

mPDF

ChangeLog

=====
mPDF v4.1.1 (21/12/2009)
=====

Error corrected in /makefont/makefonts.php file

=====
mPDF v4.1 (20/12/2009)
=====

MySQL support for embedded font subsets abandoned, and replaced with file-based.

Files no longer required

config_db.php
/unifont/RUNME.php
/unifont/*.ufm and /unifont/*.tla font files

MySQL Database no longer required

Files Updated

mpdf.php
/classes/tlasm.php
/makefont/makefonts.php

New files

/unifont/*.dat and /unifont/*.dat.php font files

Bug-fixes

-
- Image - If automatically resizing to fit maximum page size incorrectly subtracted margin-header
 - Annotation and textarea in same HTML chunk causes mPDF to crash (preg_replace textarea with /u modifier in AdjustHTML)
 - set magic_quotes_runtime error (\$mgr not \$mqr)
 - Table align did not reverse when using RTL document

Alteration: Image - if writing Image in fixedpos div position:absolute - to allow Image to be resized to full page size

=====
mPDF v4.0 (17/12/2009)
=====

Major additions

-
- Ability to embed font subsets (creating much smaller files)
 - Much improved support for Arabic languages
 - Support for Indic languages including consonant conjuncts
 - Support for Fixed position block elements
 - New utility to help create your own fonts
 - PNG alpha channel transparency supported
 - New utility to create smaller mpdf script with reduced functionality (less memory)
 - Multiple Barcode types supported

* For more details see the documentation manual: <http://mpdf1.com/manual/index.php?tid=410> *

Bug fixes (parsing CSS)

-
- <link href="" ... was not recognised if > 1 space between words
 - #Content p em { font-style:italic; } was applied to "#Content p"
 - @import url() embedded in a stylesheet file requires path fixed relative to stylesheet file

- background-image url() embedded in a stylesheet file requires path fixed relative to stylesheet file
- comment tags inside CSS <style> embedded in the HTML were removed
Now fixed so <style></style> works; is removed

Bug fixes (other)

- clear (CSS property for floating elements) caused properties for that element to reset to defaults
- width: auto caused collapse of border and padding on L & R of ordinary block elements
- text-indent not inherited correctly (including em and % values)
- named colour "steelblue" corrected RRGGBB hex code
- table cell widths in %: if width of table cells set to >=100%, and not all columns are set
This was fixed in 3.2 but led to problem where 2 cols: 1) 80% and 2) not set (see Table sizing test)
Now fixed again to work for both(?)
- parse PNG error fixed
- background-image not correctly positioned in HTMLFooter and HTMLHeader (Not fixed properly in 3.2!)
- fonts not supported with 0-9 in the name
- font list in GetCodepage() in htmltoolkit.php (now config_cp.php) containing space " " not recognised
- list number positioning
- list font size set in CSS for UL/OL not working for first level list
- table width (real value, not %) not working in nested table
- GIF file failed if PDF file not compressed
- list-style-type incorrectly inherited
- line-height inheritance in lists
- SetColumns added a new line - not required if at start of document/page
- footer_line_spacing did not work
- table cellPadding="" overwrote cell padding set on cell CSS
- could not turn off Default non-HTML footer LINE
- border specified as "em"
- default values set in mpdf.css overridden by inherited properties e.g. <div><h1>Here</h1></div> lost font-size for H1

=====
mPDF v3.2 (25/10/2009)
=====

Bug fixes

- Table cell widths in %: if width of table cells set to >=100%, and not all columns are set -> froze, because tries to produce a column of no width
- Output download file changed to allow compatibility with IE6
(http://mpdf.bpm1.com/forum/comments.php?DiscussionID=120&page=1#Item_4)
- Image error if relative path used on domain root (e.g. img src="image.png" and basepath is <http://www.yourdomain.com>) [attempted fix in 3.1 not working]
- Table: if font changed in cell, font was not restoring properly afterwards causing errors (restoreInlineProperties())
- Lists: list items containing
, font not restoring after bullet
- Graceful support for block elements inside list items e.g. <p>... (not supported, but tolerated)
- Index: Created dividing letters separately for Uppercase and lowercase
- Incorrectly changing input character set when encountering e.g. charset=iso-8859-1 in the text of the document
- Changed so only detects it if within <head>...</head>
- If Keep-with-table (i.e. H1-6 before table and use_kwt true), if pagebreak forced anyway, borders did not print on previous page
- Background-image used in HTML footer not appearing (correctly)
- RTL tables: nested tables will not automatically transpose L->R
- "Keep heading with table" - changed to allow <h1 style="..."> not just <h1>
- "Keep heading with table" - backgrounds (bgcolor, image or gradient) incorrectly handled - now removed
- Rotated table spread over more than 1 page caused enclosing block background colours to be rotated along with table
- CSS text-indent % now correctly supported (% of containing block width)
- CSS width em on a block element e.g. DIV now correctly supported
- calculating _tableheight, if remainingpage==0, get error (div by zero)
- Table moved to next page with page-break-inside=avoid, produced an enlarged table (font)
- RTL text-align override on BODY text was not working consistently
- Arab characters: Character ? (HEH) appearing in Final presentation form instead of Isolated
- Vertical position of background-image on whole page incorrect
- SetProtection can now be used with no permissions set (was not working unless at least one permission set)

Developers

Some more undefined indexes and variables declared (courtesy of DSmart
http://mpdf.bpm1.com/forum/comments.php?DiscussionID=117&page=1#Item_0)

Comment lines removed for < v3.0 to tidy up code

Enhancements

CSS style height now partially supported on block elements DIV, P, H1 etc. --IF--

- block is all on one page
- will extend the block but not shorten it
- will not force a pagebreak (max. at bottom of page)
- % is interpreted as % of printable page height (inside margins)

<TFOOT> now supported (placed at start as in HTML spec) displays at end of table, and repeats as a footer

Background-image and background-gradient now supported in TD and TH (works in all cases except: background-image is not rotated or positioned correctly if table is rotated)

NB Background images and background-gradients do not work if Columns are being used, or if \$use_kwt is TRUE (keep-with-table), or if page-break-inside:avoid is active.

