 1972, 2032, 2057, 2059, 2061, 2063, 2065, 2068, 2077
\@firstofone	67, 433, 441, 452, 491, 506, 512, 522, 541, 563, 573, 610, 622, 651, 660, 666, 676, 690, 697, 711, 748, 755, 792, 804, 851, 896, 908, 928, 938, 948, 962, 969, 983, 1020, 1032, 1054, 1102, 1179, 1249, 1327, 1344, 1381, 1389, 1398, 1408, 1418, 1428, 1435, 1455, 1504, 1561, 1567, 1591, 1597, 1608, 1672, 1687, 1816	<b>B</b>	
\@firstoftwo	499, 533, 556, 582, 871, 1036, 1046, 1075, 1083, 1115, 1123, 1159, 1167, 1194, 1205, 1213, 1231, 1263, 1272, 1283, 1291, 1309, 1459, 1476, 1484, 1508, 1525, 1533, 2173	\bysize@calc@pixelx	332, 336, 340, 342, 346, 348, 350, 351, 356
\@for	1837, 2221	\bysize@calc@pixely	333, 337, 338, 340, 342, 343, 348, 350, 357
\@found	2218, 2222, 2225, 2250	\bysize@calc@unit	331, 335, 363, 367, 370
\@gobble	69, 435, 443, 454, 493, 508, 514, 524, 543, 565, 575, 612, 624, 653, 662, 668, 678, 688, 699, 709, 746, 757, 794, 806, 849, 898, 910, 930, 940, 950, 960, 971, 981, 1018, 1030, 1056, 1100, 1181, 1251, 1329, 1346, 1383, 1391, 1400, 1410, 1420, 1426, 1437, 1453, 1502, 1559, 1565, 1589, 1599, 1610, 1670, 1689, 1818	\bysize@dvi@pdfm@factor	2028, 2102, 2104
\@gobblefour	146, 150, 151	\bysize@entries	606, 609, 617, 892, 895, 903
\@gobbletwo	143	\bysize@exif@density	480, 505, 643, 645, 647
\@ifundefined	1825, 1826, 1831, 1846, 1945, 1954, 2018, 2041, 2050, 2076	\bysize@exif@offset	578, 604, 684, 705
\@makeother	1914, 1915, 1916, 1917, 1918, 2012, 2013, 2014, 2015, 2016, 2017	\bysize@ext	1822, 1823, 1824
\@namedef	1880	\bysize@ext@type	1879, 1882, 1883, 1884, 1885, 1886, 1887, 1888, 1889, 1890, 1891, 1892, 1893, 1894, 1895, 1896, 1897, 1898, 1899, 1900
\@ne	118, 161, 2032, 2077	\bysize@file	1823, 1833, 1840, 1874
\@nil	40, 171, 225, 227, 230, 233, 237, 464, 772, 826, 859, 991, 1137, 1359, 1547, 1617, 1658, 1696, 1724, 1773, 1776, 1780, 2111, 2140	\bysize@fill@buflength	73, 78, 81
\@ratio@name	2233, 2236, 2238, 2239	\bysize@head	1190, 1193, 1261, 1262, 1271, 1337
\@secondoftwo	501, 535, 558, 584, 873, 1038, 1048, 1077, 1085, 1117, 1125, 1161, 1169, 1196, 1207, 1215, 1233, 1265, 1274, 1285, 1293, 1311, 1461, 1478, 1486, 1510, 1527, 1535, 2175	\bysize@height	357, 365, 367, 378, 380, 382, 384, 388, 391, 393, 1853, 1871, 1956, 1969, 2052, 2062, 2087, 2242
		\bysize@length	416, 426, 429, 461, 504, 511, 527, 561, 562, 766, 769
		\bysize@off	604, 605, 607, 618, 619, 1146, 1164, 1172, 1177, 1188, 1189, 1199, 1210, 1218, 1236, 1247, 1277, 1288, 1296, 1314, 1325, 1342
		\bysize@offset	73, 78, 426, 428, 446, 461, 479, 482, 517, 568, 578, 751, 769, 889, 890, 891, 904, 905, 998, 999, 1000, 1145, 1146, 1189, 1371, 1580, 1596, 1604
		\bysize@ok@false	27, 283, 286, 292, 296, 1821
		\bysize@ok@true	16
		\bysize@orientation	37, 671, 2243, 2245
		\bysize@pixel@height	31, 285, 294, 357, 424, 752, 754, 760, 799, 811, 818, 845, 945, 1134, 1335, 1544, 1577, 1640, 1644, 1649, 1681, 1721, 2228

