

Creating PDF Documents

linux.conf.au

Jan 24, 2003. Perth

Rasmus Lerdorf <rasmus@php.net>

<http://rookery.linux.conf.au/>

PDFlib License

The Aladdin Free Public License allows the use of PDFlib for:

- o Developing any non-commercial software provided source code is available
- o Personal use
- o Redistribute PDFlib

Commercial license required for any commercial use including commercial web services.

For pricing see <http://pdflib.com>

Alternatives

- o FreePDFlib (open source) by Thomas Szadel
- o ClibPDF (terms similar to pdflib)

Why use PDF with PHP?

- o Invoices for e-commerce sites
- o Report generation
- o Anything that requires precise control over printable output

Hello World Example

```
<?php
    $p = PDF_new();
    PDF_open_file($p);
    PDF_set_info($p, "Creator", "hello.php");
    PDF_set_info($p, "Author", "Rasmus Lerdorf");
    PDF_set_info($p, "Title", "Hello world (PHP)");
    PDF_begin_page($p, 595, 842);
    $font = PDF_findfont($p, "Helvetica-Bold", "host", 0);
    PDF_setfont($p, $font, 38.0);
    PDF_show_xy($p, "Hello world!", 50, 700);
    PDF_end_page($p);
    PDF_close($p);
    $buf = PDF_get_buffer($p);
    $len = strlen($buf);
    Header("Content-type: application/pdf");
    Header("Content-Length: $len");
    Header("Content-Disposition: inline; filename=hello_php.pdf");
    echo $buf;
    PDF_delete($p);
?>
```

Origin is at the lower left and the basic unit is the DTP pt.

1 pt = 1/72 inch = 0.35277777778 mm

Some common page sizes

Format	Width	Height
US-Letter	612	792
US-Legal	612	1008
US-Ledger	1224	792
11x17	792	1224
A0	2380	3368
A1	1684	2380
A2	1190	1684
A3	842	1190
A4	595	842
A5	421	595
A6	297	421
B5	501	709

Coordinate Example

```
<?php
    $p = PDF_new();
    PDF_open_file($p);
    PDF_set_info($p, "Creator", "coords.php");
    PDF_set_info($p, "Author", "Rasmus Lerdorf");
    PDF_set_info($p, "Title", "Coordinate Test (PHP)");
    PDF_begin_page($p, 595, 842);
    $font = PDF_findfont($p, "Helvetica-Bold", "host", 0);
    PDF_setfont($p, $font, 38.0);
    PDF_show_xy($p, "Bottom Left", 10, 10);
    PDF_show_xy($p, "Bottom Right", 350, 10);
    PDF_show_xy($p, "Top Left", 10, 802);
    PDF_show_xy($p, "Top Right", 410, 802);
    PDF_show_xy($p, "Center", 595/2-60, 842/2-20);
    PDF_end_page($p);
    PDF_set_parameter($p, "openaction", "fitpage");
    PDF_close($p);
    $buf = PDF_get_buffer($p);
    $len = strlen($buf);
    Header("Content-type:application/pdf");
    Header("Content-Length:$len");
    Header("Content-Disposition:inline; filename=coords.pdf");
    echo $buf;
    PDF_delete($p);
?>
```

It is possible to move the origin and also reverse the direction. ie. you can make the top-left corner the origin and have positive coordinates indicate down and to the right using something like this:

Coordinate Example

```
<?php
  $p = PDF_new();
  PDF_open_file($p);
  PDF_set_info($p, "Creator", "coords.php");
  PDF_set_info($p, "Author", "Rasmus Lerdorf");
  PDF_set_info($p, "Title", "Coordinate Test (PHP)");
  PDF_begin_page($p, 595, 842);
  PDF_translate($p, 0, 842); // Move Origin
  PDF_scale($p, 1, -1); // Reflect across horizontal axis
  PDF_set_value($p, "horizscaling", -100); // Mirror
  $font = PDF_findfont($p, "Helvetica-Bold", "host", 0);
  PDF_setfont($p, $font, -38.0);
  PDF_show_xy($p, "Top Left", 10, 40);
  PDF_end_page($p);
  PDF_set_parameter($p, "openaction", "fitpage");
  PDF_close($p);
  $buf = PDF_get_buffer($p);
  $len = strlen($buf);
  Header("Content-type:application/pdf");
  Header("Content-Length:$len");
  Header("Content-Disposition:inline; filename=coords.pdf");
  echo $buf;
  PDF_delete($p);
?>
```



pdf_show_xy()

