
PHPWord Documentation

发布 *0.16.0*

The PHPWord Team

2021 年 03 月 07 日

1 说明	3
1.1 特性	3
1.2 文件格式	4
1.3 贡献	5
2 安装/配置	7
2.1 配置	7
2.2 安装	8
2.3 示例	8
3 常见用法	9
3.1 基本示例	9
3.2 PHPWord 设置	11
3.3 文档设置	12
3.4 文件信息	14
3.5 测量单位	14
3.6 文件保护	15
3.7 自动重新计算打开时的域	15
3.8 连字	15
4 Containers 容器	17
4.1 Sections 节	17
4.2 Headers 页眉	19
4.3 Footers 页脚	19
4.4 Other containers 其他容器	20
5 Elements 元素	21
5.1 Texts 文本	23

5.2	Breaks 分隔	24
5.3	Lists 列表	25
5.4	Tables 表格	26
5.5	Images 图片	26
5.6	Objects 对象	27
5.7	Table of contents 目录	28
5.8	Footnotes & endnotes 脚注和尾注	28
5.9	Checkboxes 复选框	29
5.10	Textboxes 文本框	30
5.11	Fields 域	30
5.12	Line 行	31
5.13	Chart 图表	31
5.14	Comments 注释	31
5.15	Track Changes 跟踪变化	32
6	Styles 样式	33
6.1	Section 节	33
6.2	Font 字体	34
6.3	Paragraph 段落	35
6.4	Table 表格	36
6.5	Image 图像	37
6.6	Numbering level 编号级别	38
6.7	Chart 图表	38
7	模板处理	41
7.1	设单值	41
7.2	设置一组值	42
7.3	setImageValue 设置图像值	42
7.4	克隆块	43
7.5	replaceBlock 替换块	44
7.6	deleteBlock 删除块	44
7.7	cloneRow 克隆行	44
7.8	cloneRowAndSetValues 克隆并赋值	45
7.9	应用 xsl 样式表	45
7.10	设置复杂值	46
7.11	设置复杂块	46
8	Writers & readers	47
8.1	OOXML	47
8.2	OpenDocument	48
8.3	RTF	49
8.4	HTML	49
8.5	PDF	49

9 Recipes 小记	51
9.1 创建左浮动图像	51
9.2 自动下载生成的文件	51
9.3 创建编号标题	52
9.4 在标题中添加链接	53
9.5 删除 MS Word 标题栏中的 [Compatibility Mode] 文本	53
10 常见问题	55
10.1 如何贡献 PHPWord?	55
10.2 这与我在 CodePlex 中找到的 PHPWord 相同吗?	55
10.3 我一直在完美地运行来自 CodePlex 的 PHPWord, 但是我不能使用来自 GitHub 的最新 PHP-Word。为什么?	55
11 Credits	57
12 References 参考	59
12.1 ISO/IEC 29500, Third edition, 2012-09-01	59
12.2 正式规格	59
12.3 其他资源	59
13 Indices and tables	61



PHPWord 是一个用纯 PHP 编写的库,它提供了一组类来写入和读取不同的文档文件格式。当前版本 PHPWord 支持 Microsoft Office Open XML (OOXML 或 OpenXML), OASIS Open Office 应用程序 (OpenDocument 或 ODF) 的文档格式和富文本格式 (RTF)。

PHPWord 是一个用纯 PHP 编写的库,它提供了一组类写入和读取不同的文档文件格式。当前 PHPWord 版本支持微软的 Office Open XML <http://en.wikipedia.org/wiki/Office_Open_XML> (OOXML or OpenXML), OASIS 的 Office 开放文档格式应用程序 (OpenDocument or ODF), and 富文本格式 (RTF)。

PHPWord is an open source project licensed under the terms of [LGPL version 3](#). PHPWord is aimed to be a high quality software product by incorporating [continuous integration](#) and [unit testing](#). You can learn more about PHPWord by reading this [Developers' Documentation](#) and the [API Documentation](#).

1.1 特性

- 设置文档属性, 例如标题、主题和创建者。
- 创建具有不同设置的文档部分, 例如

纵向/横向、页面大小和页码 - 为每个部分创建页眉和页脚 - 设置默认字体类型、字体大小和段落样式 - 使用 UTF-8 和东亚字体/字符 - 定义自定义字体样式 (例如粗体、斜体、颜色) 和段落样式 (例如居中、多柱、间距) 作为命名样式或内联文本 - 插入段落, 以简单文本或复杂文本 (文本运行), 其中包含其他元素 - 插入标题 (标题) 和目录 - 插入文本符和分页符 - 从右到左插入文本 - 插入和格式化本地、远程或作为页面水印的图像 - 插入二进制 OLE 对象, 如 Excel 或 Visio - 为每一行插入和格式化具有自定义属性的表 (例如, 作为标题行重复) 和单元格 (例如, 背景色, rowspan, colspan) - 插入列表项, 如项目符号、编号或多级 - 插入超链接 - 插入脚注和尾注 - 插入绘图形状 (圆弧、曲线、线、折线、矩形、椭圆形) - 插入图表 (馅饼、甜甜圈、条、线、面积、散射、雷达) - 插入表单字段 (文本输入、复选框和下拉菜单) - 插入注释 - 从模板创建文档 - 使用 [shrna 1.0](#) 样式表转换 OOXML 模板的标题、主文档部分和页脚 - ……还有更多进展中的功能

