

# New Features in mPDF v5.4

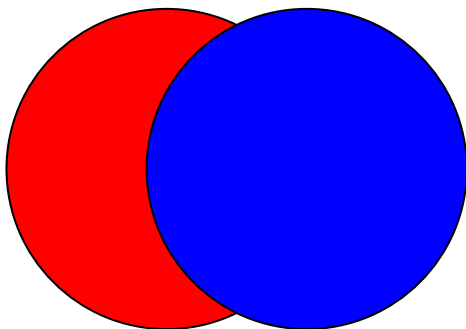
## Bookmark styles

Bookmarks can be styled by adding code as below to your script. You can define a colour (array of RGB) and/or a font-style (B, I, or BI) for each level (starting at 0). Results may depend on the PDF Reader you are using.

```
$this->bookmarkStyles = array(
    0 => array('color'=> array(0,64,128), 'style'=>'B'),
    1 => array('color'=> array(128,0,0), 'style'=>"),
    2 => array('color'=> array(0,128,0), 'style'=>'I'),
);
```

## Embedded SVG code

SVG Images can be embedded in your HTML code. This is formally part of the XHTML specification and is supported by IE9+ and most other browsers.



```
<svg xmlns="http://www.w3.org/2000/svg" version="1.1" viewBox="0 0 200 200" width="200" height="200">
  <circle cx="100" cy="50" r="40" stroke="black" stroke-width="1" fill="red" />
  <circle cx="130" cy="50" r="40" stroke="black" stroke-width="1" fill="blue" />
</svg>
```

## Improved CSS support

*border-radius* is now supported on fixed/absolute-positioned block elements.

*visibility* can be set on any block-style element e.g. DIV,P or images IMG as:  
visible|hidden|printonly|screenonly

*background-color* now supports rgba() and cmyka() alpha transparency formats in tables.

Color, underline and strike-through are now supported in table cells, including with rotated text (see example below of spread table)

*page-break-after: left|right|always* is supported on all block-style elements and tables

*text-transform: capitalize|uppercase|lowercase* is supported in table cells

### Zebra stripes in Tables

*:nth-child()* selector can be used in tables (on TR, TD or TH) to stripe rows or columns. Both the *an+b* and odd/even forms are supported e.g.

```
tr:nth-child(2n+1) { background-color: rgba(255,255,127,0.6); } or  
tr:nth-child(odd) { background-color: rgba(255,255,127,0.6); }
```

Row 1	This is data	167.00
Row 2	This is data p	444.53
Row 3	This is long data	14.00
Row 4	This is data	0.88
Row 5	Also data	144.00
Row 6	Also data	8089.00

**Note:** mPDF does NOT correctly apply specificity to all CSS. The following stylesheet:

```
table.zebra tbody tr:nth-child(2n+1) td { background-color: yellow; }  
table.zebra tbody td:nth-child(odd) { background-color: blue; }
```

should make every odd row yellow, and every odd column blue, but with the row/yellow overriding the column/blue. In mPDF the *td:nth-child(odd)* trumps the plain *td*, so the column colour wins out. You can force the effect you want by using:

```
table.zebra tbody tr:nth-child(2n+1) td:nth-child(1n+0) { background-color: yellow; }
```

The *:nth-child(1n+0)* selector just selects every *td* cell.

*border* can now be defined on in-line elements eg SPAN

- Cum sociis natoque penatibus et magnis dis parturient montes

**Note:** Remember that in mPDF, inside table cells, properties set on block elements are set when possible as in-line properties - so a P element inside a table with border set, will appear with a border around the text line as though it had been set on SPAN

## Shadows

*box-shadow* can be defined on any block-level element (P, DIV etc). It follows the CSS3 recommendation, but *inset* is not supported.