Output text at a given position.

```
<? pdf_show_xy($p, "Text", 50, 100); ?>'>>
```

pdf_show()

Output text at the current position set with pdf_set_text_pos().

```
<? pdf_set_text_pos($p, 50, 100);  
pdf_show($p, "Text"); ?>'>>
```

pdf_continue_text()

Output text on the next line.

```
<? pdf_continue_text($p, "Next line."); ?>
```

pdf_show_boxed()

Format text within the defined box.

pdf_show_boxed() Example

```
<?php  
$p = PDF_new();  
PDF_open_file($p);  
PDF_begin_page($p, 595, 842);  
$font = PDF_findfont($p, "Helvetica-Bold", "host", 0);  
PDF_setfont($p, $font, 38.0);  
$text = <<<FOO  
This is an example of some text inside a text box in a PDF document.  
FOO;  
PDF_show_boxed($p, $text, 50, 630, 300, 200, "left");  
PDF_rect($p, 50, 630, 300, 200); PDF_stroke($p);  
PDF_show_boxed($p, $text, 50, 420, 300, 200, "right");  
PDF_rect($p, 50, 420, 300, 200); PDF_stroke($p);  
PDF_show_boxed($p, $text, 50, 210, 300, 200, "justify");  
PDF_rect($p, 50, 210, 300, 200);  
PDF_stroke($p);  
PDF_show_boxed($p, $text, 50, 0, 300, 200, "fulljustify");  
PDF_rect($p, 50, 0, 300, 200);  
PDF_stroke($p);  
PDF_show_boxed($p, $text, 375, 250, 200, 300, "center");  
PDF_rect($p, 375, 250, 200, 300);  
PDF_stroke($p);  
PDF_end_page($p);  
PDF_set_parameter($p, "openaction", "fitpage");  
PDF_close($p);  
$buf = PDF_get_buffer($p);  
$len = strlen($buf);  
Header("Content-type:application/pdf");  
Header("Content-Length:$len");  
Header("Content-Disposition:inline; filename=coords.pdf");  
echo $buf;  
PDF_delete($p);  
>>
```

Underline, Overline and Strikeout Text are set using a call to `PDF_set_parameter()`.

```
<? pdf_set_parameter($p,"underline","true");
  pdf_set_parameter($p,"overline","true");
  pdf_set_parameter($p,"strikeout","true");
  pdf_show_xy($p,"Text",50,100); ?>
```

Text Rendering

0	Normal	4	Normal Text - add to clipping path
1	Stroke text (outlined)		Stroke text - add to clipping path
2	Fill and stroke text	6	Fill and stroke text - add to clipping path
3	Invisible Text	7	Invisible text - add to clipping path

Text Attributes and Rendering Example

```
<?php
  $p = PDF_new();
  PDF_open_file($p);
  PDF_begin_page($p,595,842);
  $font = PDF_findfont($p,"Helvetica-Bold","host",0);
  PDF_setfont($p,$font,38.0);
  PDF_set_parameter($p, "overline", "true");
  PDF_show_xy($p, "Overlined Text", 50,780);
  PDF_set_parameter($p, "overline", "false");
  PDF_set_parameter($p, "underline", "true");
  PDF_continue_text($p, "Underlined Text");
  PDF_set_parameter($p, "strikeout", "true");
  PDF_continue_text($p, "Underlined strikeout Text");
  PDF_set_parameter($p, "underline","false");
  PDF_set_parameter($p, "strikeout","false");
  PDF_setcolor($p,"fill","rgb", 1.0, 0.1, 0.1);
  PDF_continue_text($p, "Red Text");
  PDF_setcolor($p,"fill","rgb", 0, 0, 0);
  PDF_set_value($p,"textrendering",1);
  PDF_setcolor($p,"stroke","rgb", 0, 0.5, 0);
  PDF_continue_text($p, "Green Outlined Text");
  PDF_set_value($p,"textrendering",2);
  PDF_setcolor($p,"fill","rgb", 0, .2, 0.8);
  PDF_setlinewidth($p,2);
  PDF_continue_text($p, "Green Outlined Blue Text");

  PDF_end_page($p);
  PDF_close($p);
  $buf = PDF_get_buffer($p);
  $len = strlen($buf);
  Header("Content-type:application/pdf");
  Header("Content-Length:$len");
  Header("Content-Disposition:inline; filename=coords.pdf");
  echo $buf;
  PDF_delete($p);
?>
```