1.2 文件格式

以下是每种文件格式支持的功能。

1.2.1 写入

Features		OOXML	ODF	RTF	HTML	PDF
Document Properties	Standard	✓	✓	✓	✓	✓
	Custom	✓	✓			
Element Type	Text	✓	✓	✓	✓	✓
	Text Run	✓	✓	✓	✓	✓
	Title	✓	✓		✓	✓
	Link	✓	✓	✓	✓	✓
	Preserve Text	✓				
	Text Break	✓	✓	✓	✓	✓
	Page Break	✓		✓		
	List	✓				
	Table	✓	✓	✓	✓	✓
	Image	✓	✓	✓	✓	
	Object	✓				
	Watermark	✓				
	Table of Contents	✓				
	Header	✓				
	Footer	✓				
	Footnote	✓			✓	
	Endnote	✓			✓	
	Comments	✓				
Graphs	2D basic graphs	✓				
	2D advanced graphs					
	3D graphs	✓				
Math	OMML support					
	MathML support					
Bonus	Encryption					
	Protection					

1.2.2 读取

Features		OOXML	DOC	ODF	RTF	HTML
Document Properties	Standard	✓				
	Custom	✓				
Element Type	Text	✓	✓	✓	✓	✓
	Text Run	✓				
	Title	✓		✓		
	Link	✓	✓			
	Preserve Text	✓				
	Text Break	✓	✓			
	Page Break	✓				
	List	✓		✓		✓
	Table	✓				✓
	Image	✓	✓			
	Object					
	Watermark					
	Table of Contents					
	Header	✓				
	Footer	✓				
	Footnote	✓				
	Endnote	✓				
	Comments					
Graphs	2D basic graphs					
	2D advanced graphs					
	3D graphs					
Math	OMML support					
	MathML support					
Bonus	Encryption					
	Protection					

1.3 贡献

我们欢迎大家为 PHPWord 做出贡献。以下是一些你可以做的贡献。

- 读 我们的贡献指南。
- Fork us 然后 拉取 develop 分支。
- 提交 提交 bug 或特性请求 到 GitHub。

- 关注 @PHPWord 和 @PHPOffice 在 Twitter。

2.1 配置

强制:

- PHP 5.3.3+
- XML Parser 扩展
- Zend\Escaper 组件
- Zend\Stdlib 组件
- Zend\Validator 组件

可选:

- Zip 扩展
- GD 扩展
- XMLWriter 扩展
- XSL 扩展
- dompdf 库

2.2 安装

PHPWord 通过 Composer <<https://getcomposer.org/>>“安装”。你只需要在你的包中“添加依赖”。

Example:

```
{
  "require": {
    "phpoffice/phpword": "v0.14.*"
  }
}
```

如果您是开发人员，或者您想帮助我们进行测试，请为开发人员获取最新的分支。注意：所有贡献必须针对开发者分支机构。

Example:

```
{
  "require": {
    "phpoffice/phpword": "dev-develop"
  }
}
```

2.3 示例

“samples”目录中提供了更多示例。为了便于访问这些示例，请在 samples 目录中启动“php-s localhost:8000”，然后浏览到 <http://localhost:8000> 以查看示例。

3.1 基本示例

下面是 PHPWord 库的基本示例。更多的示例在 `samples` 目录下。

```
<?php
require_once 'bootstrap.php';

// Creating the new document...
$phpWord = new \PhpOffice\PhpWord\PhpWord();

/* Note: any element you append to a document must reside inside of a Section. */

// Adding an empty Section to the document...
$section = $phpWord->addSection();
// Adding Text element to the Section having font styled by default...
$section->addText (
    "Learn from yesterday, live for today, hope for tomorrow. '
        . 'The important thing is not to stop questioning.' '
        . '(Albert Einstein)'"
);

/*
 * Note: it's possible to customize font style of the Text element you add in three_
 ↪ways:
```

