

# PDFlib Migration Guide

Latest PDFlib versions covered in this document: PDFlib 9.4 and 10.0

The PDFlib API generally remains compatible among major releases. Sometimes API methods, parameters or options are phased out and replaced with an improved successor which is more general or more powerful than the old method. In these situations the new PDFlib version declares the old method as deprecated without changing the functionality in any way. The new method is always preferable. New code should not use deprecated API methods; existing code should switch to the replacement method as soon as possible. Deprecated methods may be removed in a future release.

This Guide contains recommendations for users who migrate existing PDFlib application code which has been developed with an older PDFlib release. It explains how to identify deprecated API features in application code. Once identified the deprecated features should be replaced with the recommended newer ones.

PDFlib 10 removes several old features which have been deprecated in earlier versions. Legacy applications which still use these features must apply the replacements detailed in this document before they can switch to PDFlib 10. In most cases one-to-one replacement methods or options are available.

The majority of changes affects only very old applications which haven't been updated in many years.

*Note* Since PDFlib 10 removes deprecated features it is strongly recommended to check your application for deprecated method calls and options with PDFlib 9.4 to ensure forward compatibility.

## 1 Check for deprecated PDFlib API Features

*Note* PDFlib 10 contains almost no deprecated features. You can use PDFlib 9.3 or above to check whether your application uses legacy features from earlier PDFlib versions.

### 1.1 Identify deprecated API Method Calls at Compile Time

Deprecated API calls can be identified at compile time for some language bindings. Note that only warnings are emitted; the code can still be compiled successfully. Nevertheless it is strongly recommended to replace deprecated API methods with the recommended replacement method.

**Java binding.** The *pdflib.jar* module is created from source code which contains *@Deprecated* annotations and *@deprecated* Javadoc comments for all deprecated API methods so that the compiler can warn about the use of deprecated methods. The Java compiler emits a warning similar to the following:

```
javac -classpath pdflib.jar:. image.java
Note: image.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
```

If you supply the compiler option *-Xlint:deprecation* the name and location of the deprecated API methods are shown:

```
p.set_option("searchpath={" + searchpath + "}");  
p.get_value("major", 0);
```

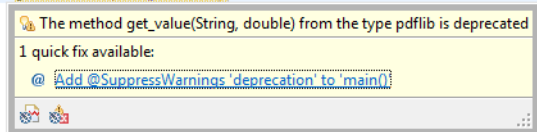


Fig. 1.1  
Presentation of deprecated Java  
API methods in Eclipse

```
javac -classpath pdflib.jar:. -Xlint:deprecation image.java  
image.java:37: warning: [deprecation]  
    get_value(String,double) in pdflib has been deprecated  
    p.get_value( "major", 0.000000);  
      ^
```

The Javadoc documentation for PDFlib lists all deprecated methods along with the recommended replacement method, e.g.

```
double get_value(java.lang.String key, double modifier) Deprecated.  
Use PDF_get_option().
```

In Eclipse you must attach *pdflib.jar* as library to make information about deprecated API methods available to the integrated Java compiler. If deprecated API methods are found in your code, the name of the deprecated method is displayed with a strike-out, and a pop-up box informs about the method's status (see Figure 1.1). These messages are also listed in the *Problems* view.

**.NET binding with Visual Studio.** The PDFlib package marks all deprecated API methods so that calls to deprecated methods are marked with an underline, and IntelliSense pop-ups are shown with a warning for calls to deprecated API methods when editing the code.

The PDFlib assembly also contains »Obsolete« attributes for all deprecated API methods so that the compiler can warn about the use of deprecated API methods.

Visual Studio emits warnings for calls to deprecated API methods if warning level 2 is enabled in *Project, Properties, Build, Errors and warnings*. The warning includes the recommended replacement method:

```
image.cs(37,13): warning CS0618: 'PDFlib_dotnet.PDFlib.get_value(string, double)' is  
obsolete: 'Deprecated, use PDF_get_option().'
```

**.NET binding with command-line compiler.** The *dotnet run* command also emits warnings for deprecated API methods in the compilation phase, provided the *<WarningLevel>* element in the project file is set to 2 or higher:

```
image.cs(37,13): warning CS0618: 'PDFlib.get_value(string, double)' is obsolete:  
'Deprecated, use PDF_get_option().'
```

**C++ binding with Visual Studio, GCC or Clang.** The header file *pdflib.hpp* contains suitable attributes for all deprecated API methods so that the compiler can warn about the use of deprecated methods.

Visual Studio emits warnings for deprecated API calls if warning level 3 is enabled in *Project, Properties, Configuration Properties, C/C++, General/Warning Level: Level 3 (/W3)*:

```
image.cpp(33) : warning C4996: 'pdflib::basic_PDFlib<pstring,conv>::get_value':
was declared deprecated
    with
    [
        pstring=std::wstring,
        conv=pdflib::NoOpConverter<std::wstring>
    ]
```

With GCC and Clang the option `-Wdeprecated-declarations` must be supplied:

```
image.cpp:33:25: warning: 'double pdflib::basic_PDFlib<pstring, conv>::get_value(const
pstring&, double) [with pstring = std::__cxx11::basic_string<wchar_t>; conv =
pdflib::NoOpConverter<std::__cxx11::basic_string<wchar_t> >]' is deprecated
[-Wdeprecated-declarations]
    p.get_value(L"major", 0);
```

**C binding with Visual Studio, GCC or Clang.** The `pdflib.h` header file contains suitable attributes for all deprecated API methods so that the compiler can warn about the use of deprecated methods.