14 Built-in Fonts

Courier	Courier-Bold
Courier-Oblique	Courier-BoldOblique
Helvetica	Helvetica-Bold
Helvetica-Oblique	Helvetica-BoldOblique
Times-Roman	Times-Bold
Times-Italic	Times-BoldItalic
Symbol	ZapfDingbats

You can also use other AFM fonts, postscript type-1 and TTF fonts. To use a TTF font, for example:

```
<?php
PDF_set_parameter($p, "FontOutline",
                  "CANDY==/usr/fonts/candy.ttf");
$font = PDF_findfont($p, "CANDY", "host", 1);
?>
```

CANDY Font Example

Fonts can also be defined in the pdflib.upr file.

Character Encodings

winansi	Windows code page 1252, a superset of ISO 8859-1
macroman	Mac Roman encoding, i.e., the default Macintosh character set
ebcdic	EBCDIC code page 1047 as used on IBM AS/400 and S/390 systems
builtin	Original encoding used by non-text (symbol) or non-Latin text fonts
host	Macroman on the Mac, ebcdic on EBCDIC-based systems, and winansi on all other

Font Example

```
<?php
$p = PDF_new();
PDF_open_file($p);
PDF_set_info($p, "Creator", "hello.php");
PDF_set_info($p, "Author", "Rasmus Lerdorf");
PDF_set_info($p, "Title", "Hello world (PHP)");
pdf_set_parameter($p, "resourcefile", "/usr/share/fonts/pdflib/pdflib.upr");
PDF_begin_page($p, 595, 842);
PDF_set_text_pos($p, 25, 800);
$fonts = array('Courier'=>0, 'Courier-Bold'=>0, 'Courier-BoldOblique'=>0,
              'Courier-Oblique'=>0, 'Helvetica'=>0, 'Helvetica-Bold'=>0,
              'Helvetica-BoldOblique'=>0, 'Helvetica-Oblique'=>0,
              'Times-Bold'=>0, 'Times-BoldItalic'=>0, 'Times-Italic'=>0,
              'Times-Roman'=>0, 'LuciduxSans'=>1, 'Utopia-Regular'=>1,
              'URWGothicL-BookObli'=>1, 'URWPalladioL-Roma'=>1,
              'NimbusMonL-ReguObli'=>1, 'CANDY'=>1, 'Arial'=>1
              );

foreach($fonts as $f=>$embed) {
```

```

    $font = PDF_findfont($p,$f,"host",$embed);
    PDF_setfont($p,$font,25.0);
    PDF_continue_text($p,"$f (".chr(128)." Ç à á â ã ç è é ê)");
}
PDF_end_page($p);
PDF_close($p);
$buf = PDF_get_buffer($p);
$len = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$len");
Header("Content-Disposition:inline; filename=hello_php.pdf");
echo $buf;
PDF_delete($p);
?>

```

The pdflib.upr file entries for this would look like this:

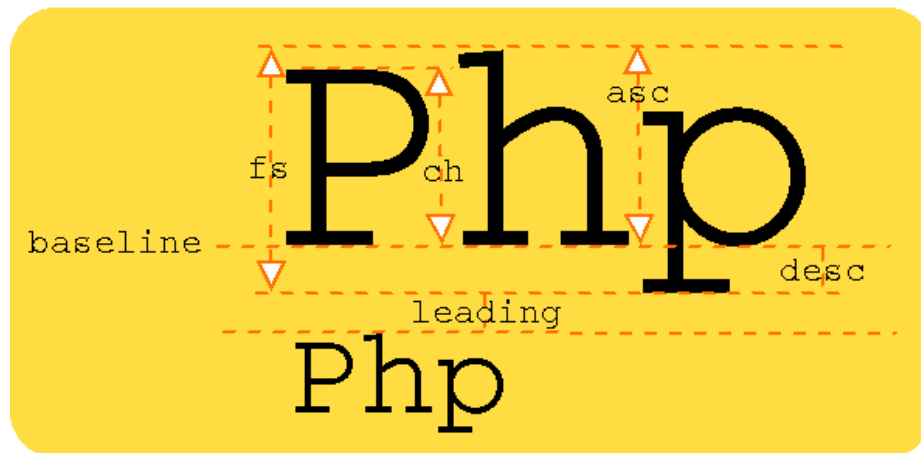
```

FontAFM
LuciduxSans=lcdxsr.afm
Utopia-Regular=UTRG____.afm
URWGothicL-BookObli=a0100331.afm
URWPalladioL-Roma=p0520031.afm
NimbusMonL-ReguObli=n0220231.afm

FontOutline
LuciduxSans=lcdxsr.pfa
Utopia-Regular=UTRG____.pfa
URWGothicL-BookObli=a0100331.pfa
URWPalladioL-Roma=p0520031.pfa
NimbusMonL-ReguObli=n0220231.pfa
CANDY=CANDY.ttf
Arial=arial.ttf

```

The hardest part of creating a PDF document this way is to keep control of blocks of text. You need to know the height of a block of text in order to handle pagination correctly in multipage documents.



Font Characteristics

- o fontsize
- o capheight
- o ascender
- o descender

Text Rendering Attributes

- o leading
- o textrise
- o horizscaling
- o charspacing
- o wordspacing

Calculating the height of a blurb

```
<?
pdf_save($pdf);
pdf_translate($pdf,0,$page_height);
pdf_scale($pdf,1,-1);
pdf_set_font($pdf,$font,12,'winansi');
$leading = pdf_get_value($pdf,"leading");
$height = $inc = 12+$leading;

while(pdf_show_boxed($pdf,$text,$x,$page_height-$y,
                    $page_width-2*$x,$height,$align,
                    'blind')!=0)
    $height+=$inc;

pdf_restore($pdf);
?>
```



Formats

- o PNG (alpha-channel is lost)
- o JPEG (Progressive jpegs supported in Acrobat 4, not 3)
- o GIF (no interlacing is allowed, only first image of an animated gif will be loaded)
- o TIFF
- o CCITT compressed image data
- o Raw image data

Image Example 1

```
<?php
$p = PDF_new();
PDF_open_file($p);
PDF_begin_page($p,595,842);
$im = pdf_open_jpeg($p, "php-big.jpg");
pdf_place_image($p, $im, 200, 700, 1.0);
pdf_place_image($p, $im, 200, 600, 0.75);
pdf_place_image($p, $im, 200, 535, 0.50);
pdf_place_image($p, $im, 200, 501, 0.25);
pdf_place_image($p, $im, 200, 486, 0.10);
$x = pdf_get_value($p, "imagewidth", $im);
$y = pdf_get_value($p, "imageheight", $im);
pdf_close_image ($p,$im);
$font = PDF_findfont($p,"Times-Bold","host",0);
PDF_setfont($p,$font,28.0);
pdf_show_xy($p,"$x by $y",25,800);
PDF_end_page($p);
PDF_close($p);
$buf = PDF_get_buffer($p);
$len = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$len");
Header("Content-Disposition:inline; filename=coords.pdf");
echo $buf;
PDF_delete($p);
?>
```

Scaling an Image

We saw in the previous example how to scale an image linearly. We can also scale non-linearly. To do this we scale the coordinate system appropriately:

Image Example 2

```
<?php
$p = PDF_new();
PDF_open_file($p);
PDF_begin_page($p,595,842);
$im = pdf_open_jpeg($p, "php-big.jpg");
pdf_place_image($p, $im, 200, 700, 1.0);
PDF_save($p); // Save current coordinate system settings
$nx = 50/PDF_get_value($p,"imagewidth",$im);
$ny = 100/PDF_get_value($p,"imageheight",$im);
PDF_scale($p, $nx, $ny);
pdf_place_image($p, $im, 200/$nx, 600/$ny, 1.0);
PDF_restore($p); // Restore previous
pdf_close_image ($p,$im);
PDF_end_page($p);
PDF_close($p);
$buf = PDF_get_buffer($p);
$len = strlen($buf);
Header("Content-type:application/pdf");
```

```
Header("Content-Length:$len");  
Header("Content-Disposition:inline; filename=coords.pdf");  
echo $buf;  
PDF_delete($p);  
?>
```

Paths

To draw graphics in a PDF you define a path which you then perform operations on. For example:

Graphics Example 1

```
<?php
  $p = PDF_new();
  PDF_open_file($p);
  PDF_begin_page($p,595,842);
  PDF_moveto($p,150,750);
  PDF_lineto($p,450,750);
  PDF_lineto($p,100,800);
  PDF_curveto($p,80,500,70,550,250,650);
  PDF_stroke($p);
  PDF_end_page($p);
  PDF_close($p);
  $buf = PDF_get_buffer($p);
  $len = strlen($buf);
  Header("Content-type:application/pdf");
  Header("Content-Length:$len");
  Header("Content-Disposition:inline; filename=gra1.pdf");
  echo $buf;
  PDF_delete($p);
?>
```

We can use `PDF_closepath()` and `pdf_fill_stroke()` to close the path automatically and fill it with the current fill color.

Graphics Example 2

```
<?php
  $p = PDF_new();
  PDF_open_file($p);
  PDF_begin_page($p,595,842);
  PDF_setcolor($p,"fill","rgb", 1.0, 0.8, 0.1);
  PDF_moveto($p,150,750);
  PDF_lineto($p,450,750);
  PDF_lineto($p,100,800);
  PDF_curveto($p,80,500,70,550,250,650);
  PDF_closepath($p);
  PDF_fill_stroke($p);
  PDF_end_page($p);
  PDF_close($p);
  $buf = PDF_get_buffer($p);
  $len = strlen($buf);
  Header("Content-type:application/pdf");
  Header("Content-Length:$len");
  Header("Content-Disposition:inline; filename=gra2.pdf");
  echo $buf;
  PDF_delete($p);
?>
```

Adding a circle, a funky arc and a dashed green box.

Graphics Example 3

```
<?php
  $p = PDF_new();
  PDF_open_file($p);
  PDF_begin_page($p,595,842);
  PDF_setcolor($p,"fill","rgb", 1.0, 0.8, 0.1);
  PDF_moveto($p,150,750);
  PDF_lineto($p,450,750);
  PDF_lineto($p,100,800);
  PDF_curveto($p,80,500,70,550,250,650);
```

```
PDF_closepath($p);
PDF_fill_stroke($p);

// Circle
PDF_setcolor($p,"fill","rgb", 0.8, 0.5, 0.8);
PDF_circle($p,400,600,75);
PDF_fill_stroke($p);

// Funky Arc
PDF_setcolor($p,"fill","rgb", 0.8, 0.5, 0.5);
PDF_moveto($p, 200, 600);
PDF_arc($p,300,600,50,0,120);
PDF_closepath($p);
PDF_fill_stroke($p);

// Dashed rectangle
PDF_setcolor($p,"stroke","rgb", 0.3, 0.8, 0.3);
PDF_setdash($p,4,6);
PDF_rect($p,50,500,500,300);
PDF_stroke($p);

PDF_end_page($p);
PDF_close($p);
$buf = PDF_get_buffer($p);
$len = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$len");
Header("Content-Disposition:inline; filename=gra3.pdf");
echo $buf;
PDF_delete($p);
?>
```