\bysize@pixelwidth	1228, 1230, 1237, 1239, 1245,
..... 30, 282, 290, 356,	1248, 1259, 1261, 1280, 1282,
423, 753, 798, 810, 816, 844,	1290, 1303, 1306, 1308, 1315,
935, 1094, 1257, 1495, 1576,	1317, 1323, 1326, 1340, 1343,
1639, 1642, 1645, 1680, 1720, 2227	1367, 1372, 1378, 1380, 1388,
\bysize@pixelx	1395, 1397, 1405, 1407, 1415,
.. 32, 301, 303, 315, 323, 332,	1417, 1425, 1434, 1450, 1452,
339, 342, 347, 351, 447, 549,	1458, 1464, 1473, 1475, 1481,
685, 822, 854, 957, 1605, 1613,	1483, 1499, 1501, 1507, 1513,
1654, 1683, 1686, 1692, 2229, 2238	1522, 1524, 1530, 1532, 1556,
\bysize@pixelx@default	1558, 1564, 1579, 1582, 1588,
. 336, 399, 1789, 1791, 1797, 2196	1626, 1628, 1633, 1634, 1641,
\bysize@pixelxdenom	1642, 1643, 1644, 1667, 1669,
... 35, 305,	1675, 1705, 1707, 1711, 1713, 1716
308, 318, 321, 323, 325, 693, 965	\bysize@tempnum
\bysize@pixely	..... 1069, 1080, 1094, 1109,
.... 33, 302, 304,	1120, 1134, 1225, 1237, 1257,
316, 324, 333, 343, 350, 448,	1303, 1315, 1335, 1447, 1464,
550, 706, 823, 855, 978, 1606,	1481, 1495, 1496, 1513, 1530, 1544
1607, 1614, 1655, 1684, 1693, 2239	\bysize@types
\bysize@pixely@default	..... 402, 1837, 2221
. 337, 400, 1794, 1797, 1799, 2198	\bysize@unit
\bysize@pixelydenom	..... 34, 317, 328,
... 36, 306,	331, 457, 538, 546, 632, 634,
310, 312, 319, 324, 326, 714, 986	636, 638, 824, 866, 918, 920,
\bysize@read@bmp	922, 924, 1656, 1685, 2231, 2235
..... 776	\bysize@unit@default
\bysize@read@gif	..... 335, 398, 1768, 2201
..... 830	\bysize@width
\bysize@read@jpg	..... 356, 361, 363, 377, 379, 381,
..... 468	383, 387, 390, 392, 1852, 1870,
\bysize@read@misp	1955, 1962, 2051, 2058, 2087, 2241
..... 1662	\bysize@setup
\bysize@read@pam	..... 1811
..... 1141	
\bysize@read@pcx	
..... 1621	
\bysize@read@png	
..... 404	
\bysize@read@pnm	
..... 995	
\bysize@read@sgi	
..... 1700	
\bysize@read@tga	
..... 1551	
\bysize@read@tiff	
..... 863	
\bysize@read@xpm	
..... 1363	
\bysize@tag	
..... 620, 621, 650,	
675, 696, 906, 927, 937, 947, 968	
\bysize@temp	
..... 409,	
412, 417, 419, 430, 432, 440,	
449, 451, 472, 475, 483, 485,	
489, 490, 498, 518, 521, 531,	
532, 540, 555, 569, 572, 579,	
581, 589, 594, 595, 599, 600,	
604, 629, 630, 640, 658, 659,	
665, 671, 683, 684, 686, 687,	
693, 704, 705, 707, 708, 714,	
722, 723, 728, 733, 738, 743,	
745, 781, 783, 790, 791, 803,	
835, 838, 847, 848, 854, 868,	
870, 878, 884, 885, 907, 915,	
916, 955, 956, 958, 959, 965,	
976, 977, 979, 980, 986, 1001,	
1003, 1007, 1009, 1012, 1015,	
1017, 1027, 1029, 1035, 1043,	
1045, 1053, 1064, 1072, 1074,	
1080, 1082, 1097, 1099, 1105,	
1112, 1114, 1120, 1122, 1147,	
1150, 1156, 1158, 1166, 1175,	
1178, 1202, 1204, 1212, 1225,	