(下页继续)

```
* - inline;
* - using named font style (new font style object will be implicitly created);
* - using explicitly created font style object.
*/

// Adding Text element with font customized inline...
$section->addText(
    '"Great achievement is usually born of great sacrifice, '
    . 'and is never the result of selfishness." '
    . '(Napoleon Hill)',
    array('name' => 'Tahoma', 'size' => 10)
);

// Adding Text element with font customized using named font style...
$fontStyleName = 'oneUserDefinedStyle';
$phpWord->addFontStyle(
    $fontStyleName,
    array('name' => 'Tahoma', 'size' => 10, 'color' => '1B2232', 'bold' => true)
);
$section->addText(
    '"The greatest accomplishment is not in never falling, '
    . 'but in rising again after you fall." '
    . '(Vince Lombardi)',
    $fontStyleName
);

// Adding Text element with font customized using explicitly created font style.
↳object...
$fontStyle = new \PhpOffice\PhpWord\Style\Font();
$fontStyle->setBold(true);
$fontStyle->setName('Tahoma');
$fontStyle->setSize(13);
$myTextElement = $section->addText('"Believe you can and you\'re halfway there."↳
↳(Theodor Roosevelt)');
$myTextElement->setFontStyle($fontStyle);

// Saving the document as OOXML file...
$objWriter = \PhpOffice\PhpWord\IOFactory::createWriter($phpWord, 'Word2007');
$objWriter->save('helloWorld.docx');

// Saving the document as ODF file...
$objWriter = \PhpOffice\PhpWord\IOFactory::createWriter($phpWord, 'ODText');
$objWriter->save('helloWorld.odt');
```


(续上页)

```
// Saving the document as HTML file...
$objWriter = \PhpOffice\PhpWord\IOFactory::createWriter($phpWord, 'HTML');
$objWriter->save('helloWorld.html');

/* Note: we skip RTF, because it's not XML-based and requires a different example. */
/* Note: we skip PDF, because "HTML-to-PDF" approach is used to create PDF documents. */
↪*/
```

3.2 PHPWord 设置

PhpOffice\PhpWord\Settings 类提供了一些选项，这些选项将影响 PHPWord 的行为。以下是选项。

3.2.1 XML Writer compatibility

此选项设置 XMLWriter::setIndent 和 XMLWriter::setIndentString。默认值是 true (兼容), 是为了 OpenOffice 来正确呈现 OOXML 文档。在开发过程中, 你可以设置为 false 使生成的 XML 文件更易于阅读。

```
\PhpOffice\PhpWord\Settings::setCompatibility(false);
```

3.2.2 Zip 类

默认的, PHPWord 使用 Zip extension 处理压缩的压缩档案和其中的文件。如果你不能 Zip 扩展安装在您的服务器上, 您可以使用纯 PHP 库替代, PclZip, 其已经内嵌在 PHPWord 中。

```
\PhpOffice\PhpWord\Settings::setZipClass(\PhpOffice\PhpWord\Settings::PCLZIP);
```

3.2.3 输出转义

编写某些格式的文档, 尤其是基于 XML 的文档, 需要正确的输出转义。没有它, 当你在文档中添加特殊字符, 如 & 符号、引号和其他字符时, 您的文档可能会损坏。转义可以通过两种方式执行: 软件开发人员在库之外, 通过内置机制在库内部。默认情况下, 为了与 v0.13.0 之前的版本向后兼容, 禁用了内置机制。要将其打开, 请在 PHPWord 配置文件中将 *outputescape ingenabled* 选项设置为 'true', 或在运行时使用以下指令:

```
\PhpOffice\PhpWord\Settings::setOutputEscapingEnabled(true);
```

3.2.4 默认字体

默认情况下，每个文本都显示在 Arial 10 字号中。您可以更改使用以下两个函数默认字体：

```
$phpWord->setDefaultFontName('Times New Roman');  
$phpWord->setDefaultFontSize(12);
```

3.3 文档设置

可以使用“`$phpWord->getSettings()`”获取生成的文档的设置

3.3.1 放大设置

默认缩放值为 100%。这可以更改为另一个百分比

```
$phpWord->getSettings()->setZoom(75);
```

或者预定义的值 `fullPage`, `bestFit`, `textFit`

```
$phpWord->getSettings()->setZoom(Zoom::BEST_FIT);
```

3.3.2 镜像页边距

使用镜像边距为双面文档 (如书籍或杂志) 设置面向页面。

```
$phpWord->getSettings()->setMirrorMargins(true);
```

3.3.3 拼写和语法检查

默认情况下，打开 word 文档后会显示拼写和语法错误。对于大型文档，这会减慢文档的打开速度。您可以通过以下方式隐藏拼写和/或语法错误：

```
$phpWord->getSettings()->setHideGrammaticalErrors(true);  
$phpWord->getSettings()->setHideSpellingErrors(true);
```

您还可以指定拼写和语法检查的状态，将拼写或语法标记为脏将在打开文档时强制重新检查。

```
$proofState = new ProofState();  
$proofState->setGrammar(ProofState::CLEAN);  
$proofState->setSpelling(ProofState::DIRTY);
```