Visual Studio emits warnings for deprecated API calls if warning level 3 is enabled in *Project, Properties, Configuration Properties, C/C++, General/Warning Level: Level 3 (/W3)*:

```
image.c(44): warning C4996: 'PDF_get_value': was declared deprecated
```

With GCC and Clang the option `-Wdeprecated-declarations` must be supplied:

```
image.c:44:2: warning: 'PDF_get_value' is deprecated [-Wdeprecated-declarations]
    PDF_get_value(p, "major", 0);
```

## 1.2 Identify deprecated API Method Calls at Runtime

Using the methods below you can create a list of deprecated API method calls at runtime. These API methods must be replaced in the application; see Section 2.1, »Deprecated PDFlib API Methods«, for details.

**Logging for all language bindings.** Calls to deprecated API methods are identified in the log file for all language bindings with logging class `api=1`. It can be set as follows (see PDFlib API Reference for more details on logging):

```
p.set_option("logging={filename=deprecated.log classes={api=1}}");
```

With this setting the log file contains a warning for calls to deprecated API methods, e.g.

```
PDF_get_value(p_0x2599c20, "major", 0.000000)
[PDF_get_value() is deprecated since PDFlib 9]
```

**PHP language binding.** The PDFlib extension module for PHP contains information about deprecated API methods so that the PHP interpreter can warn about deprecated methods. Depending on the PHP configuration warnings about deprecated method calls are written to the configured PHP error log file, the Web server log file, the Windows event log/system log, or `stderr/stdout`. The following directive in `php.ini` configures the log file:

```
error_log = /var/log/php.errors
```

The default is empty; see [www.php.net/manual/en/errorfunc.configuration.php](http://www.php.net/manual/en/errorfunc.configuration.php) for details.

Whether or not warnings about deprecated calls are written to the log file depends on the `error_reporting` directive which must include `E_DEPRECATED`, e.g.

```
error_reporting = E_ALL
```

See [www.php.net/manual/en/function.error-reporting.php](http://www.php.net/manual/en/function.error-reporting.php) for details. Once logging for deprecated calls is configured, PHP reports all deprecated calls as follows in the PHP error log file (not the PDFlib log file):

```
PHP Deprecated: PDFlib::get_value(): Deprecated, use PDF_get_option(). in /home/bind/php/image.php on line 37
```

**Perl language binding.** The PDFlib module for Perl emits warnings about deprecated API methods at runtime if the predefined warning category `deprecated` is enabled, which is true by default. Perl emits a warning similar to the following:

```
PDF_get_value(): Deprecated, use PDF_get_option(). at image.pl line 31.
```

Use the following instruction or the `-X` command-line option of the Perl interpreter to disable warnings about deprecated API methods on a module basis:

```
no warnings 'deprecated';
```

Warnings for deprecated API methods can be enabled on a module basis with the following instruction in the application code:

```
use warnings 'deprecated';
```

or by calling the Perl interpreter with the `-W` command-line option.

## 1.3 Identify deprecated Options

**All language bindings.** Deprecated options cannot be detected at compile time, but only at runtime. They can be identified in the logging output with the logging class `api=1` which can be set as follows:

```
p.set_option("logging={filename=deprecated.log classes={api=1}}");
```

With this setting the log file contains a message for calls with deprecated options, e.g.

```
PDF_create_annotation(p_0x1529c20, 100.000000, 400.000000, 400.000000, 500.000000,
"FileAttachment", "filename=foo.jpg mimetype=image/jpeg")
[Option "filename" is deprecated since PDFlib 9]
[Option "mimetype" is deprecated since PDFlib 9]
```

These options must be replaced in the application; see Section 2.2, »Deprecated Options and Keywords which are no longer available in PDFlib 10«, for details.

## 2 Deprecated PDFlib API Features

### 2.1 Deprecated PDFlib API Methods

Table 2.1 lists deprecated API methods which are no longer available in PDFlib 10. Applications must use the replacement methods shown in the table.