## C

\catcode	1906, 1907, 1908, 1909, 1910,
	1911, 1913, 2003, 2004, 2005,
	2006, 2007, 2008, 2009, 2011, 2160
\csname	..... 119, 122, 124,
	127, 1738, 1748, 1753, 1761,
	1828, 1833, 1840, 1905, 2002,
	2019, 2023, 2079, 2084, 2086,
	2106, 2111, 2149, 2153, 2178, 2223

## D

\define@key	... 1760, 1767, 1772, 1775
\dimen@	..... 1946, 1958,
	1959, 1960, 1961, 1962, 1963,
	1964, 1965, 1966, 1967, 1968,
	1969, 1970, 1971, 1972, 1974,
	1977, 1980, 2042, 2054, 2055,
	2056, 2057, 2058, 2059, 2060,
	2061, 2062, 2063, 2064, 2065,
	2066, 2067, 2068, 2070, 2073, 2127
\dimexpr	..... 362, 363, 366,
	367, 370, 2101, 2103, 2113, 2116
\do	..... 1837, 2221

## E

\empty	..... 2206
\end	..... 2179

<code>\endcsname</code> . . . . .	119, 122, 124, 127, 1738, 1748, 1753, 1761, 1828, 1833, 1840, 1905, 2002, 2019, 2023, 2079, 2084, 2086, 2106, 2111, 2149, 2153, 2178, 2223	<code>\if@bysize@fast</code> . . . . .	21, 360, 2194
<code>\endinput</code> . . . . .	1734, 1742	<code>\if@bysize@user@resolution</code> . . . . .	20, 329, 2195, 2200
<code>\endlinechar</code> . . . . .	2159	<code>\ifbysize@ok</code> . . . . .	15, 281, 289, 299, 1835, 1838, 1844, 2224
<b>F</b>		<code>\ifcase</code> . . . . .	184, 241, 242, 243, 244, 245, 246, 247, 248, 249, 300, 328, 630, 640, 723, 916, 1017, 1029, 1074, 1082, 1099, 1114, 1122, 1158, 1204, 1212, 1230, 1282, 1290, 1308, 1452, 1458, 1475, 1483, 1501, 1507, 1524, 1532, 1781, 1788, 1793, 1815
<code>\FPdiv</code> . . . . .	220, 342, 350, 356, 357, 381, 382, 383, 384	<code>\ifdim</code> . . . . .	1974, 1980, 2070, 2127
<code>\FPifint</code> . . . . .	224	<code>\ifGin@bbox</code> . . . . .	1845
<code>\FPMul</code> . . . . .	343, 351, 372, 377, 378, 379, 380, 387, 388	<code>\ifGin@clip</code> . . . . .	1923, 1985, 2031, 2134
<code>\FPround</code> . . . . .	390, 391	<code>\ifnum</code> 53, 91, 93, 95, 97, 107, 110, 118, 161, 171, 173, 211, 290, 294, 303, 304, 308, 312, 371, 412, 419, 432, 440, 451, 475, 485, 490, 498, 505, 511, 521, 527, 532, 540, 555, 562, 572, 581, 589, 595, 600, 609, 621, 650, 659, 665, 675, 687, 696, 708, 745, 754, 783, 791, 803, 838, 848, 870, 878, 885, 895, 907, 927, 937, 947, 959, 968, 980, 1003, 1009, 1012, 1035, 1045, 1053, 1150, 1166, 1178, 1193, 1248, 1262, 1271, 1326, 1343, 1372, 1380, 1388, 1397, 1407, 1417, 1425, 1434, 1558, 1564, 1588, 1596, 1607, 1628, 1634, 1645, 1649, 1669, 1675, 1686, 1707, 1713, 1716, 1782, 1784, 2181	