(下页继续)

(续上页)

```
$phpWord->getSettings()->setProofState($proofState);
```

3.3.4 跟踪修订

可以使用 `setTrackRevisions` 激活跟踪修订, 您可以进一步指定

- 不使用移动语法, 而是移动的项目将在一个地方被视为删除, 并在另一个地方被添加
- 不跟踪格式修订

```
$phpWord->getSettings()->setTrackRevisions(true);  
$phpWord->getSettings()->setDoNotTrackMoves(true);  
$phpWord->getSettings()->setDoNotTrackFormatting(true);
```

3.3.5 十进制符号

表示十进制数字的默认符号是英语中的 `.`。例如, 在法语中, 您可能需要将其更改为 `,`。

```
$phpWord->getSettings()->setDecimalSymbol(',');
```

3.3.6 文档语言

可以通过以下方式更改文档的默认语言。

```
$phpWord->getSettings()->setThemeFontLang(new Language(Language::FR_BE));
```

`Language` 有 3 个参数, 一个用于拉丁语, 一个用于东亚语言, 一个用于复杂 (双向) 语言。几个语言代码在 `PhpOffice\PhpWord\ComplexType\Language` 类里提供了但是也可使用任何有效的 `code/ID`。

如果您要生成 **RTF** 文档, 则需要对语言进行不同的设置。

```
$lang = new Language();  
$lang->setLangId(Language::EN_GB_ID);  
$phpWord->getSettings()->setThemeFontLang($lang);
```

3.4 文件信息

您可以设置文档信息，如标题、创建者和公司名称。使用以下功能:

```
$properties = $phpWord->getDocInfo();
$properties->setCreator('My name');
$properties->setCompany('My factory');
$properties->setTitle('My title');
$properties->setDescription('My description');
$properties->setCategory('My category');
$properties->setLastModifiedBy('My name');
$properties->setCreated(mktime(0, 0, 0, 3, 12, 2014));
$properties->setModified(mktime(0, 0, 0, 3, 14, 2014));
$properties->setSubject('My subject');
$properties->setKeywords('my, key, word');
```

3.5 测量单位

Open Office XML 中的基本长度单位是 twip。Twip 的意思是“二十一分之一英寸点”，即 1 twip = 1/1440 英寸。

您可以使用 PHPWord 帮助函数来转换英寸、厘米或点 twip。

```
// Paragraph with 6 points space after
$phpWord->addParagraphStyle('My Style', array(
    'spaceAfter' => \PhpOffice\PhpWord\Shared\Converter::pointToTwip(6)
));

$section = $phpWord->addSection();
$sectionStyle = $section->getStyle();
// half inch left margin
$sectionStyle->setMarginLeft(\PhpOffice\PhpWord\Shared\Converter::inchToTwip(.5));
// 2 cm right margin
$sectionStyle->setMarginRight(\PhpOffice\PhpWord\Shared\Converter::cmToTwip(2));
```

3.6 文件保护

文档 (或部分文档) 可以通过密码保护。

```
$documentProtection = $phpWord->getSettings()->getDocumentProtection();  
$documentProtection->setEditing(DocProtect::READ_ONLY);  
$documentProtection->setPassword('myPassword');
```

3.7 自动重新计算打开时的域

要强制更新文档中存在的字段，请将 `updateFields` 设置为 `true`

```
$phpWord->getSettings()->setUpdateFields(true);
```

3.8 连字

连字符描述了用连字符破译单词的过程。有几个选项可以控制连字符。

3.8.1 自动连字符

自动断字文本套装 `autoHyphenation` 转 `'true'`。

```
$phpWord->getSettings()->setAutoHyphenation(true);
```

3.8.2 连续连字符限制

以连字符结尾的文本的最大连续行数可以由 `连续连字符限制` 选项控制。如果选项未设置或提供的值为 0，则没有限制。

```
$phpWord->getSettings()->setConsecutiveHyphenLimit(2);
```

3.8.3 连字符区

连字符区域 (在 *twip* 中) 是应用连字符之前允许的空格量。连字符区越小，连字符越多。或者换句话说，连字符区越宽，连字符就越少。

```
$phpWord->getSettings()->setHyphenationZone(\PhpOffice\PhpWord\Shared\  
↳Converter::cmToTwip(1));
```

3.8.4 连字符帽

为了控制所有大写字母中的单词是否应使用 *donohyphenatecap* 选项连字符。

```
$phpWord->getSettings()->setDoNotHyphenateCaps(true);
```

容器是可以放置元素(文本、列表、表、等等)。有 3 个主要集装箱, 即节、页眉和页脚。有 3 个元素也可以充当容器, 即 `textruns`, 表格单元格和脚注。