Table 2.1 *Deprecated PDFlib API methods and their replacements*

<b>deprecated API method</b>	<b>deprecated since</b>	<b>replacement method</b>
<i>PDF_add_bookmark()</i> <i>PDF_add_bookmark2()</i>	PDFlib 6	<i>PDF_create_bookmark()</i> ; the parent and open parameters correspond to the same-named options
<i>PDF_add_launchlink()</i>	PDFlib 6	<i>PDF_create_action()</i> with type=Launch and <i>PDF_create_annotation()</i> with type=Launch
<i>PDF_add_locallink()</i>	PDFlib 6	<i>PDF_create_action()</i> with type=GoTo and <i>PDF_create_annotation()</i> with type=Link
<i>PDF_add_note()</i> <i>PDF_add_note2()</i>	PDFlib 6	<i>PDF_create_annotation()</i> with type=Text
<i>PDF_add_pdflink()</i>	PDFlib 6	<i>PDF_create_action()</i> with type=GoToR and <i>PDF_create_annotation()</i> with type=Link
<i>PDF_add_thumbnail()</i>	PDFlib 9	none; not functional since PDFlib 9
<i>PDF_add_weblink()</i>	PDFlib 6	<i>PDF_create_action()</i> with type=URI and <i>PDF_create_annotation()</i> with type=Link
<i>PDF_attach_file()</i> <i>PDF_attach_file2()</i>	PDFlib 6	<i>PDF_create_annotation()</i> with type=FileAttachment; the description parameter corresponds to the contents option, the author parameter to the title option and the icon parameter to the iconname option.
<i>PDF_begin_glyph()</i>	PDFlib 9	<i>PDF_begin_glyph_ext()</i>
<i>PDF_begin_page()</i>	PDFlib 6	<i>PDF_begin_page_ext()</i>
<i>PDF_begin_pattern()</i>	PDFlib 9	<i>PDF_begin_pattern_ext()</i>
<i>PDF_begin_template()</i>	PDFlib 7	<i>PDF_begin_template_ext()</i>
<i>PDF_boot()</i> (only C/C++)	PDFlib 7	none; was never functional
<i>PDF_close()</i>	PDFlib 6	<i>PDF_end_document()</i>
<i>PDF_close_pdi()</i>	PDFlib 7	<i>PDF_close_pdi_document()</i>
<i>PDF_end_page()</i>	PDFlib 6	<i>PDF_end_page_ext()</i>
<i>PDF_end_template()</i>	PDFlib 8	<i>PDF_end_template_ext()</i>
<i>PDF_findfont()</i>	PDFlib 5	<i>PDF_load_font()</i>
<i>PDF_get_majorversion()</i> <i>PDF_get_minorversion()</i>	PDFlib 4	<i>PDF_get_option()</i> with keys major/minor
<i>PDF_get_parameter()</i> <i>PDF_get_value()</i>	PDFlib 9	For many keys the same-named option of <i>PDF_get_option()</i> can be used; see Table 2.3 for keys which require other replacement methods.
<i>PDF_get_pdi_parameter()</i> <i>PDF_get_pdi_value()</i>	PDFlib 7	<i>PDF_pcos_get_string()</i> and <i>PDF_pcos_get_number()</i> with pCOS paths according to Table 2.2
<i>PDF_initgraphics()</i>	PDFlib 9	<i>PDF_set_graphics_option()</i> with option initgraphicsstate
<i>PDF_open_CCITT()</i>	PDFlib 5	<i>PDF_load_image()</i> with CCITT-compressed TIFF images
<i>PDF_open_file()</i>	PDFlib 6	<i>PDF_begin_document()</i> and <i>PDF_end_document()</i>
<i>PDF_open_image()</i> <i>PDF_open_image_file()</i>	PDFlib 5	<i>PDF_load_image()</i>
<i>PDF_open_mem()</i>	PDFlib 6	<i>PDF_begin_document()</i> with empty filename and <i>PDF_get_buffer()</i>

Table 2.1 *Deprecated PDFlib API methods and their replacements*

<b>deprecated API method</b>	<b>deprecated since</b>	<b>replacement method</b>
<code>PDF_open_pdi()</code>	PDFlib 7	<code>PDF_open_pdi_document()</code>
<code>PDF_place_image()</code>	PDFlib 5	<code>PDF_fit_image()</code> with <code>dpi=None</code>
<code>PDF_place_pdi_page()</code>	PDFlib 5	<code>PDF_fit_pdi_page()</code>
<code>PDF_set_border_color()</code>	PDFlib 6	<code>PDF_create_annotation()</code> with option <code>annotcolor</code>
<code>PDF_set_border_dash()</code>	PDFlib 6	<code>PDF_create_annotation()</code> with option <code>dasharray</code>
<code>PDF_set_border_style()</code>	PDFlib 6	<code>PDF_create_annotation()</code> with options <code>borderstyle</code> and <code>linewidth</code>
<code>PDF_set_parameter()</code> <code>PDF_set_value()</code>	PDFlib 9	For many keys the same-named option of <code>PDF_set_option()</code> can be used; see Table 2.3 for keys which require other replacement methods.
<code>PDF_setdash()</code> <code>PDF_setdashpattern</code>	PDFlib 9	<code>PDF_set_graphics_option()</code> and other methods with graphics appearance options with option <code>dasharray</code> and <code>dashphase</code>
<code>PDF_setflat()</code>	PDFlib 9	<code>PDF_set_graphics_option()</code> or <code>PDF_create_gstate()</code> with option <code>flatness</code>
<code>PDF_setgray()</code> <code>PDF_setgray_fill()</code> <code>PDF_setgray_stroke()</code>	PDFlib 4	<code>PDF_set_graphics_option()</code> and other methods with color options
<code>PDF_setlinejoin()</code>	PDFlib 9	<code>PDF_set_graphics_option()</code> or <code>PDF_create_gstate()</code> with option <code>linejoin</code>
<code>PDF_setlinecap()</code>	PDFlib 9	<code>PDF_set_graphics_option()</code> or <code>PDF_create_gstate()</code> with option <code>linecap</code>
<code>PDF_setmiterlimit()</code>	PDFlib 9	<code>PDF_set_graphics_option()</code> or <code>PDF_create_gstate()</code> with option <code>miterlimit</code>
<code>PDF_setpolydash()</code>	PDFlib 5	<code>PDF_set_graphics_option()</code> and other methods with graphics appearance options with option <code>dasharray</code>
<code>PDF_setrgbcolor()</code> <code>PDF_setrgbcolor_fill()</code> <code>PDF_setrgbcolor_stroke()</code>	PDFlib 4	<code>PDF_set_graphics_option()</code> and other methods with color options
<code>PDF_show_boxed()</code> <code>PDF_show_boxed2()</code>	PDFlib 6	single lines: <code>PDF_fit_textline()</code> multi-line formatting: <code>PDF_add/create_textflow()</code> and <code>PDF_fit_textflow()</code> with options <code>minspacing=100%</code> <code>maxspacing=10000%</code> <code>nofitlimit=100%</code> <code>shrinklimit=100%</code> to achieve similar formatting If <code>PDF_show_boxed()</code> has been called with <code>mode=fulljustify</code> use the following Textflow options: <code>lastalignment=justify</code> <code>alignment=justify</code>
<code>PDF_shutdown()</code> (only C/C++)	PDFlib 7	none; was never functional
<code>PDF_utf16_to_utf8()</code> <code>PDF_utf8_to_utf16()</code> <code>PDF_utf32_to_utf8()</code> <code>PDF_utf8_to_utf32()</code> <code>PDF_utf16_to_utf32()</code> <code>PDF_utf32_to_utf16()</code>	PDFlib 8.1	<code>PDF_convert_to_unicode()</code> with appropriate parameter <code>inputformat</code> and option <code>outputformat</code>
<code>PDF_xshow()</code> (only C/C++)	PDFlib 9	<code>PDF_fit_textline()</code> with option <code>xadvancelist</code>