<b>G</b>		<code>\ifpdf</code> . . . . .	1732
<code>\Gin@base</code> . . . . .	1823	<code>\ifx</code> . . . . .	50, 66, 75, 123, 182, 235, 282, 285, 301, 302, 305, 306, 310, 321, 328, 338, 339, 346, 347, 385, 1738, 1748, 1753, 1761, 1781, 1783, 1796, 1822, 1849, 1948, 1951, 2023, 2044, 2047, 2084, 2149, 2153, 2172, 2206, 2222, 2229, 2231, 2243, 2250, 2252, 2260
<code>\Gin@bboxtrue</code> . . . . .	1872	<code>\imagenam</code> . . . . .	2171, 2172, 2181, 2220, 2223, 2251
<code>\Gin@driver</code> . . . . .	1758	<code>\immediate</code> . . . . .	2161
<code>\Gin@eext</code> . . . . .	1822	<code>\init</code> . . . . .	2163, 2185, 2207, 2260
<code>\Gin@exclamation</code> . . . . .	338, 346, 400, 1782, 1784, 1788, 1789, 1793, 1794, 1948, 1951, 2044, 2047	<code>\input</code> . . . . .	2150
<code>\Gin@ext</code> . . . . .	1822	<code>\InputIfFileExists</code> . . . . .	1758
<code>\Gin@llx</code> . . . . .	1854, 1868, 1958, 2054	<code>\iterate</code> . . . . .	7, 9, 11, 12
<code>\Gin@lly</code> . . . . .	1855, 1869, 1966, 2066	<b>K</b>	
<code>\Gin@llx</code> . . . . .	1850, 1851	<code>\kern</code> 1926, 1928, 1960, 1977, 2056, 2073	
<code>\Gin@lly</code> . . . . .	1851	<code>\KV@err</code> . . . . .	1754
<code>\Gin@ourx</code> . . . . .	1852, 1955, 2051	<code>\KV@errx</code> . . . . .	1749, 1754
<code>\Gin@oury</code> . . . . .	1853, 1956, 2052		
<code>\Gin@req@height</code> . . . . .	. 1921, 1926, 1968, 1972, 1979, 2037, 2065, 2068, 2094, 2104, 2115		
<code>\Gin@req@width</code> . . . . .	. 1922, 1928, 1961, 1965, 1979, 2036, 2057, 2061, 2093, 2102, 2112		
<code>\Gin@scalex</code> 1948, 1949, 1952, 1959, 1964, 2044, 2045, 2048, 2055, 2060			
<code>\Gin@scaley</code> 1949, 1951, 1952, 1967, 1971, 2045, 2047, 2048, 2064, 2067			
<code>\Gin@setfile</code> . . . . .	1813, 1814		
<code>\Gin@urx</code> . . . . .	1856, 1870, 1963, 2059		
<code>\Gin@ury</code> . . . . .	1857, 1871, 1970, 2063		
<code>\Gin@viewport</code> . . . . .	1858		
<code>\Gin@viewport@code</code> . . . . .	1849, 1858		
<code>\Gin@vllx</code> . . . . .	1854		
<code>\Gin@vllly</code> . . . . .	1855		
<code>\Gin@vurx</code> . . . . .	1856		
<code>\Gin@vury</code> . . . . .	1857		
<code>\Gin@include@bmp</code> . . . . .	1919, 2029		
<b>H</b>			
<code>\hbox</code> . . . . .	1922, 1927, 2040		
<code>\hss</code> . . . . .	1940, 1988, 2132		
<b>I</b>			
<code>\if@bysize@absnum</code> . . . . .	19, 170		
<code>\if@bysize@bigendian</code> . . . . .	18, 133		