Template Example

```
<?php
  $p = PDF_new();
  PDF_open_file($p);
  $im = pdf_open_jpeg($p, "php-big.jpg");
  $template = pdf_begin_template($p,595,842);
  pdf_save($p);
  pdf_place_image($p, $im, 4, 803, 0.25);
  pdf_place_image($p, $im, 525, 803, 0.25);
  pdf_moveto($p,0,795);
  pdf_lineto($p,595,795);
  pdf_stroke($p);
  $font = PDF_findfont($p,"Times-Bold","host",0);
  PDF_setfont($p,$font,38.0);
  pdf_show_xy($p,"PDF Template Example",100,807);
  pdf_restore($p);
  pdf_end_template($p);
  pdf_close_image ($p,$im);
  PDF_begin_page($p,595,842);
  pdf_place_image($p, $template, 0, 0, 1.0);
  PDF_end_page($p);
  PDF_begin_page($p,595,842);
  pdf_place_image($p, $template, 0, 0, 1.0);
  PDF_end_page($p);
  PDF_close($p);
  $buf = PDF_get_buffer($p);
  $len = strlen($buf);
  Header("Content-type:application/pdf");
  Header("Content-Length:$len");
  Header("Content-Disposition:inline; filename=gra2.pdf");
  echo $buf;
  PDF_delete($p);
?>
```

Pattern Example

```
<?php
  $p = PDF_new();
  PDF_open_file($p);
  $im = pdf_open_png($p, "wa.png");
  $pattern = pdf_begin_pattern($p, 81, 41, 81, 41, 1);
  pdf_save($p);
  pdf_place_image($p, $im, 0, 0, 1);
  pdf_restore($p);
  pdf_end_pattern($p);
  pdf_close_image ($p, $im);
  PDF_begin_page($p, 595, 842);
  PDF_setcolor($p, "fill", "pattern", $pattern);
  PDF_setcolor($p, "stroke", "pattern", $pattern);
  pdf_setlinewidth($p, 50.0);
  PDF_circle($p, 200, 680, 140);
  PDF_stroke($p);
  PDF_end_page($p);
  PDF_close($p);
  $buf = PDF_get_buffer($p);
  $len = strlen($buf);
  Header("Content-type:application/pdf");
  Header("Content-Length:$len");
  Header("Content-Disposition:inline; filename=gra2.pdf");
  echo $buf;
  PDF_delete($p);
?>
```

Bookmarks and thumbnails are easy to add:

Bookmark Example

```
<?php
  $p = PDF_new();
  PDF_open_file($p);

  PDF_begin_page($p,595,842);
  $im = pdf_open_png($p, "fr-flag.png");
  pdf_add_thumbnail($p, $im);
  pdf_close_image($p,$im);
  $top = PDF_add_bookmark($p, "Countries");
  $font = PDF_findfont($p,"Helvetica-Bold","host",0);
  PDF_setfont($p, $font, 20);
  PDF_add_bookmark($p, "France", $top);
  PDF_show_xy($p, "This is a page about France", 50, 800);
  PDF_end_page($p);

  PDF_begin_page($p,595,842);
  $im = pdf_open_png($p, "dk-flag.png");
  pdf_add_thumbnail($p, $im);
  pdf_close_image($p,$im);
  PDF_setfont($p, $font, 20);
  PDF_add_bookmark($p, "Denmark", $top);
  PDF_show_xy($p, "This is a page about Denmark", 50, 800);
  PDF_end_page($p);

  PDF_close($p);
  $buf = PDF_get_buffer($p);
  $len = strlen($buf);
  Header("Content-type:application/pdf");
  Header("Content-Length:$len");
  Header("Content-Disposition:inline; filename=gra2.pdf");
  echo $buf;
  PDF_delete($p);
?>
```

Home Page: <http://www.php.net>

These Slides: <http://conf.php.net>

Manual: <http://www.php.net/pdf>

pdflib: <http://www.pdflib.com>

Index

Introduction	2
Hello World	3
Coordinates	4
Saner Coordinates	5
Carl!	6
Text	7
Text Attributes	8
Fonts	9
Sizing Blocks of text	11
Carl!	12
Images	13
Graphics	15
Templates	17
Patterns	18
Bookmarks	19
Resources	20