**pCOS paths for deprecated `PDF_get_pdi_value/parameter()` methods.** The deprecated (as of PDFlib 7) methods `PDF_get_pdi_value()` and `PDF_get_pdi_parameter()` use custom keys for addressing objects in a PDF documents. When switching to `PDF_pcos_get_number()` and `PDF_pcos_get_string()` these keys must be replaced with the corresponding pCOS paths according to Table 2.2 (see pCOS Path Reference for details).

Table 2.2 pCOS paths as replacements for deprecated keys of `PDF_get_pdi_value()` and `PDF_get_pdi_parameter()`

key for <code>PDF_get_pdi_value()</code> and <code>PDF_get_pdi_parameter()</code>	pCOS path for <code>PDF_pcos_get_number()</code> and <code>PDF_pcos_get_string()</code>
vdp/blockcount	length:pages[...]/blocks
/Root/Pages/Count	length:pages
width, height	pages[...]/width, pages[...]/height
/Rotate	pages[...]/Rotate
version	pdfversion
/CropBox, /BleedBox, /ArtBox, /TrimBox, /MediaBox	pages[...]/CropBox[0] for llx, pages[...]/CropBox[1] for lly, pages[...]/CropBox[2] for urx, pages[...]/CropBox[3] for ury etc.
vdp/Blocks/<name>/<property>	pages[...]/blocks/<name>/<property>
vdp/Blocks[...]/<property>	pages[...]/blocks[...]/<property>
vdp/Blocks/<name>/Custom/<property>	pages[...]/blocks/<name>/Custom/<property> or pages[...]/blocks[...]/Custom/<property>
isempty	pages[...]/isempty
filename	filename
/Info/<key>	/Info/<key>
tagged	tagged
pdfx	pdfx

**Replacements for deprecated global values and parameters.** In most cases the deprecated methods `PDF_set/get_value()` and `PDF_set/get_parameter()` can be replaced with `PDF_set/get_option()` and the previous parameter `key` as option name. Table 2.3 lists parameters which must be replaced with other methods and options.

Table 2.3 *Deprecated global parameters for `PDF_set/get_value()` and `PDF_set/get_parameter()` and their replacements*

<i>deprecated parameter</i>	<i>deprecated since</i>	<i>replacement method and option</i>
<b>resource categories</b>		
Encoding, FontAFM, FontOutline, FontPFM, HostFont, ICCProfile, resourcefile, searchpath	PDFlib 9	same-named options of <code>PDF_set_option()</code> ; multiple entries can be set with a single call; <code>PDF_get_option()</code> : use option <code>resourcenum</code> to iterate over all values
StandardOutputIntent	PDFlib 9	none; use embedded or referenced ICC profile as output intent
<b>document properties</b>		
compatibility	PDFlib 6	<code>PDF_begin_document()</code> with option <code>compatibility</code>
userpassword, masterpassword, permissions	PDFlib 6	<code>PDF_begin_document()</code> with options <code>userpassword</code> , <code>masterpassword</code> , <code>permissions</code>
pdfx	PDFlib 6	<code>PDF_begin_document()</code> with option <code>pdfx</code>
base	PDFlib 6	<code>PDF_begin_document()</code> with option <code>uri</code>
openmode	PDFlib 6	<code>PDF_begin_document()</code> with option <code>openmode</code>
openaction	PDFlib 6	<code>PDF_begin/end_document()</code> with option <code>destination</code> ; the following deprecated option keywords must be replaced: <code>fitbbox</code> : use <code>type=fitvisible</code> <code>fitheight</code> : use <code>type=fitheight left=0</code> <code>fitpage</code> : use <code>type=fitwindow</code> <code>fitwidth</code> : use <code>type=fitwidth top=10000</code> <code>retain</code> : use <code>type=fixed</code>
hidetoolbar, hidemenubar, hidewindowui, fitwindow, centerwindow, displaydoctitle, nonfullscreenpagemode, direction, viewarea, viewclip, printarea, printclip	PDFlib 6	<code>PDF_begin/end_document()</code> with option <code>viewer-preferences</code> and respective suboptions;  the suboptions <code>viewarea</code> , <code>viewclip</code> , <code>printarea</code> , <code>printclip</code> are deprecated in PDF 2.0 (see Table 2.5)
<b>page properties</b>		
pageheight, pagewidth in <code>PDF_set_parameter()</code>	PDFlib 6	<code>PDF_begin/end_page_ext()</code> with options <code>pagewidth</code> and <code>pageheight</code>
ArtBox, BleedBox, CropBox, TrimBox	PDFlib 6	<code>PDF_begin/end_page_ext()</code> with options <code>artbox</code> , <code>bleedbox</code> , <code>cropbox</code> , <code>trimbox</code>
duration, transition	PDFlib 6	<code>PDF_begin/end_page_ext()</code> with same-named options
<b>color</b>		
defaultgray, defaultrgb, defaultcmyk	PDFlib 6	<code>PDF_begin/end_page_ext()</code> with options <code>defaultgray</code> , <code>defaultrgb</code> , <code>defaultcmyk</code>
setcolor:iccprofilegray/rgb/cmyk	PDFlib 9	global options <code>iccprofilegray/rgb/cmyk</code>
preserveoldpantone	PDFlib 9.3	none; old-style PANTONE color names are no longer in use