*text-shadow* can be defined on any element. It follows the CSS3 recommendation, but *blur* is not supported.

```
<span style="text-shadow: 0.03em 0.03em #666, -0.015em -0.015em white;">  
<div style="box-shadow: 0.3em 0.3em #888888;">
```

## Other Enhancements

### Column Totals (Tables)

{colsum} placed in the footer of a table will automatically display the sum of that column. If the table breaks across more than one page, the sum of the values on that page will be displayed. A number following the colsum e.g. {colsum2} will force that number of decimal places to be displayed.

Header Row	Header Row	Header Row
Row 1	This is data	167.00
Row 2	This is data p	444.53
Row 3	This is long data	14.00
Row 4	This is data	0.88
Row 5	Also data	144.00
Row 6	Also data	8089.00
Row 7	Also data	3.00
Row 8	Also data	23.00
	Column total: (using colsum2 in {})	<b>8885.41</b>

**Table caption goes here**

### Table CAPTION

The caption element for tables is partially supported (see example above).

```
<caption align="top|bottom" style="caption-side: top|bottom">
```

- The caption must come immediately after <table>.
- The CSS *caption-side* or HTML *align* attribute of top|bottom supported
- Left or right placement are not supported.
- The caption is handled as a separate block element brought outside the table, so:
  - CSS will not cascade correctly from the table
  - the width of the caption block is that of page or of the block element containing the table
  - text alignment will be to the page-width not the table width
  - if table page-break-after: always, the caption will follow the pagebreak

## Core fonts in non-core font document

Core fonts, which do not need to be embedded in a PDF, can now be included in a document which uses non-core fonts. The pseudo font-family names: `chelvetica`, `ctimes` and `ccourier` should be used.

```
<div style="font-family: chelvetica">
```

NB You could force mPDF to always use core fonts when Arial/Helvetica/Courier are specified, by editing `$this->fonttrans` in `config_fonts.php`:

```
$this->fonttrans = array(
    'arial' => 'chelvetica',
    'helvetica' => 'chelvetica',
    'timesnewroman' => 'ctimes',
    'times' => 'ctimes',
    'couriernew' => 'ccourier',
    'courier' => 'ccourier',
    ...
);
```

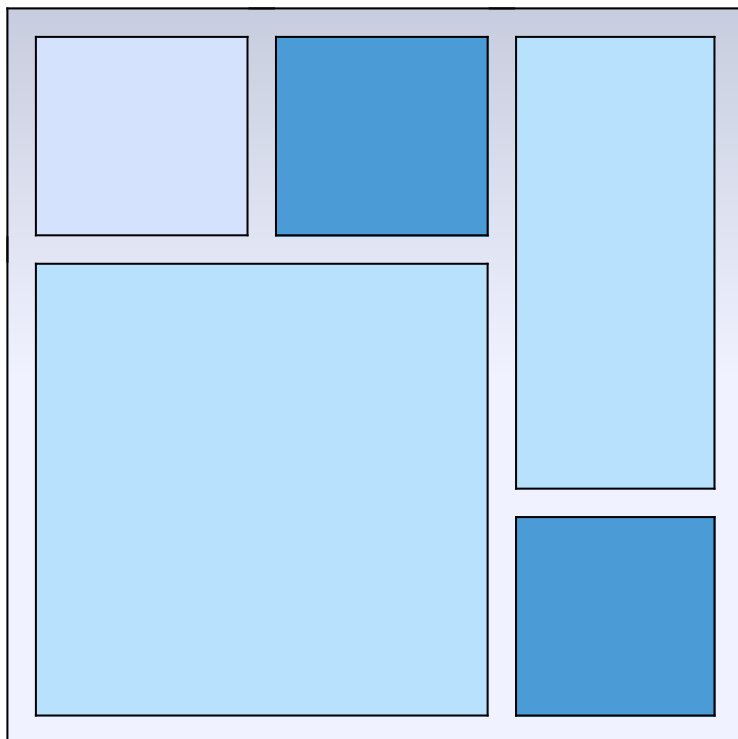
## Javascript in Forms

Javascript used in (active) forms has been altered to reflect the Adobe Acrobat specification for Javascript in PDF documents.