4.1 Sections 节

`word` 中的每个可见元素都放置在一个部分内。创建一节, 使用以下代码:

```
$section = $phpWord->addSection($sectionStyle);
```

“`$sectionStyle`”是一个可选的关联数组, 它设置在节上。示例:

```
$sectionStyle = array(  
    'orientation' => 'landscape',  
    'marginTop' => 600,  
    'colsNum' => 2,  
);
```

4.1.1 Page number 页码

您可以使用 `pageNumberingStart` 更改节页码部分的样式。

```
// Method 1
$section = $phpWord->addSection(array('pageNumberingStart' => 1));

// Method 2
$section = $phpWord->addSection();
$section->getStyle()->setPageNumberingStart(1);
```

4.1.2 Multicolumn 多列

您可以通过以下方式将节布局更改为多列 (如在报纸中) 使用节的 `breaktype` 和 `colsnum` 样式。

```
// Method 1
$section = $phpWord->addSection(array('breakType' => 'continuous', 'colsNum' => 2));

// Method 2
$section = $phpWord->addSection();
$section->getStyle()->setBreakType('continuous');
$section->getStyle()->setColsNum(2);
```

4.1.3 Line numbering 行号

您可以使用“`lineNumbering`”的样式将行号应用于节。

```
// Method 1
$section = $phpWord->addSection(array('lineNumbering' => array()));

// Method 2
$section = $phpWord->addSection();
$section->getStyle()->setLineNumbering(array());
```

以下是行号样式的属性。

- `start` Line numbering starting value
- `increment` Line number increments
- `distance` Distance between text and line numbering in *twip*
- `restart` Line numbering restart setting continuous/newPage/newSection

4.2 Headers 页眉

每个部分都可以有自己的标题引用。要创建标题，请使用 `Addheader` 方法：

```
$header = $section->addHeader();
```

确保将结果保存在本地对象中。您可以使用所有元素可用于页脚。有关详细信息，请参见 页脚部分。此外，只有在标题引用内部，您才能添加水印或者背景图片。参见 水印部分。

您可以传递一个可选参数来指定页眉/页脚的应用位置，它可以是

- `Footer:::AUTO` default, all pages except if overridden by first or even
- `Footer:::FIRST` each first page of the section
- `Footer:::EVEN` each even page of the section. Will only be applied if the `evenAndOddHeaders` is set to true in `phpWord->settings`

要更改 `evenandodheader`，请使用 `getSettings` 方法返回 `Settings` 对象，然后调用 `setEvenAndOddHeaders` 方法：

```
$phpWord->getSettings()->setEvenAndOddHeaders(true);
```

4.3 Footers 页脚

每个部分都可以有自己的页脚参考。要创建页脚，请使用“`addFooter`”方法：

```
$footer = $section->addFooter();
```

确保将结果保存在本地对象中以将元素添加到页脚。您可以将以下元素添加到页脚：

- Texts `addText` and `createTextRun`
- Text breaks
- Images
- Tables
- Preserve text

有关每个元素的详细信息，请参见 `Elements` 部分。

4.4 Other containers 其他容器

Textruns、表格单元格和脚注是也可以充当集装箱。有关以下内容的详细信息，请参阅相应的“元素”部分每个元素。

CHAPTER 5

Elements 元素

以下是每个容器中的元素可用性矩阵。列显示容器，而行列出元素。

Num	Element	Section	Header	Footer	Cell	Text Run	Footnote
1	Text	v	v	v	v	v	v
2	Text Run	v	v	v	v	•	•
3	Link	v	v	v	v	v	v
4	Title	v	?	?	?	?	?
5	Preserve Text	?	v	v	v*	•	•
6	Text Break	v	v	v	v	v	v
7	Page Break	v	•	•	•	•	•
8	List	v	v	v	v	•	•
9	Table	v	v	v	v	•	•
10	Image	v	v	v	v	v	v
11	Watermark	•	v	•	•	•	•
12	OLEObject	v	v	v	v	v	v
13	TOC	v	•	•	•	•	•
14	Footnote	v	•	•	v**	v**	•
15	Endnote	v	•	•	v**	v**	•
16	CheckBox	v	v	v	v	v	•
17	TextBox	v	v	v	v	•	•
18	Field	v	v	v	v	v	v
19	Line	v	v	v	v	v	v
20	Chart	v			v		

Legend:

- v. 可用.
- v*. 只在 header/footer 中可用.

- `v**`. 只在节中可用。
- `-`. 不可用。
- `?`. 应该可用。

5.1 Texts 文本

文本可以使用 `addText` 和 `addTextRun` 被添加。`addText` 用于创建仅包含相同样式的文本的简单段落。`addTextRun` 用于创建包含不同样式的文本的复杂段落(一些粗体, 其他斜体等)或其他元素, 例如图像或链接。语法如下:

```
$section->addText($text, [$fontStyle], [$paragraphStyle]);
$textrun = $section->addTextRun([$paragraphStyle]);
```

- `$text`. Text to be displayed in the document.
- `$fontStyle`. See *Font* 字体.
- `$paragraphStyle`. See *Paragraph* 段落.