Table 2.3 *Deprecated global parameters for PDF\_set/get\_value() and PDF\_set/get\_parameter() and their replacements*

<i>deprecated parameter</i>	<i>deprecated since</i>	<i>replacement method and option</i>
<b>font handling</b>		
autosubsetting, subsetlimit, subsetminsize, fontstyle, unicodemap	PDFlib 7	PDF_load_font() with same-named options
ascender, ascenderfaked, capheight, capheightfaked, descender, descenderfaked, fontencoding, fontname, fontmaxcode, xheight, xheightfaked	PDFlib 7	PDF_info_font() with same-named keywords; for the numerical values fontsize=1 must be supplied; parameter fontencoding: use keyword encoding
autocidfont	PDFlib 8	none; not functional since PDFlib 9
monospace	PDFlib 7	none; only for the obsolete concept of standard CJK fonts
<b>text output</b>		
charref, charspacing, decoration-above, escapesequene, fakebold, font, fontsize, glyphcheck, hori-zscaling, italicangle, kerning, leading, overline, strikeout, strokewidth, textformat, text-rendering, textrise, underline, underlineposition, underlinewidth, wordspacing	PDFlib 9	PDF_set_text_option(), PDF_fit/info_textline(), PDF_fill_textblock() and PDF_add/create_textflow() with same-named options
<b>images</b>		
honoriccprofile	PDFlib 8.2	PDF_load_image() with option honoriccprofile
renderingintent	PDFlib 8.2	PDF_load_image() with option renderingintent
imagewidth, imageheight, image:iccprofile, orientation, resx, resy	PDFlib 8	PDF_info_image() with same-named options
<b>graphics state</b>		
fillrule in PDF_set_parameter()	PDFlib 9.3	PDF_set_graphics_option() with option fillrule
inheritgstate	PDFlib 6	none; not functional
<b>interactive elements</b>		
bookmarkdest	PDFlib 6	PDF_create_bookmark() with options action, destina-tion, fontstyle and textcolor
launchlink:parameters launchlink:operation launchlink:defaultdir	PDFlib 6	PDF_create_action() with options parameters, operation, and defaultdir; however, these are deprecated in PDF 2.0 (see Table 2.5)
<b>miscellaneous</b>		
flush	PDFlib 6	PDF_begin_document_callback() with option flush
pdiusebox	PDFlib 6	PDF_open_pdi_page() with option pdiusebox
iccwarning, fontwarning, glyphwarning, imagewarning, pdiwarning, warning	PDFlib 7	corresponding API method with option errorpolicy
openwarning	PDFlib 6	PDF_begin_document() with option errorpolicy

Table 2.3 *Deprecated global parameters for `PDF_set/get_value()` and `PDF_set/get_parameter()` and their replacements*

<i>deprecated parameter</i>	<i>deprecated since</i>	<i>replacement method and option</i>
honorlang	PDFlib 8	<code>PDF_set_option()</code> with option <code>filenamehandling=honorlang</code>
trace, tracefile, tracemsg	PDFlib 7	<code>PDF_set_option()</code> with option <code>logging</code>
logmsg	PDFlib 9	<code>PDF_set_option()</code> with option <code>userlog</code>
nodemostamp	PDFlib 9	<code>PDF_set_option()</code> with option <code>avoiddemostamp</code>
string	PDFlib 9	<code>PDF_get_string()</code>
version	PDFlib 9	<code>PDF_get_string()</code> with option <code>version</code> and <code>idx=-1</code>

## 2.2 Deprecated Options and Keywords which are no longer available in PDFlib 10

Table 2.4 lists API methods for which some options or keywords for an option are deprecated in PDFlib 9 and no longer available in PDFlib 10.