`textarea` and `input (text-types)` now accept javascript as: `onKeystroke`, `onValidate`, `onCalculate` and `onFormat`. `onChange` is deprecated but is not ignored; it works as though for `onCalculate`. (PS Select still accepts `onChange`)

## Overlapping Rows in Tables

Support for overlapping rowspans in tables has been improved (although probably not foolproof!)



# Circular Text

Circular Text can be included in a PDF document as a custom HTML tag (or a function)

- top-text and/or bottom-text can be specified
- Radius (r) and font-size (using CSS) are user-defined
- Width and height are calculated from radius and font-size
- Other CSS styles supported on Circular Text: border, margin, padding, color, background-color, font-family, font-size, font-weight, font-style, display, visibility, and opacity
- space-width should be specified as an integer defining the letter-spacing as a percentage of normal (default 120)
- char-width should be specified as an integer defining the width of each character as a percentage of normal (default 100)
- Circular Text is displayed as though an in-line element

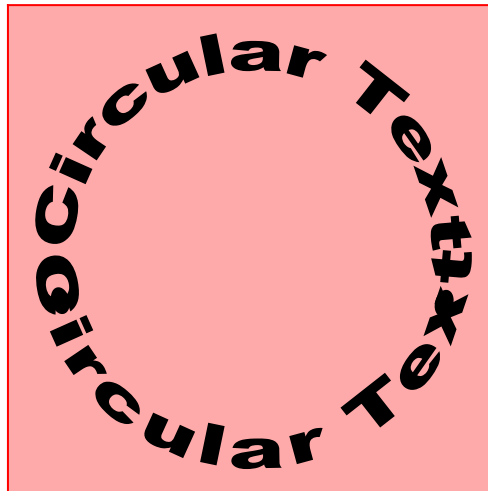
NB If `$mpdf->useKerning` is true then automatic kerning will be used on Circular Text.

```
<textcircle r="30mm" top-text="Circular Text Circular Text" style="color: blue; font-size: 34pt; font-style: italic" />

<textcircle r="30mm" space-width="120" char-width="150" top-text="&bull; Circular Text &bull;" bottom-text="Circular Text" style="background-color: #FFAAAA; border: 1px solid red; padding: 0.3em; margin: 0.3em; color: #000000; font-size: 21pt; font-weight: bold; font-family: Arial" />
```



Circular Text Circular Text



## Spread tables

Setting the CSS property "overflow: visible" on a table now has the effect of cancelling resizing, and allowing tables to split columns across multiple pages. The maximum width for a column (or group of columns set by colspan) is the page width. It is recommended to specify absolute values of width on each column (not percentages).

**Periodic Table (table caption)**

1	2	3	4	5	6	7	8	9	10	11	12	13
H												
Li	Be										B	
Na	Mg										Al	
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga
Rb	Sr	Y	Zr	Nb	Mo	Tc	<u>Ru</u>	Rh	Pd	Ag	Cd	In
Cs	Ba	<del>Lu</del>	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl
Fr	Ra											

14	15	16	17	18
				He
C	N	O	F	Ne
Si	P	S	Cl	Ar
Ge	As	Se	Br	Kr
Sn	Sb	Te	I	Xe
Pb	Bi	Po	At	Rn

### Limitations of Spread tables

Spread tables cannot be used with: keep-headings-with-table (`$mpdf->use_kwt`), table rotate, table page-break-inside:avoid, columns, CJK (chinese-japanese-korean) or RTL (right-to-left) languages. They will also cause problems with `$mpdf->forcePortraitHeaders` or `$mpdf->forcePortraitMargins`.  
Warning: If a table row is too tall to fit on a page, mPDF will crash with an error message. If the width settings within the table cause conflicts, it will override some of these settings.