有关可用样式选项, 请参阅:ref:font-style and *Paragraph* 段落.

如果要对添加的文本启用跟踪更改, 可以为特定用户在给定时间将其标记 `INSERTED` 或 `DELETED` :

```
$text = $section->addText('Hello World!');
$text->setChanged(\PhpOffice\PhpWord\Element\ChangedElement::TYPE_INSERTED, 'Fred',
↪ (new \DateTime()));
```

5.1.1 Titles 标题

如果要构建文档或构建目录, 则需要标题或标题。要向文档添加标题, 请使用 `addTitleStyle`和 addTitle`方法。如果 depth 为 0, 将插入标题, 否则插入标题 1, 标题 2, ...`

```
$phpWord->addTitleStyle($depth, [$fontStyle], [$paragraphStyle]);
$section->addTitle($text, [$depth]);
```

- `depth`.
- `$fontStyle`. 参见*Font* 字体.
- `$paragraphStyle`. 参见*Paragraph* 段落.
- `$text`. 文档中要显示的文本。可以使 `'string'` 或一个 `PhpOfficePhpWordElementTextRun`

有必要在文档中添加标题样式, 否则标题将不会被检测为真实标题。

5.1.2 Links 链接

您可以使用函数 `addLink` 向文档添加超链接:

```
$section->addLink($linkSrc, [$linkName], [$fontStyle], [$paragraphStyle]);
```

- `$linkSrc`. The URL of the link.
- `$linkName`. Placeholder of the URL that appears in the document.
- `$fontStyle`. See *Font* 字体.
- `$paragraphStyle`. See *Paragraph* 段落.

5.1.3 Preserve texts

The `addPreserveText` method is used to add a page number or page count to headers or footers.

```
$footer->addPreserveText('Page {PAGE} of {NUMPAGES}.');
```

5.2 Breaks 分隔

5.2.1 Text breaks 文本分隔

文本分隔是空的新行。要添加文本分隔, 请使用以下语法。所有参数都是可选的。

```
$section->addTextBreak([$breakCount], [$fontStyle], [$paragraphStyle]);
```

- `$breakCount`. How many lines.
- `$fontStyle`. See *Font* 字体.
- `$paragraphStyle`. See *Paragraph* 段落.

5.2.2 Page breaks

有两种方法可以使用 `addPageBreak` 插入分页符或使用段落的 `pageBreakBefore` 样式。

```
$section->addPageBreak();
```

5.3 Lists 列表

可以使用 `addListItem` 和 “`addListItemRun`” 方法添加列表。`addListItem` 用于创建仅包含纯文本的列表。`addListItemRun` 用于创建包含文本的复杂列表项具有不同的风格 (一些粗体、其他斜体等) 或其他元素, 例如图像或链接。语法如下:

基本用法:

```
$section->addListItem($text, [$depth], [$fontStyle], [$listStyle], [$paragraphStyle]);
$listItemRun = $section->addListItemRun([$depth], [$listStyle], [$paragraphStyle])
```

Parameters:

- `$text`. Text that appears in the document.
- `$depth`. Depth of list item.
- `$fontStyle`. See *Font* 字体.
- `$listStyle`. List style of the current element `TYPE_NUMBER`, `TYPE_ALPHANUM`, `TYPE_BULLET_FILLED`, etc. See list of constants in `PHPWord\Style\ListItem`.
- `$paragraphStyle`. See *Paragraph* 段落.

参考 `Sample_09_Tables.php` 获得更多示例.

高级用法:

您还可以通过使用编号样式的名称更改 `$listStyle` 参数来创建自己的编号样式。

```
$phpWord->addNumberingStyle(
    'multilevel',
    array(
        'type' => 'multilevel',
        'levels' => array(
            array('format' => 'decimal', 'text' => '%1.', 'left' => 360, 'hanging' =>
↳360, 'tabPos' => 360),
            array('format' => 'upperLetter', 'text' => '%2.', 'left' => 720, 'hanging
↳' => 360, 'tabPos' => 720),
        )
    )
);
$section->addListItem('List Item I', 0, null, 'multilevel');
$section->addListItem('List Item I.a', 1, null, 'multilevel');
$section->addListItem('List Item I.b', 1, null, 'multilevel');
$section->addListItem('List Item II', 0, null, 'multilevel');
```

可用样式参考 *Numbering level* 编号级别.