Table 2.4 *Deprecated PDFlib API options or keywords which have been removed in PDFlib 10, and their replacements*

<i>deprecated option or keyword</i>	<i>deprecated since</i>	<i>replacement option or action</i>
<b>font and text</b>		
<i>PDF_create_textflow()</i> : textwarning	PDFlib 7	errorpolicy
<i>PDF_fit_textline()</i> : locallink, pdflink, weblink	PDFlib 7	matchbox in <i>PDF_fit_textline()</i> and <i>PDF_create_annotation()</i> to create links
<i>PDF_load_font()</i> : autocidfont	PDFlib 9	none; not functional
<i>PDF_load_font()</i> , <i>PDF_info_font()</i> and <i>PDF_fill_textblock()</i> : option and Block property monospace	PDFlib 9	configure TrueType/OpenType fonts or use host fonts
<i>PDF_load_font()</i> : fontwarning	PDFlib 7	errorpolicy
<i>PDF_load_font()</i> : kerning	PDFlib 8	readkerning
text option glyphwarning	PDFlib 7	glyphcheck
<i>PDF_info_textline()</i> : keywords scalex, scaley	PDFlib 9	fitscalex/fitscaley
<i>PDF_info_textline()</i> : keyword unmappedglyphs	PDFlib 8	unmappedchars
<i>PDF_info_textflow()</i> : keyword remainchars	PDFlib 8	none; worked reliably only under certain conditions
<i>PDF_fit/info_textline()</i> , <i>PDF_add/create_textflow()</i> and <i>PDF_fill_textblock()</i> : option features with keywords vrt2 and vert	PDFlib 9.3	none; these features are enabled automatically for fonts in vertical mode
<b>interactive elements</b>		
<i>PDF_create_action()</i> : actionwarning	PDFlib 7	errorpolicy
<i>PDF_create_action()</i> : defaultdir, parameters and operation for type=Launch <sup>1</sup>	PDFlib 9.3	none
<i>PDF_create_annotation()</i> : annotwarning	PDFlib 7	errorpolicy
<i>PDF_create_annotation()</i> : filename and mimetype	PDFlib 9	attachment and options filename and mimetype of <i>PDF_load_asset()</i>
<i>PDF_create_field()</i> , <i>PDF_create_fieldgroup()</i> : fieldwarning	PDFlib 7	none; didn't have any effect
<i>PDF_create_field()</i> , <i>PDF_create_fieldgroup()</i> : errorpolicy	PDFlib 9.3	none; didn't have any effect
<i>PDF_create_bookmark()</i> : option destination with sub-options fontstyle and color	PDFlib 6	options fontstyle and textcolor of <i>PDF_create_bookmark()</i>
Destination options for <i>PDF_add_nameddest()</i> and for the destination option in <i>PDF_create_action()</i> , <i>PDF_create_annotation()</i> , <i>PDF_create_bookmark()</i> , and <i>PDF_begin/end_document()</i> : options fitbbox, fitheight, fitpage, fitwidth, retain	PDFlib 7	type=visible, fitheight, fitwindow, fitwidth, fixed
option filename and type=file		action option with a GoToR action
option name and type=nameddest		action option with a GoTo action
<i>PDF_create_annotation()</i> : type=Movie <sup>1</sup>	PDFlib 9.3	type=RichMedia
<i>PDF_create_action()</i> : type=Movie <sup>1</sup>	PDFlib 9.3	type=RichMediaExecute

Table 2.4 *Deprecated PDFlib API options or keywords which have been removed in PDFlib 10, and their replacements*

<b>deprecated option or keyword</b>	<b>deprecated since</b>	<b>replacement option or action</b>
<b>image and PDF import</b>		
<i>PDF_load_image()</i> , <i>PDF_fill_imageblock()</i> : template	PDFlib 8	templateoptions
<i>PDF_load_image()</i> , <i>PDF_fill_imageblock()</i> : createtemplate, iconname	PDFlib 9	templateoptions
<i>PDF_load_image()</i> , <i>PDF_fill_imageblock()</i> : downsamplmask	PDFlib 9.2	none; rendering bug fixed in Acrobat X
<i>PDF_load_image()</i> , <i>PDF_fill_imageblock()</i> : imagewarning	PDFlib 7	errorpolicy
<i>PDF_open_pdi_document()</i> , <i>PDF_open_pdi_page()</i> , <i>PDF_process_pdi()</i> , <i>PDF_fill_pdfblock()</i> : pdiwarning	PDFlib 7	errorpolicy
<b>ICC profiles</b>		
<i>PDF_load_iccprofile()</i> : metadata	PDFlib 9.3	none; XMP for profiles no longer supported
<i>PDF_load_iccprofile()</i> : iccwarning	PDFlib 7	errorpolicy
<b>miscellaneous</b>		
Common XObject options and <i>PDF_begin_page_ext()</i> : transparencygroup with suboptions CS, I, K	PDFlib 8.1	suboptions colorspace, isolated, knockout
<i>PDF_add_table_cell()</i> : checkwordsplitting	PDFlib 8.1	avoidwordsplitting
<i>PDF_begin_document()</i> : moddate	PDFlib 9.2	none; Acrobat Preflight bug fixed
<i>PDF_begin_item()</i> and option tag in many methods: inline	PDFlib 9.2	tagging option direct
<i>PDF_set_option()</i> and many other API methods: errorpolicy=legacy	PDFlib 8	errorpolicy=return or errorpolicy=exception
<i>PDF_set_option()</i> : topdown	PDFlib 9	<i>PDF_begin_page_ext()</i> , <i>PDF_begin_template_ext()</i> or <i>PDF_begin_pattern_ext()</i> with option topdown
<i>PDF_get_option()</i> : topdown	PDFlib 9.3	none; applications must take care of options specified earlier
<i>PDF_set_option()</i> : filenamehandling=legacy	PDFlib 9.3	filenamehandling with explicit encoding name or honorlang
<i>PDF_set_option()</i> : licence and licencefile	(was never documented)	use corresponding options license and licencefile