Updated files

mpdf.php
htmltoolkit.php

=====
mPDF v3.1 (30/08/2009)
=====

Bug fixes

- Image error if relative path used on domain root (e.g. img src="image.png" and basepath is http://www.yourdomain.com was giving http://www.yourdomain.com//image.png) [3.1]
- Errors in parsing background CSS (background-repeat, background-position etc) [3.1]
- Textarea did not correctly convert width or height in units relating to font e.g. em [3.0beta_01]
- If page margin-bottom set to zero, SetHTMLfooter() crashes with "Division by zero" error [3.0beta_01]
- Table with header row and rowspan in tbody, not calculating maxrowheightcorrectly
- Prevent Index breaking column just after a dividing letter
- Select or input form field when text around it is justified had text in the form field justified
- TocBookMarkText needs to be htmlspecialchars-ed - decoded when entered inside <tocpagebreak>
- caused crash
- DisplayPreferences used as a variable name and a function: function renamed to SetDisplayPreferences()
- Image with src file not including a "." incorrectly parsed (e.g. http://www.domain.com/imagegenerator?params=23)

New Features

- var \$debug (true|false) default false; show or hide error reporting at output stage [3.1]
- var \$autoPageBreak (true|false) default true; allows overriding of automatic page breaks [3.0beta_02]
- <indexinsert /> HTML equivalent of CreateIndex() [was CreateReference()]
- 2nd attribute/parameter "xref" in IndexEntry() and <indexentry> - works like IndexEntrySee() as cross-reference entry
- function SetWatermarkText allows null parameters to be passed i.e. SetWatermarkText() - will clear the WatermarkText
- <watermarktext content="" alpha="" /> - HTML equivalent of SetWatermarkText()
- <watermarkimage src="" alpha="" position="" size="" /> - HTML equivalent of SetWatermarkImage()

Documentation

See Manual at <http://mpdf.bpm1.com/manual/> for more information - especially:

- User's Guide>>What Else Can I Do?>>Backgrounds & Borders
- User's Guide>>What Else Can I Do?>>Floating blocks

Files updated:

mpdf.php
htmltoolkit.php
graph.php

Developers only

mPDF<=3.1 generated a large number of warning "Notices" if run with full error_reporting on, due to array indexes not being initiated e.g.

```
$arr = array();
```

```
...  
if ($arr['index'] == 5 ) {...}
```

To prevent this, lines were added at the start of the mPDF.php script to turn error notices OFF. In a move towards making mPDF able to run with full error_reporting on, a large amount of the script has been altered

e.g. the line above would be changed to:

```
if (isset($arr['index'] && $arr['index'] == 5 ) {...}
```

Although I have tested this with a number of examples, it is almost certainly not complete. Therefore the error_reporting for Notices is still turned off in mPDF 3.1

If you care to test it, please uncomment line 43 (//error_reporting(E_ALL);) and report any warning notices that you get.

NB This has added about 40kB to the script size.

```
=====  
mPDF v3.0beta (14/06/2009)  
=====
```

New Features

- CSS "float" partially supported (as well as clear:left|right|both)
- CSS "background-image" "background-position" "background-repeat" "background-color" "background" supported for block-level elements
- CSS background-color and background-image for <body > element added: this covers the whole page i.e. not just inside the "margins"
- CSS background-color and background-image can be defined for CSS @page{}
- Background gradients (linear or radial) can be defined using a custom CSS style property
- Border radius can be defined to give rounded edges to block elements (uses draft CSS3 spec.)
- page number can be reset to any value during the document (in AddPage() <pagebreak> etc.)
- PNG images: Interlaced and alpha-channel-set PNG images now supported
- internal links supported in Indexes (parameter added to CreateIndex()/CreateReference(): \$useLinking=true;)
- HTML Headers and footers now support hyperlinks
- improved handling of
, block elements, and text lines inside table - approximates better to browser handling
- borders of block-level elements & table cell borders supported (partially) in columns
- optional error reporting for problems with Images (\$showImageErrors)
- ToC will word-wrap long entries
- internal links (Bookmarks, IndexEntry and ToCEntry) rewritten to give more accurate positioning (when used as <tag>)
- autofont algorithm improved for CJK languages
- define text before and after page numbers (\$pagenumPrefix; \$pagenumSuffix; \$nbpPrefix; \$nbpSuffix;)
- Additional color names supported - full list from SVG1.0

Bug fixes

- Column width not resetting after an automatic pagebreak, or after setting HTMLheader
- using AutoFont unnecessarily changed htmlspecialchars to code causing errors
- Lists inside a table - incorrectly calculating table cell height
- CJK - 4-byte utf-8 chars not displaying properly (includes HKCS characters)
- mailto: links incorrectly handled
- TOCpagebreak() - usePaging default clarified: true unless specified as '', 0, '0' or false; (null ->true)
- <tocpagebreak> (as html tag) with no "name" defined, used at start of page, added a further blank page(s)
- Lists - inaccurate calculation of space required for numbers in certain circumstances
- Generated images (.php) only working if cURL enabled - (fixed, but requires allow_url_fopen if remote file)
- flag added to turn off error reporting when buffering used (\$allow_output_buffering = false;)
- RTL text in Bookmark, Title, Author, Creator, Keywords, and Subject was reversed - Adobe Reader 9 now correctly handles RTL text (which Reader 8 did not)
- TOC - if not using ODD/EVEN paging, did not add extra page and messed up
- Rotated table which did not fit on remaining page resized to bigger than default
- HR of width less than 100% - text continued on line after it
- HR alignment not working (fixed so both CSS text-align and margin: 0 0 0 auto etc work)
- HR in table did not correctly re-size when necessary
- characters in symbols/zapfdingbats which in non-utf-8 mode are represented as chr(173) incorrectly handled as soft-hyphens

- (bug introduced 2.5 with soft-hyphens - affects symbols ↑ arrow-up and Zapfdingbats ® encircled 2)
- Internal links (anchors) - Annotation/Bookmarks etc. incorrectly positioned when page orientation changed
 - ToC - when using multiple ToCs, internal links were not correctly adjusted
 - anchor (a name="") used inside a table was incorrectly positioned at the end of table
 - Tables: cell height calculated incorrectly when BR used
 - Table rotated with "page-break-inside:avoid" not kept on one page
 - Table rotated and split over > 1 page - vertical alignment inaccurate
 - Headers/Footers (non_html) when no style set caused errors
 - Table: breaking page when using rowspan error (line 17142)
 - ToC: If no indent defined in HTML tag <tocpagebreak> or defined as 0 gave error