<b>L</b>		<code>\repeat</code> ..... 6
<code>\loop</code> ..... 25, 26, 396		<code>\RequirePackage</code> .... 4, 5, 14, 1731, 1736, 1745, 1746, 1747, 1756, 2157
<b>M</b>		<b>S</b>
<code>\message</code> ..... 1920, 2030, 2170, 2204		<code>\setkeys</code> ..... 1811, 2210
<code>\MessageBreak</code> ..... 1860		<code>\space</code> .. 104, 114, 2035, 2036, 2092, 2093, 2094, 2102, 2120, 2197, 2226, 2227, 2228, 2235, 2238, 2239, 2241, 2242, 2245, 2253, 2255
<code>\msg</code> ..... 2161, 2164, 2165, 2166, 2167, 2168, 2169, 2190, 2191, 2192, 2193, 2194, 2195, 2200, 2201, 2202, 2203, 2219, 2220, 2226, 2227, 2228, 2238, 2239, 2241, 2242, 2245, 2253, 2255		<code>\special</code> ..... 1925, 1929, 1979, 1986, 2034, 2091, 2099, 2106, 2107, 2118, 2123, 2124, 2135, 2136
<b>N</b>		<code>\starting</code> ..... 2184, 2217
<code>\newcommand</code> ..... 1811, 1879		<code>\strip@pt</code> ..... 362, 366, 2101, 2103, 2112, 2113, 2115, 2116
<code>\newif</code> ..... 15, 18, 19, 20, 21		
<code>\noinit</code> ..... 2260		<b>T</b>
<code>\number</code> ..... 195, 203, 214, 370		<code>\the</code> ... 78, 128, 162, 176, 426, 461, 578, 604, 607, 617, 619, 693, 714, 722, 769, 854, 891, 903, 905, 965, 986, 1164, 1172, 1177, 1189, 1199, 1210, 1218, 1236, 1247, 1277, 1288, 1296, 1314, 1325, 1342, 1579, 1604, 1642, 1644
<code>\numexpr</code> 78, 128, 162, 176, 426, 446, 461, 517, 568, 578, 604, 607, 617, 619, 684, 693, 705, 714, 722, 751, 769, 854, 891, 903, 905, 965, 986, 1164, 1172, 1177, 1189, 1199, 1210, 1218, 1236, 1247, 1277, 1288, 1296, 1314, 1325, 1342, 1579, 1604, 1642, 1644		
<b>O</b>		
<code>\optionmenu</code> ..... 2182, 2189, 2213		
<code>\options</code> ..... 2205, 2206, 2210		
<b>P</b>		<b>U</b>
<code>\PackageError</code> ..... 1739, 1804		<code>\unless</code> ..... 511, 562, 659, 665
<code>\PackageInfo</code> ..... 1733, 1874		
<code>\pdf@escapestring</code> ..... 2026		<b>V</b>
<code>\pdf@filedump</code> ..... 59, 73		<code>\vbox</code> ..... 1924, 1976, 2072
<code>\pdf@filemoddate</code> ..... 2251		<code>\vss</code> ..... 1942, 1982, 2129
<code>\pdf@filesize</code> ..... 49		<b>W</b>
<code>\pdf@strcmp</code> ..... 91, 93, 95, 97, 107, 110, 173, 211, 412, 419, 432, 440, 475, 485, 521, 572, 581, 589, 783, 838, 870, 878, 1003, 1009, 1012, 1035, 1045, 1053, 1150, 1166, 1178, 1193, 1248, 1262, 1271, 1326, 1343, 1372, 1380, 1388, 1397, 1407, 1417, 1425, 1434, 1558, 1564, 1588, 1628, 1669, 1675, 1707, 1782, 1784, 1788, 1793, 1815, 2181		<code>\write</code> ..... 2161
<code>\pdf@unescapehex</code> ..... 1094, 1134, 1257, 1335, 1495, 1544		
<code>\pdfoutput</code> ..... 2155		<b>X</b>
<code>\ProvidesFile</code> ..... 1903, 1994, 1998		<code>\x</code> ..... 1771, 1779
<code>\ProvidesPackage</code> ..... 2, 1729		
<b>R</b>		<b>Z</b>
<code>\raise</code> ..... 1921		<code>\z@</code> ..... 91, 93, 95, 97, 107, 110, 173, 211, 290, 294, 303, 304, 308, 312, 412, 419, 432, 440, 475, 485, 521, 572, 581, 589, 783, 838, 870, 878, 1003, 1009, 1012, 1035, 1045, 1053, 1150, 1166, 1178, 1193, 1248, 1262, 1271, 1326, 1343, 1372, 1380, 1388, 1397, 1407, 1417, 1425, 1434, 1558, 1564, 1588, 1628, 1669, 1675, 1707, 1782, 1784, 1924, 1927, 1946, 1974, 1976, 1980, 2020, 2040, 2042, 2070, 2072, 2127, 2181
<code>\read</code> ..... 2171, 2205		