1. This feature is deprecated in PDF 2.0 according to ISO 32000-2.

## 2.3 Deprecated Features in PDF 2.0 and non-ISO Features

**Deprecated PDF 2.0 features which are still available in PDF 1.x.** PDF 2.0 according to ISO 32000-2 deprecates the features listed in Table 2.5. If any of these features is used, PDFlib 9.4 emits a warning (PDFlib 10: warning in PDF 1.x mode and an error in PDF 2.0 mode). It is recommended to avoid these features even in PDF 1.x mode.

Table 2.5 Features deprecated in PDF 2.0 which are still available in PDF 1.x mode

PDFlib features deprecated in PDF 2.0	notes
<code>PDF_begin_document()</code> : permissions with keyword noaccessible	Restricting document accessibility is deprecated in PDF 2.0.
<code>PDF_begin/end_document()</code> : viewerpreferences, suboptions printarea, printclip, viewarea, viewclip	Prepress viewer preferences are deprecated in PDF 2.0.
<code>PDF_begin_page_ext()</code> and <code>PDF_end_page_ext()</code> : separationinfo	Separation dictionaries are deprecated in PDF 2.0.
<code>PDF_create_annotation()</code> : type=Movie	Use type=RichMedia.
<code>PDF_create_action()</code> : type=Movie	Use type=RichMediaExecute.
<code>PDF_create_gstate()</code> : blendmode accepts only a single keyword, but not a list with multiple values	Blendmode arrays are deprecated in PDF 2.0.

**Features outside of ISO 32000-1.** The features listed in Table 2.6 were never part of ISO 32000-1. They work only in Acrobat, but not in third-party viewers. If any of these features is used, PDFlib 9.4 emits a warning (PDFlib 10: warning in PDF 1.x mode and an error in PDF 2.0 mode). It is recommended to avoid these features since they are not part of standard PDF.

Table 2.6 Features outside of ISO 32000 which are still available in PDF 1.x mode

features outside of ISO 32000	notes
<code>PDF_create_field()</code> and <code>PDF_create_fieldgroup()</code> : barcode	Barcode fields work only in the full version of Acrobat, but neither in Acrobat Reader nor third-party viewers.
<code>PDF_begin_document()</code> : search	An attached search index works only in Acrobat, but not in third-party viewers.
<code>PDF_begin_template_ext()</code> : watermark	Editable watermarks work only in Acrobat, but not in third-party viewers.

## 2.4 Obsolete Concepts which have been removed from PDFlib 10

Some concepts are obsolete because there is no longer demand in today's workflows or they haven't been accepted in the marketplace. The items listed in have been removed in PDFlib 10.

Table 2.7 Obsolete PDF concepts which have been removed from PDFlib 10

<b>deprecated concept</b>	<b>deprecated since</b>	<b>notes</b>
<b>PDF versions and standards</b>		
PDF 1.3 (1999): <code>PDF_begin_document()</code> with option <code>compatibility=1.3</code>	PDFlib 8.1	not relevant
PDF/X-1:2001: <code>PDF_begin_document()</code> with option <code>pdfx=PDF/X-1:2001</code>	PDFlib 7	superseded by PDF/X-4
PDF/X-1a:2001, PDF/X-3:2002: <code>PDF_begin_document()</code> with option <code>pdfx=PDF/X-1a:2001</code> and <code>PDF/X-3:2002</code>	PDFlib 8.1	superseded by PDF/X-4
PDF/X-3 standard output intents without embedded ICC profile and resource category <code>StandardOutputIntent</code>	PDFlib 9	superseded by PDF/X-3 and PDF/X-4 with embedded or referenced ICC profile
PDF/X-5g, PDF/X-5pg: <code>PDF_begin_document()</code> with option <code>pdfx=PDF/X-5g</code> and <code>PDF/X-5pg</code>	PDFlib 9.3	not accepted in the marketplace
PDF/VT-2: <code>PDF_begin_document()</code> with option <code>pdfvt=PDF/VT-2</code>	PDFlib 9.3	not accepted in the marketplace
<b>password security</b>		
password encryption: <code>PDF_begin_document()</code> with option <code>userpassword</code> or <code>masterpassword</code> and <code>compatibility=1.4</code> or <code>1.5</code>	PDFlib 9.3.1	This combination results in weak RC4 encryption which should be avoided. Create a newer PDF version, preferably PDF 1.7ext8.
<b>font handling</b>		
Multiple Master PostScript Type 1 fonts	PDFlib 9.3	not relevant
resource fork PostScript Type 1 (LWFN) fonts on macOS	PDFlib 9.3	not relevant
SVG fonts, i.e. fonts in SVG files (not OpenType color fonts with SVG glyphs)	PDFlib 9.3	deprecated in SVG 2.0 and unsupported in most browsers
CEF fonts	PDFlib 9.3	not used as stand-alone fonts
standard CJK fonts and font option/ <code>Block</code> property <code>monospace</code>	PDFlib 9	Configure TrueType/OpenType fonts or use host fonts.
CJK TrueType fonts without Unicode <code>cmap</code> table	PDFlib 9.3	Use Unicode-compatible fonts.
<b>image handling</b>		
images in GIF and CCITT formats	PDFlib 9.4	outdated formats; use PNG or TIFF instead
UCS-2 CMaps	PDFlib 9.3	Use <code>encoding=unicode</code> .
<b>PANTONE spot colors</b>		
PANTONE Goe coated/uncoated PANTONE hexachrome coated/uncoated PANTONE solid in hexachrome coated PANTONE solid to process coated PANTONE solid to process coated EURO	PDFlib 9.1	no longer supported by Pantone, Inc.