Note

In mPDF 3.0 the following sections of code have been significantly rewritten:

- painting of borders and background colours for block-elements
- table of contents
- Index
- vertical justification in columns (uses scaling to stretch vertically)

NB changed htmltoolkit AdjustHTML - does not now remove
 before </div>
Warning - may display differently in normal text as well as tables

Files updated:

mpdf.php
htmltoolkit.php

Developers only

-
- Background-color handling in CSS changed so only inherited/cascaded when Columns active or Keep-block-together
 - otherwise would overwrite background image with inherited color
 - all %.2f used in sprintf() changed to %.3f in htmltoolkit.php and mpdf.php to increase accuracy of div border lines in columns etc.
 - variable \$use_embeddedfonts_1252 renamed to \$useOnlyCoreFonts as more precise: deprecated but still supported.
 - this version included quite a bit of tidying up/future-proofing some code:
 - \$var{0} changed to substr(\$var,0,1) etc. (due to go in PHP6)
 - ereg_ changed to preg_ (deprecated in PHP5.3) - (NB mainly in htmltoolkit.php)

=====
mPDF v2.5 (01/05/2009)
=====

New Features

-
- Automatic Hyphenation added, and support for soft-hyphens
 - Encryption works now for CJK language documents
 - Improved text justification
 - Support for 'generated' images e.g. "../onthe-fly-image.php"

Bug fixes

-
- Tables: cell height did not reduce if font-size used was smaller than table default
 - Columns: if setcolumns() to the same number already active - did not print out last bit of previous columns
 - Page-break in the middle of a block caused incorrect margin and padding on next lines until end of block (\$cMargin reset to 0 in AddPage)
 - <HR> in table cell was printing in incorrect position (bug introduced in mPDF 2.4)
 - Justification
 - if only one word on line, did not respect maximum character spacing
 - last character of line incorrectly had character spacing applied
 - Space at the end of last line of a Right Justify block - e.g. "end. </p>" now correctly ignored
 - incorrectly treated as a character when justifying text with word/char spacing
 - CJK punctuation (.,) added as 'orphans' to keep at end of line
 - PNG files - was still buggy reading larger PNG files (due to fread)

Files updated:

mpdf.php

htmltoolkit.php
CJKdata.php
/patterns/.. (new files)

Developers only

Variables renamed as more accurate or appropriate:

- var \$isunicode renamed as \$is_MB
- var \$usingembeddedfonts renamed as \$usingCoreFont

CJK changed to act internally as UTF-8 encoded

- (NB CJK Half-widths not supported from 2.5+ i.e. big5-hw gb-hw)

=====

mPDF v2.4 (23/04/2009)

=====

Files updated

mpdf.php
htmltoolkit.php
mpdfi/mpdfi.php

New files

graph.php

Graphs - Requires new folder: path_to_mpdf/graph_cache/ (must be writeable)

New features

Annotations improved so they appear as a pop-up

Re-use Document Templates (cf. RestartDocTemplate() in manual)

Limited support for CSS float property on an IMG element allowing text wrapping e.g. (cf. Images in manual)

Utility function PreparePreText() allows output of a text file which may include <pre>

Automatic generation of graphs from data in tables (requires integration with JGraph) (cf. Graphs in manual)

Other Changes

IMPORTANT - User rights removed as not working with newer version of Adobe Reader 9 (affects Active forms and ability for users to modify annotations)

Corrects text alignment when using 17 or 17 in (non-HTML) headers/footers

Sets default timezone if not already set (at top of mpdf.php) to prevent E_STRICT ERROR message

Suppresses E_NOTICE error reporting (at top of mpdf.php)

Error capture in Output() to avoid PDF header being sent when error messages generated

A function str_ireplace added to htmltoolkit to allow PHP4 to function

Bug fixes

WMF images incorrectly positioned when in-line

PNG images > 8kB failed to load - (fix in 2.3 didn't work - fixed properly this time)

Annotations containing a new line (\n) causing an error

Evaluation of <pre> text: "<code>[TAB] " evaluated incorrect number of spaces to follow to align tabs, because < was calculated as 4 chars (<);

=====

mPDF v2.3 (22/03/2009)

=====

New Features

- Optionally detect language and when to use special fonts i.e. RTL (arabic), CJK (chinese), Thai (see SetAutoFont() etc.)

- Supports HTML attribute "lang" in all tags and uses special fonts when required (see \$useLang)

- Joins Arabic and Farsi/Persian text into presentation forms

- Import another PDF file and use as templates in your document (see UseTemplate() and mPDFI in the manual.)

- Replace specified text strings in an existing PDF file (see OverWrite() etc.)

- More than one Table of Contents can be used in a document (see tocpagebreak etc.)

- Restore properties of open HTML block elements after a page break (variable \$restoreBlockPagebreaks or new tag <formfeed>)

- <annotation> <bookmark> <indexentry> <tocentry> can now accept characters <'& as htmlentities - htmlspecialchars(..., ENT_QUOTES)

- <annotation> can now accept "\n" for new line

- support for opacity (CSS3 property) for images
- specify the number of spaces to substitute for TAB when parsing <pre> tags
- greater control over margins and display when changing page orientation during document (see \$forcePortraitMargins and \$displayDefaultOrientation)

Bug fixes

Fonts in CSS - Not parsing font-family: Trebuchet MS; correctly as trebuchet
 Fonts in CSS - CSS font-family: [unknown]; setting first \$available_unifont rather than ignoring
 Images - not displaying on IIS platform
 Images - .wmf not displaying if (allow_url_fopen) not set
 Table borders - in defaultCSS, 'MARGIN-COLLAPSE'=> collapse not quoted therefore not working
 Line-break inside table - printing a blank background across page rather than just going down a line
 Form fields inside tables - will now resize if the table is autosized (shrunk)
 <pre> containing a '<' was changed to '<'
 Tabs inside <pre> were all changed to 8 spaces, not the remainder following a string
 Header on first page was inset by 1mm left and right (\$cMarginL and \$cMarginR not set to zero)
 Table nested inside a cell with colspan > 1 was incorrectly handled
 PNG file crashed (?incorrectly defined PNG file) [adapted _parsepng to account for unexpected]
 Table or Cell - if font-size not recognised, mPDF set font-size to zero
 Font-sizes - [xx-small|x-small|small|medium|large|x-large|xx-large] were not recognised in tables

=====
 mPDF v2.2.1 (17/02/2009)
 =====

Bug fix - (bug introduced in 2.2)
 Table - header row did not return to top of page when repeating across pages.