5.4 Tables 表格

要添加表格，行和单元格使用 `addTable`, `addRow`, and `addCell` 方法:

```
$table = $section->addTable([$tableStyle]);  
$table->addRow([$height], [$rowStyle]);  
$cell = $table->addCell($width, [$cellStyle]);
```

可以通过 `addTableStyle` 定义表格样式:

```
$tableStyle = array(  
    'borderColor' => '006699',  
    'borderSize'  => 6,  
    'cellMargin'  => 50  
);  
$firstRowStyle = array('bgColor' => '66BBFF');  
$phpWord->addTableStyle('myTable', $tableStyle, $firstRowStyle);  
$table = $section->addTable('myTable');
```

可用样式参考 *Table* 表格.

5.4.1 Cell span

您可以使用 `gridSpan` 跨多列的单元格，也可以使用 `vMerge` 跨多行。

```
$cell = $table->addCell(200);  
$cell->getStyle()->setGridSpan(5);
```

See `Sample_09_Tables.php` for more code sample.

5.5 Images 图片

要添加图像，请对节、页眉、页脚、`textrun` 或表格单元格使用“`addImage`”方法。

```
$section->addImage($src, [$style]);
```

- `$src`. 本地图像的字符串路径、远程图像的 URL 或图像数据 (作为字符串)。警告: 不要在此传递用户生成的字符串，因为这将允许攻击者通过传递文件路径或 url 而不是图像数据来读取任意文件或执行服务器端请求伪造。
- `$style`. See *Image* 图像.

示例:


```

$section = $phpWord->addSection();
$section->addImage(
    'mars.jpg',
    array(
        'width'           => 100,
        'height'          => 100,
        'marginTop'       => -1,
        'marginLeft'      => -1,
        'wrappingStyle'   => 'behind'
    )
);
$footer = $section->addFooter();
$footer->addImage('http://example.com/image.php');
$textRun = $section->addTextRun();
$textRun->addImage('http://php.net/logo.jpg');
$source = file_get_contents('/path/to/my/images/earth.jpg');
$textRun->addImage($source);

```

5.5.1 Watermarks 水印

要添加水印 (或页面背景图像), 您的节需要标题参考。创建标题后, 您可以使用添加水印的 `addWatermark` 方法。

```

$section = $phpWord->addSection();
$header = $section->addHeader();
$header->addWatermark('resources/_earth.jpg', array('marginTop' => 200, 'marginLeft' => 55));

```

5.6 Objects 对象

您可以添加 OLE 嵌入, 例如 Excel 电子表格或 PowerPoint 使用 `addOLEObject` 方法对文档进行演示。

```

$section->addOLEObject($src, [$style]);

```

5.7 Table of contents 目录

要添加目录 (TOC), 可以使用 `addTOC` 方法。只有添加了至少一个标题 (请参阅 “Titles”), 才能生成 TOC。

```
$section->addTOC([$fontStyle], [$tocStyle], [$minDepth], [$maxDepth]);
```

- `$fontStyle`. See font style section.
- `$tocStyle`. See available options below.
- `$minDepth`. Minimum depth of header to be shown. Default 1.
- `$maxDepth`. Maximum depth of header to be shown. Default 9.

Options for `$tocStyle`:

- `tabLeader`. Fill type between the title text and the page number. Use the defined constants in `\PhpOffice\PhpWord\Style\TOC`.
- `tabPos`. The position of the tab where the page number appears in *twip*.
- `indent`. The indent factor of the titles in *twip*.

5.8 Footnotes & endnotes 脚注和尾注

您可以使用 `addFootnote` 创建脚注, 并使用文本或 `textruns` 中的 `addEndnote`, 但建议使用 `textrun` 获得更好的布局。您可以脚注和尾注上使用 `addText`、`addLink`、`addTextBreak`、`addImage`、`addOLEObject`。

在 `textrun`:

```
$textrun = $section->addTextRun();
$textrun->addText('Lead text.');
```

```
$footnote = $textrun->addFootnote();
$footnote->addText('Footnote text can have ');
$footnote->addLink('http://test.com', 'links');
$footnote->addText('. ');
$footnote->addTextBreak();
$footnote->addText('And text break.');
```

```
$textrun->addText('Trailing text.');
```

```
$endnote = $textrun->addEndnote();
$endnote->addText('Endnote put at the end');
```

在 `text`:

```
$section->addText('Lead text.');
```

```
$footnote = $section->addFootnote();
$footnote->addText('Footnote text.');
```

默认情况下，脚注参考编号将显示为十进制编号。从 1 开始。该数字使用“FooterReference”样式，您可以使用 `addFontStyle` 方法重新定义。此样式的默认值为 `array('superScript' => true)`;