Table 2.7 Obsolete PDF concepts which have been removed from PDFlib 10

<b>deprecated concept</b>	<b>deprecated since</b>	<b>notes</b>
PANTONE color bridge CMYK PC	2010	no longer supported by Pantone, Inc. since 2010
PANTONE color bridge CMYK EURO		
PANTONE color bridge uncoated		
PANTONE solid matte		
PANTONE process coated EURO		
PANTONE process uncoated EURO		
PANTONE process coated		
PANTONE process uncoated		
mapping of deprecated PANTONE spot color name suffixes CV, CVV, CVU, CVC and CVP	PDFlib 9.3	old-style PANTONE spot color names are no longer in use
<b>Flash<sup>1</sup></b>		
PDF_load_asset() with type=Flash/Generic/JPEG/PNG	PDFlib 9.3	Flash in RichMedia annotations and Flash-based navigators are no longer supported in Acrobat since December 2020.
PDF_create_annotation() with type=RichMedia and option configuration, suboption instances, suboption params		
PDF_end_document() with option portfolio and suboptions navigator and initialview=custom		
<b>other PDF features</b>		
Reference XObjects: common XObject option reference	PDFlib 9.3	not accepted in the marketplace
page thumbnails: PDF_add_thumbnail()	PDFlib 9	not functional since PDFlib 9.1; PDF viewers create page thumbnails automatically
Open Prepress Interface (OPI) <sup>1</sup> : common XObject options OPI-1.3, OPI-2.0	PDFlib 9.1	OPI is a concept of the 1980s which has very rarely been used in PDF.
PostScript XObjects <sup>1</sup> : PDF_begin_template_ext() with option postscript	PDFlib 9.3	not relevant

1. This feature is deprecated or not included in PDF 2.0 according to ISO 32000-2.

## 2.5 Deprecated Features which are still available in PDFlib 10

Table 2.8 lists deprecated features which are no longer documented, but are still available in PDFlib 10 for the benefit of legacy applications. Users should prepare for future removal of these features and adjust their applications accordingly. New applications should not use any of these features.

Table 2.8 *Deprecated features which are still available in PDFlib 10, but shouldn't be used*

<b>deprecated feature</b>	<b>deprecated since</b>	<b>recommended replacement option or action</b>
Encryption with <code>PDF_begin_document()</code> and the options <code>userpassword/masterpassword</code> with <code>compatibility=1.6, 1.7</code> or <code>1.7ext3</code>	PDFlib 10	Remove the password option or increase PDF output compatibility to <code>1.7ext8</code> or above.
CMaps for CJK legacy encodings, i.e. loading a font with a predefined CMap (this affects only non-Unicode-aware language bindings); font option <code>keepnative</code>	PDFlib 10	Switch to a Unicode workflow.
PostScript Type 1 fonts	PDFlib 9.3.1	Use TrueType or OpenType fonts.
PDF/X-1a:2003: <code>PDF_begin_document()</code> with option <code>pdfx=PDF/X-1a:2003</code>	PDFlib 9.3	Use PDF/X-4.
PDF/X-3:2003: <code>PDF_begin_document()</code> with option <code>pdfx=PDF/X-3:2003</code>	PDFlib 9.4	Use PDF/X-4.
<code>PDF_begin_font()</code> : option <code>colorized</code>	PDFlib 10	Use option <code>colorized</code> of <code>PDF_begin_glyph()</code> ; the color status can now be specified for individual glyphs instead of only for all glyphs in the font.
<code>PDF_info_font()</code> : option <code>shapingsupport</code>	PDFlib 9.3.1	This keyword couldn't be used reliably to test for shaping support of a font.



# 3 PDFlib History

Table 3.1 Release history of PDFlib, PDFlib+PDI and PPS

<i>PDFlib version</i>	<i>release date</i>
<i>PDFlib 4</i>	2001
<i>PDFlib 5</i>	2003
<i>PDFlib 6</i>	2004
<i>PDFlib 7</i>	2006
<i>PDFlib 8</i>	2009
<i>PDFlib 8.1</i>	2011
<i>PDFlib 9</i>	2013
<i>PDFlib 9.1</i>	2016
<i>PDFlib 9.2</i>	2019
<i>PDFlib 9.3</i>	2020
<i>PDFlib 9.3.1</i>	2021
<i>PDFlib 9.4</i>	2022
<i>PDFlib 10</i>	2022