=====
 mPDF v2.2 (15/02/2009)
 =====

Updated files from mPDF 2.1
 mpdf.php
 htmltoolkit.php
 mpdf.css (new)

=====
 New files from mPDF <2.0 (only required for EAN Barcodes)
 font/ocrb.xx (several)
 unifont/ocrb.xx (several)
 IMPORTANT - you need to make sure the ocrb font is added to the config.php file
 - add 'ocrb' to the end of 3 arrays: \$this->available_fonts \$this->available_unifonts and
 \$this->mono_fonts
 =====

Summary of changes

- external stylesheet file (mpdf.css) is used to configure default values (\$useDefaultCSS2 and \$defaultCSS2 are no longer used)
- special comment tags to hide mPDF tags from browsers: ...
- AddColumn() function added (equivalent to <columnbreak>)
- annotations - pop-up messages the reader can move or delete (if you set permissions)
- support for WMF images as well as GIF, JPG, PNG
- watermark image can be set instead of, or as well as text
- nested tables can include other content
- improved control over table layout
- margin: auto now supported for table and block elements

A number of methods and variables have been renamed or reCapitalised for consistency.
 Changes should be backwards compatible.
 All user methods start with a Capital, all user-defined variables start with lowercase.

Affected:

Reference() -> IndexEntry()
 CreateReference() -> CreateIndex()
 \$TopicIsUnvalidated -> \$showWatermark
 setUnvalidatedText() -> SetWatermarkText()

PHP appears at present to be case-insensitive for function/method names

All the following functions have been renamed in the script with a capital first letter:

setUserRights()
 setBasePath()
 setAnchor2Bookmark()
 setHeader()
 setFooter()
 defHeaderByName()
 defFooterByName()

```
setHeaderByName()
setFooterByName()
setHTMLHeader()
setHTMLFooter()
defHTMLHeaderByName()
defHTMLFooterByName()
setHTMLHeaderByName()
setHTMLFooterByName()
shaded_box()
writeBarcode()
```

Variable names changed to lowercase first letter:
(Variables are case-sensitive therefore aliases have been set up)

```
Anchor2Bookmark
BiDirectional
KeepColumns
AliasNbPg
AliasNbPgGp
```

```
=====
Bug fixes
=====
```

```
Columns - $keepColumns=true was incorrectly calculating the place to continue printing after 1 and
half columns (of 3)
Table cell height - incorrectly setting table cell height when cell contained a line of text of large
size which wrapped to more than one line
HR in Table cell - if table cell contains only HR (and column otherwise empty), HR was printed outside
cell
HR in Table cell - if table cell ended with a HR, height was one line too much
Table page-break-inside:avoid - caused mPDF into permanent loop in some circumstances
Paging - Total Pages/Group 17 and {nbgp} didn't work in CJK
CSS - Border size thin, medium and thick were only recognised in lowercase
Table-header - rowspan not correctly output in THEAD
Default CSS - table empty-cell:hide changed to show (CSS specification)
```

```
=====
mPDF v2.1 (24/01/2009)
=====
```

New Features in Version 2.1

- CSS support improved generally (especially for cascading CSS, lists)
- TableHeader changed to allow multiple rows in THEAD

CSS changes

- display: none (block elements only - not lists or tables, nor HR)
- width (TD/TH)
- list-style-type (will also recognise the list-style-type from list-style) (OL/UL)
recognised values:
disc|circle|square|decimal|lower-roman|upper-roman|lower-latin|upper-latin|lower-alpha|upper-alpha|none
- CSS support for : font-family, font-size, font-style, font-weight, color, background-color, text-decoration, text-transform, and list-style-type (will also recognise the list-style-type from list-style)
- table cell borders - CSS rules have been adapted slightly - if a coloured/black line conflicts with a white line, and is the same width, coloured/black will overwrite even if Bottom or Right

Numbered Lists

- Variables set at the top of mpdf.php can be set to change:
- text alignment of numbers in numbered lists (default Right)
var \$list_align_style = 'R';
 - content to follow a numbered list marker e.g. '.' gives 1. or IV. whereas ')' gives 1) or a
var \$list_number_suffix = '.';
- (These can be altered at run time, but are not changeable through stylesheets or in-line style)

Writing broken segments of HTML

```
2 new parameters have been added to WriteHTML()
function WriteHTML($html,$sub=0,$init=true,$close=true) {
$close - Leaves buffers/variables etc. in current state, so that it can continue to write the HTML
where it leaves off
$init - Clears and resets buffers/variables
(N.B. You must end with a WriteHTML that calls $close=true)
```


Example:

```
$mpdf->WriteHTML('<p>This is the beginning...</p>', 2, true, false);  
$mpdf->WriteHTML('...this is the middle...</p>', 2, false, false);  
$mpdf->WriteHTML('...and this is the end</p>', 2, false, true);
```

Rotated text in table cells

NB This UPDATE will change expected output from previous versions*****

Prior to v2.1 any cell set to rotate text anticlockwise was forced to vertical align = bottom. This has been changed so that it only overrides when the rotate angle is between 45 and 89 degrees: text rotated exactly 90 degrees will respect the set value for vertical-align.

=====

Bug fixes

=====

- List - list starting after "<div>Then some text not in a block element" incorrectly output
- Tables - if cell font-size set smaller than default for the table, does not shrink the cell height
- Columns (tables) - columns breaking across rows e.g. in the middle of a table cell
- Tables - if table width set to e.g. 100% but cols are less, was not making up to set width
- Watermark - was not printing if using HTMLFooter
- Lists - not aligning numbering correctly if different font sizes used for bullet & text etc.
- Lists - indent of text did not correctly allow for Maximum number in list
- Table does not always move correctly to a new page
- Table cell incorrectly calculated height causing text to overflow cell when printed
- Table borders in columns not being correctly handled (bug since 2.0 introduced a buffer to save the borders and print at the end of the table - fixed so does not use buffer if in columns - potentially does not deal with conflicting borders as well, but works in columns)
- Table cell width if set as a percent was being downsized when autosizing table
- Table CSS was buggy - improved
- SetBasePath (when fetching remote website) - now handles string with query string on it e.g. <http://www.domain.com/index.php?tid=42>
- Table cells with Rotated text - text not positioned correctly
- Page number totals not working in utf-8 mode

=====

Code efficiency improvements

=====

- BIG speed improvement (compared with 2.0) with tables (especially large tables)
- considerable increase in speed if writing long HTML segments to mPDF
- speed improvement for tables (may be very significant if some cells have a lot of text in them causing uneven column widths)

NB To speed up program more, consider setting \$mpdf->useSubstitutions=false; if you do not use any characters outside the codepage selected

=====

Keep-with-table

=====

(This was introduced in v2.0 but I forgot to document it)
If you set \$this->use kwt = true;
All H1-H6 elements will try to keep with a table that follows immediately afterwards - (this is done in htmltoolkit, by adding an attribute KEEP-WITH-TABLE)
See Known Issues re: Using kwt inside a div with border/background (doesn't work)

=====

mPDF v2.0 (07/12/2008)

=====

Main New Features in Version 2.0

- nested tables are supported
- supports both models of table border: separate and collapsed
- improved parsing of CSS stylesheets, and better handling of styles throughout
- additional recognised CSS styles
- page orientation, size, and margins can be changed within the document, using PHP script or custom HTML
- some limited support for @page CSS to define page-box areas, with crop/cross marks for printing
- improved control over headers and footers (including HTML headers/footers)
- improved presentation of Form elements including image-type input fields
- generates an EAN barcode suitable for a book/printed publication
- active forms can be generated - EXPERIMENTAL at present
- change document permissions to allow the user to make annotations - EXPERIMENTAL at present

NB Lines are commented in mPDF script as changes for mPDF 1.4 - this became v2.0

=====
UPGRADING
=====

IMPORTANT - Before Upgrading: Please note that some of the changes will cause mPDF 2.0 to render the pages differently from earlier versions i.e. it is not totally backwards compatible. Read the notes on Backward compatibility before deleting your original set-up.

To upgrade from v<=1.3 to v2.0 you only need to copy and overwrite the following 2 files:
mpdf.php
htmltoolkit.php

Plus (optionally) if you want to use the EAN Barcode function, you will also need:
font/ocrb.xx (several)
unifont/ocrb.xx (several)

IMPORTANT - you need to make sure the ocrb font is added to the config.php file
- add 'ocrb' to the end of 3 arrays: \$this->available_fonts \$this->available_unifonts and \$this->mono_fonts

=====
Bug Fixes
=====

<columns column-count="0"> did not turn off columns - Fixed

Margins as % - e.g. margin-right: 50% set from CSS incorrectly applied 50% of the fontsize
(Fixed - adding parameter to fn.ConvertSize in htmltoolkit.php and in calls to that function)

DIV Width - e.g. div style="width: 50% was not working
(Fixed - fn.SetCSS and OpenTag('DIV'))

CSS values as Zero - p { margin: 0; } did not work in stylesheet unless 0 had a unit

Multiple Non-breaking spaces collapsed - e.g. "1 2" - was contracted to "1 2"
(Fixed - fn.adjustHTML in htmltoolkit.php)

Table cell too narrow causing incorrect printing - If two characters are too wide to print (only likely within a table cf. example tables - Periodic table) the first character was not printed, just a new line
(Fixed in fn.WriteFlowingBlock)

Font size by inline style for form elements <input> <textarea> gave wrong size when using a relative size 0.9em
(Fixed - fn.ConvertSize in htmltoolkit.php)

Creation Date not correctly showing
(Fixed - did not need to convert to UTF16)

New block element started at end of page - block borders not painted
(Fixed - fn.AddPage)

DL did not close block correctly
(Fixed - mistype in fn.CloseTag)

Transparent not recognised as color/background-color
(Attempted a fix by setting to ignore it! - fn.convertcolor in htmltoolkit.php)

Zero (0) not displaying if only thing in table cell or tags e.g. <td>0</td> <p>0</p>
(Fixed - fn.WriteHTML)

Page Headers/Footers - Simple Headers or Footers defined as e.g. '|{PAGENO}|' were not split into 3 components, but output |6| in the outer margin.
(Fixed)

Could not copy from a completed PDF doc to clipboard when using a TrueTypeUnicode font
(Fixed - fn.puttruetypeunicode - added CIDToGIDMap)

Creating an Index (confusing called CreateReference in mPDF) based on only 1 column (i.e. columns off) caused it to print FAIL
(Fixed - fn.CreateReference())

Table of Contents - If a ToC entry reached other side of page a warning notice was produced
Fixed - printing is now suppressed and it is moved down a line (but not when using rtl)

"Keep Block Together" (i.e. page-break-inside: avoid for a block element)
If this property causes some text to be moved to the next page, internal link targets (i.e.) were incremented pagenumber by +1 - as this used the calculated document page number,

didn't work if using e.g. roman numerals

(Fixed)

NB This is now ALWAYS disabled when it meets a table - can use page-break-inside:avoid for the table

@import url(style.css) without quotation marks "" was not picked up, although it is valid HTML

(Fixed - fn.ReadCSS())

Reading CSS from external style sheets included all media

Now set by default to ignore media="aural|braille"

Allows media="print" but can exclude by: \$mpdf->disablePrintCSS = true; (default = false)

Works on both <link... and <style media="print">@import...

See the web page example - this stops the CSS stylesheets specifically marked for "print" to be ignored

Table borders (in collapsed model) incorrectly calculated which border had dominance (mPDF <=1.3 determined the overriding border by its color)

(Fixed to follow CSS 2.1 specifications: width >> CSS dominance (cell>table) >> T & L > B & R)

In the border-collapse=collapse mode, the following rules determine which border style "wins" in case of a conflict:

1. Borders with the 'border-style' of 'hidden' take precedence over all other conflicting borders.
2. narrow borders are discarded in favor of wider ones.
3. styles are preferred in this order: 'double', 'solid', 'dashed', 'dotted', 'ridge', 'outset', 'groove', and the lowest: 'inset'.
4. If border styles differ only in color, then a style set on a cell wins over one on a table.
5. When two elements of the same type conflict, then the one further to the left or top wins out.

Parsing CSS, a missed level would not be picked up i.e. CSS = div.refstr1 p {...} would not pick up:

<div class="refstr1"> <div class="another"> <p>...

(Fixed - fn.MergeCSS by carrying everything forwards)

Note: I removed - &&

!\$this->blk[\$this->blklvl-1]['cascadeCSS']['CLASS>>'].\$attr['CLASS']['depth']

Not sure why I put it there in the first place!

In a later bit of function - && \$this->blk[\$this->blklvl-1]['cascadeCSS'][\$tag]['depth']>1

Haven't removed this, but is it needed???

CSS inline style set in TD/TH (font-size, color, weight, font-family or italic) didn't turn off at the end of the cell (and also formatted the top left cell)

(Fixed)

CSS properties color, font-weight, and font-style set on a table were not inherited by the table cells

(Fixed - var \$base_table_properties)

When rotating a table, the text following was positioned incorrectly

(Fixed - fn.printtablebuffer)

When rotating a table, links were not adjusted in position e.g. ...

(Fixed - fn.printtablebuffer)

If a larger border-thickness was set for a cell in the middle of a table, cells before that inaccurately calculated the cell width needed

(Fixed)

Text in a table cell which was not in a block and followed a list, there was no line break e.g.

Text following</td>

(Fixed - var \$listjustfinished;)

Setting the default font-family from the body tag using in-line CSS did not work e.g. <body

style="font-family:mono"> (did work in external stylesheets)

(Fixed - fn.WriteHTML)

<link href="..." rel="stylesheet"> was not recognised (because the href comes before the rel)

(Fixed)

Read linked CSS stylesheet OR @import stylesheet - now includes both.

<tag class="class1 class2"> did not set either class1 or class2. Fixed so that it will now pick out class1 (better than nothing!)

CSS line-height as % - interpreted 120% as 120 (x the font-size) - Fixed so it now accepts % and numbers

Setting the basepath (used for relative links/external stylesheets etc) with \$mpdf->setBasePath() was generally buggy!

Now allows a domain e.g. \$mpdf->setBasePath("http://www.domain.com"); (previously needed slash on end)

(Fixed - hopefully!)

If you were repeatedly calling mPDF in a loop to produce more than one PDF file, it would crash with error: "You have restricted the number of available fonts to 0". Fixed by editing line 751

```
require(_MPDF_PATH.'mpdf_config.php'); to require(...
```

```
=====  
Improvements(?)  
=====
```

Unsupported image files - mPDF died with an error message if image files didn't meet expected format e.g. an interlaced PNG file
Changed so images are now replaced by the NOIMG image.

Footer margin (HTML and normal footers) now determines the lowest point that is printed (rather than the place to start printing the footer)
NB IMPORTANT CHANGE - not backwards compatible

Tabs in <pre> or <textarea> are now replaced by 8 spaces rather than 6 (consistent with <http://www.w3.org/TR/1998/REC-html40-19980424/struct/text.html#edef-PRE>) [fn. AdjustHTML() in htmltoolkit.php]

To insert the total number of pages in the document anywhere in the doc, just use '17'
The line: `$mpdf->AliasNbPages();`
has now been uncommented allowing 17 to be used
NB This will always give the total no. of pages in the whole document regardless of any changes you have made to page numbering.
You can change the default placeholder '17' to any string using: `$mpdf->AliasNbPages(['**myChars**'])`;

`$mpdf->AliasNbPageGroups();` default="17"
Can be used to set the total number of pages in the current group i.e. between where page numbering is reset

CSS border correctly fixes "solid 3mm #000000" i.e. (style width color) - (not a bug, but this is often incorrectly specified)

Can now print div background behind a rotated table.

You can keep columns as they are i.e. 1st column will finish page then start on second, by setting `$mpdf->KeepColumns = true`;

```
Image constrain  
=====
```

Image size is constrained to current margins and page position. Extra parameter added to end of fn. allows you to override this.
`$mpdf->Image('files/images/frontcover.jpg',0,0,210,297,'jpg','','true, false);` // e.g. the last "false" allows a full page picture
Useful for e.g. a cover page for your document

```
Cumulative CSS  
=====
```

In version <=1.3, if you call:
`$mpdf->WriteHTML($stylesheet,1);`
`$mpdf->WriteHTML($html);` // this one cleared the array `$this->cascadeCSS`; containing cascaded CSS information from stylesheets
You were meant to call:
`$mpdf->WriteHTML($html,2);` // which doesn't re-parse the CSS information
v2.0 changed so that
`$mpdf->WriteHTML($html);` no longer clears the array `$this->cascadeCSS` and so can be used repeatedly;

```
=====  
Additional CSSstyles & HTML attributes supported  
=====
```

<BODY> - font-style, font-weight, color

 - html attributes width="" and height=""

<TABLE|TD|TH> - border: 'thin' 'medium' and 'thick' are now converted to 1px, 3px and 5px

<TABLE|TD|TH> - border now respects all types - e.g. 'double', 'solid', 'dashed', 'dotted', 'ridge', 'outset', 'groove', and 'inset'

(NB mPDF only supports a full declaration of border e.g. "border: thin double #000000;")

<TD|TH> - CSS style="white-space: nowrap" and HTML attribute nowrap="nowrap"

<TABLE> - page-break-inside: avoid

<TABLE> - border-collapse: separate|collapse

<TABLE> - border-spacing: 2px 2px; (horizontal/vertical) or just one figure (both) NB same as cellSpacing

<TABLE> - empty-cells: hide|show (border-collapse:separate only)

<TABLE> - margin-left and margin-right (previously only supported top and bottom)

<TABLE> - padding: (this was incorrectly used for TD/TH before) (border-collapse:separate only)

<TD|TH> - padding:

<TABLE|TD|TH> - inline style "background" works (with a color only) the same as "background-color"

NB Table page-break-inside, autosize values and rotate are only respected for that set on first level table of nested tables

<TABLE> - cellSpacing and cellPadding HTML attributes:

NB cellSpacing is the same as CSS style "border-spacing"

NB cellPadding is a <table> attribute, but sets the cell padding - not table padding

<TABLE> - align="..." now works with a rotated table - but sets the alignment ignoring the rotation i.e. align=right sets the table to the right side of the page (looking as though it is bottom-aligned)

<BLOCK ELEMENTS> - page-break-before: always|left|right - NB any surrounding block tags will be closed before the new page is inserted.

@page - see notes on Paged Media

=====
Unsupported HTML attribute

=====
<TD border="1"> - not valid HTML - no longer supported

=====
Additions

=====
"Keep-with-table"

\$mpdf->use_kwt = true; // default=false

If set to true, will force any H1-H6 header immediately preceding a table to be kept together with the table

- automatically sets the table to fit on one page (i.e. page-break-inside=avoid) if it is a rotated table

- ignored if: Columns on, Keep-block-together active (page-break-inside=avoid for surrounding BLOCK), active Forms

=====
Notes

NB Not a change - but note you can use this to allow you to feed html code encoded in other than utf-8:

\$mpdf->allow_charset_conversion=true;

\$mpdf->charset_in='windows-1252'; (needs suitable codes for iconv i.e. windows-1252 not win-1252)

Fixing Optional tags

=====
php.net website has illegal nesting of <dt>.. <dd> .. </dd> .. </dt> and <p>.. <div> .. </div> .. </p>

The example which parses the php.net webpage will not show correctly unless you change:

\$mpdf->allow_html_optional_endtags=false;

Trying to fix incorrect (X)HTML with \$mpdf->allow_html_optional_endtags==true cancels the P when it meets a DIV etc.

EAN barcode

=====
An EAN barcode can be generated

function writeBarcode(\$code, \$showisbn=1, \$x='', \$y='', \$size=1, \$border=0, \$paddingL=1, \$paddingR=1, \$paddingT=2, \$paddingB=2) {

It accepts 12 or 13 digits with or without - hyphens as \$code e.g.

\$mpdf->writeBarcode('978-1234-567-890', 1, \$mpdf->x, \$mpdf->y);

NB - IMPORTANT***

A new font - OCR-B font/unifont is required, and needs to be added to the config.php file

cf. http://www.gsluk.org/downloads/bar_code/Bar_coding_getting_it_right.pdf

Barcode size must be between 0.8 and 2.0 (80% to 200%)

CMYK Colors

=====
Functions - SetDrawColor, SetTextColor and SetFillColor all now take an optional 4th parameter.

If defined this will interpret the input as CMYK color i.e.

SetDrawColor(15,82,0,10) // NB all values out of 100 - not 255 as for RGB

htmltoolkit.php fn.ConvertColor() - now interprets custom color definition: cmyk(15,82,0,10)

like rgb(r,g,b) except values out of 100

Intended to be used for calling the functions separately - BUT works in a limited way with CSS - does

not get reset or inherited correctly
<p style="color:cmymk(215,31,15,10)"> does work

DEFAULT CSS

=====
\$defaultCSS has been updated to reflect better the standard HTML default e.g. using serif, table borders separate, cell vertical-align top
To keep mPDF 1.3 (my favourites) I have introduced:
\$mpdf->useDefaultCSS2 = true;

Permissions - forms and Annotations - Experimental!

=====
You can set the Permissions for the PDF file to allow the user to make Comments (annotations)
\$mpdf->setUserRights(\$enabled=true[default]|false, \$annots="/Create/Delete/Modify/Copy/Import/Export",
\$form="/Add/Delete/FillIn/Import/Export/SubmitStandalone/SpawnTemplate",
\$signature="/Modify")

If you encrypt the file, make sure the permissions match e.g.:
\$mpdf->setUserRights();
\$mpdf->SetProtection(array('print','annots'),'yourPassword','myPassword');

To allow this, changed the PDF-file version to %PDF1.5 (NB Probably needs PDF version > 1.5 but can't test for this...)

NOTE: If you output the PDF file straight to the browser, it will only allow annotations after you save the document

Active Forms - Experimental!

=====
At present , using active forms will prevent any internal and external links - that appear before active forms(?) - from working (why?)
May need to save form for proper use - see example, when scroll forwards and back, the form disappears??
Need to set Userrights (see above), and \$mpdf->useActiveForms=true;
For Output options, see separate notes.

Internal Programming changes

=====
NB fn. tablerotate in htmltoolkit no longer used; now uses a 'transform' to shift the whole block of PDF code

Graphics State

=====
ExtGState does not need to be redefined - e.g. if a watermark added on every page, turning on and off alpha/transparency
Unnecessary file size.
function AddExtGState() edited to check if graphics state already exists before adding new one

ASCII-proof code

=====
"»«-»" used as a special identifier in the program changed to "\xbb\xa4\xac" to make the mpdf.php script file immune from someone saving it as a utf-8 encoded file

Images

=====
1) When copying remote images locally - incorrectly used "unset" now changed to "unlink"
2) When parsing image files - was using CURL for any image src="http://... - even if this was on the local server - edited so it only uses CURL if necessary/appropriate.
NB Handling images was updated in v1.3 because my ISP changed allow_url_fopen to false
Fixed so mPDF tests if the file is available as a local call e.g. getimage('images/test.jpg') even if it is defined as a full URI e.g. http://www.mydomain.com/images/test.jpg as this is quicker(?), and permitted even if allow_url_fopen is false.
If not available as a local file (and allow_url_fopen is set) mPDF tries to use fopen/file_get_contents using an http wrapper;
Else, if CURL is available and allow_url_fopen is false: then tries using CURL.
(Clear as mud???)

=====

Page Numbering

Program changes:

fn. startPageNums() replaced with blank function
fn. stopPageNums() - deleted
fn. numPageNo() - deleted (all it did was return this->page anyway).
var \$_numbering - deleted
var \$_numberingFooter - deleted
var \$_numPageNum - deleted

NEW

New: fn. AddPages() (as for AddPage but with type=NEXT-ODD or NEXT-EVEN see below)

Edited: fn. AddPage() - new parameters added

AddPage(orientation(NO),type="E|O", resetpagenum="1|0", pagenumstyle="I|i|A|a|1", suppress="on|off")

New: fn. docPageNum() - returns the document page number (or '') based on...

New : PageNumSubstitutions(array)

New attributes:

<pagebreak resetpagenum="1" // resets document page numbering to 1 from the new page onwards
<pagebreak suppress="on" // turns on suppression of page numbering i.e. in headers and footers,
{PAGENO} is replaced by blank string
// ="0" turns suppression off
<pagebreak pagenumstyle="I|i|A|a|1" // (re)sets page number stle/type from the new page onwards - as
for lists
// 1 - decimal; A/a - alpha (uppercase/lowercase); I/i - Roman
(uppercase/lowercase)
<pagebreak type="NEXT-ODD" // always adds a page + another if required to make odd
<pagebreak type="NEXT-EVEN" // always adds a page + another if required to make even

Edited: fn. TOC() // sets the marker for a Table of Contents

New parameters allow the page-numbering details to be set

NB the page-numbering details set are for the page which follows on after the TOC marker is inserted.

The page-numbering for the actual ToC is set later, when the ToC is generated and inserted here

new parameters as above for pagebreak resetpagenum="1|0", pagenumstyle="I|i|A|a|1", suppress="on|off"

Can also be set by attribute in the <TOC>

<TOC resetpagenum="1" pagenumstyle="I|i|A|a|1", suppress="on|off" />

Changes to allow Rotated Text in table Cells

Edited:

fn. OpenTag()
fn. _tableColumnWidth()
fn. _tableHeight()
fn. _tableWrite()
fn. tableHeader()

New custom style or attribute -- "text-rotate" -- can be set for either <tr> or <th|td>

Allowed values: 45 - 90 (written as integers) - rotates text anticlockwise, and -90 (clockwise)

Positive values less than 90 force cell to: vertical-align:bottom

Limitations:

Only allows single line of text;

Font, font-size, and style are determined by those set fro the cell, and cannot be changed;

No changes in font (or any other in-line changes e.g. <sup>) are supported within the text

Example: <tr style="text-rotate:90">...

Bug fixes

1) HTML footer containing table was triggering page break.

Added \$this->InHTMLFooter as flag to prevent page triggering in footers containing table

Set in fn.writeHTMLFooters() -> in fn.tableWrite() stops the pageBreak being reset

2) Crashing when libcurl not installed.

Edited OpenTag() curl_init - added if (function_exists) to exclude crash when libcurl not installed

3) Single cell with borders not showing the borders.

e.g. <table><tr><td style="border:1px solid #000000?>Hi</td></tr></table>

Problem: mPDF overrides cell border properties with table border properties for external borders.

\$defaultCSS had CSS defined for table as '0px solid #000000'

Quick fix - line 273 removed. A more complete fix will require reprogramming to distinguish between "border property not set" and border property set as "none".

4) Empty textarea incorrectly handled (the following HTML code being output in the textarea)

The html code for an empty textarea was incorrectly handled in htmltoolkit fn. AdjustHTML which has been edited

```
=====  
mPDF v1.2 (2008-05-01)  
=====
```

```
// Added v1.2 option to continue if invalid UTF-8 chars - used in function is_utf8()  
var $ignore_invalid_utf8 = false;
```

```
Reading CSS in fn. ReadCSS() and applying in fn. MergeCSS() -  
    Edited to allow Tags, class and id with the same name to be distinct i.e. h5 {...} .h5 {...}
```

```
#h5 {...}  
* mPDF 1.2 This version supports: .class {...} / #id { .... }  
* ADDED p {...} h1[-h6] {...} a {...} table {...} thead {...} th {...} td {...} hr {...}  
* body {...} sets default font and fontsize  
* It supports some cascaded CSS e.g. div.topic table.type1 td  
* Does not support non-block level e.g. a#hover { ... }
```

```
Table: font-size, weight, style, family and color should all work  
TD/TH: font-size, weight, style, family and color should all work
```

```
Added to htmltoolkit - fn.array_merge_recursive_unique()
```

```
memory_opt Removed in mPDF v1.2 - not working properly
```

```
fn. _begindoc() - changed to %PDF1.4 (was 1.3) as PDF version
```

```
Write HTML Headers and Footers
```

```
-----  
fn. Close() - calls writeHTMLHeaders/Footers() before finishing doc  
fn. WriteHTML() - added parameter  
fn. _out - writes to outputbuffer when writing HTML footers/headers
```

```
New
```

```
fn. writeHTMLHeaders()  
fn. writeHTMLFooters()
```

```
=====  
mPDF v1.1 (2008-05-01)  
=====
```

```
Programming changes to increase efficiency
```

```
-----  
fn. WriteHTML() - added lines to combine substituted characters <tta> etc
```

```
Memory Optimization added (script from FPDF site) - edited fn. _putpages() and fn. _endpage()
```

```
fn. SetFont() edited to return val quicker if font already set (increase efficiency)
```

```
new vars chrs and ords are used to store chr() and ord() - quicker than using functions
```

```
fn.setMBencoding() - only call mb_internal_encoding if need to change
```

```
Bugs
```

```
-----  
fn. SetDefaultFontSize() - edited to allow to override that set in defaultCSS
```

```
fn. Output() - Added temporary(?) disablement of encryption in CJK as it doesn't work!
```

```
fn. OpenTag() [LI] $this->blockjustfinished=false to prevents newline after first bullet of list  
within table
```

```
Uses of mb_ereg_replace removed, and mb_split changed - requires regex_encoding (regex_encoding only  
used as UTF-8)
```

```
fn. WriteHTML: attributes are trimmed with trim() to allow correct handling of e.g. class="bpmBook "
```

```
fn. printbuffer() and fn. openTag() to ensure  
<div><div><p> outputs top margins/padding for both 1st and 2nd div  
and </p></div></div> ...
```

```
fn. SetFont() added line - bug fixing in CJK fonts
```

```
CSS functionality
```

Added special CSS 'thead-underline' (similar to topntail)

var \$thead_font_weight; added (openTag) to enable setting of font-weight for <TH> cells

Fixed table border inheritance: Table border inherits border="1" to cells, but not table style="border..."

"page-break-inside: avoid" added (var keep_block_together) to allow a DIV or P to be kept on one page

- not compatible with table autosize or table rotate
- only works over maximum of 2 pages

Enhancements

Orphans in line justification: R. Bracket) added to defined list of orphans

allow_url_open

Following a change in the PHP ini config set on my website by my ISP, changes made to allow mPDF to work with allow_url_open=OFF.

- file_get_contents() changed to use libcurl (for CSS files)
- openTag('IMG') @fopen() and 3 functions _parsegif, _parseJPG, _parsePNG, edited to copy remote image files to local file to include images

FlowChart

Changes to enable mPDF work with a custom script producing Flowcharts:

- WriteHTML() second parameter=3 will allow HTML to be parsed but not output
- fn. Arrow() added
- TableWordWrap() added parameter to force array return