脚注编号可以通过在该节设置脚注属性来控制。

```
$fp = new PhpWord\SimpleType\FootnoteProperties();
//sets the position of the footnote (pageBottom (default), beneathText, sectEnd, ↵
↳ docEnd)
$fp->setPos(FootnoteProperties::POSITION_DOC_END);
//set the number format to use (decimal (default), upperRoman, upperLetter, ...)
$fp->setNumFmt(FootnoteProperties::NUMBER_FORMAT_LOWER_ROMAN);
//force starting at other than 1
$fp->setNumStart(2);
//when to restart counting (continuous (default), eachSect, eachPage)
$fp->setNumRestart(FootnoteProperties::RESTART_NUMBER_EACH_PAGE);
//And finally, set it on the Section
$section->setFootnoteProperties($properties);
```

5.9 Checkboxes 复选框

可以通过 `addCheckBox` 将复选框元素添加到节或表格单元格中。

```
$section->addCheckBox($name, $text, [$fontStyle], [$paragraphStyle])
```

- `$name`. Name of the check box.
- `$text`. Text to be displayed in the document.
- `$fontStyle`. See *Font* 字体.
- `$paragraphStyle`. See *Paragraph* 段落.

5.10 Textboxes 文本框

待完成

5.11 Fields 域

目前支持以下字段:

- PAGE
- NUMPAGES
- DATE
- XE
- INDEX

```
$section->addField($fieldType, [$properties], [$options], [$fieldText])
```

参考 `\PhpOffice\PhpWord\Element\Field` for list of properties and options available for each field type. Options which are not specifically defined can be added. Those must start with a `\`.

For instance for the INDEX field, you can do the following (See [Index Field](#) for list of available options):

```
//the $fieldText can be either a simple string
$fieldText = 'The index value';

//or a 'TextRun', to be able to format the text you want in the index
$fieldText = new TextRun();
$fieldText->addText('My ');
$fieldText->addText('bold index', ['bold' => true]);
$fieldText->addText(' entry');
$section->addField('XE', array(), array(), $fieldText);

//this actually adds the index
$section->addField('INDEX', array(), array('\e " " \h "A" \c "3"'), 'right click_
↳to update index');
```

5.12 Line 行

可以使用“addLine”将行元素添加到节中。

```
$lineStyle = array('weight' => 1, 'width' => 100, 'height' => 0, 'color' => 635552);
$section->addLine($lineStyle);
```

可用的行样式属性:

- `weight`. Line width in *twip*.
- `color`. Defines the color of stroke.
- `dash`. Line types: dash, rounddot, squaredot, dashdot, longdash, longdashdot, longdashdotdot.
- `beginArrow`. Start type of arrow: block, open, classic, diamond, oval.
- `endArrow`. End type of arrow: block, open, classic, diamond, oval.
- `width`. Line-object width in *pt*.
- `height`. Line-object height in *pt*.
- `flip`. Flip the line element: true, false.

5.13 Chart 图表

可以通过下方代码添加图表

```
$categories = array('A', 'B', 'C', 'D', 'E');
$series = array(1, 3, 2, 5, 4);
$chart = $section->addChart('line', $categories, $series, $style);
```

可见样式选项参考 *Chart* 图表。

查看 `Sample_32_Chart.php` 获得更多选项和样式。

5.14 Comments 注释

可以使用 `addComment` 将注释添加到文档中。注释可以包含格式化文本。添加注释后，可以将其链接到具有 `setCommentStart` 的任何元素。

```
// first create a comment
$comment = new \PhpOffice\PhpWord\Element\Comment('Authors name', new \DateTime(), 'my_
↪initials');
$comment->addText('Test', array('bold' => true));
```

(下页继续)

```
// add it to the document
$phpWord->addComment($comment);

$textrun = $section->addTextRun();
$textrun->addText('This ');
$text = $textrun->addText('is');
// link the comment to the text you just created
$text->setCommentStart($comment);
```

如果没有使用 `setCommentEnd` 为注释设置结束，则注释将在其启动的元素的末尾自动结束。

5.15 Track Changes 跟踪变化

可以在文本元素上设置轨道更改。有两种方法可以设置元素的更改信息。通过调用 `setChangeInfo()` 或使用 `setTrackChange()` 在元素上设置 `TrackChange` 实例。

```
$phpWord = new \PhpOffice\PhpWord\PhpWord();

// New portrait section
$section = $phpWord->addSection();
$textRun = $section->addTextRun();

$text = $textRun->addText('Hello World! Time to ');

$text = $textRun->addText('wake ', array('bold' => true));
$text->setChangeInfo(TrackChange::INSERTED, 'Fred', time() - 1800);

$text = $textRun->addText('up');
$text->setTrackChange(new TrackChange(TrackChange::INSERTED, 'Fred'));

$text = $textRun->addText('go to sleep');
$text->setChangeInfo(TrackChange::DELETED, 'Barney', new \DateTime('@' . (time() - 3600)));
```

6.1 Section 节

可用节样式选项:

- `borderBottomColor`. Border bottom color.
- `borderBottomSize`. Border bottom size in *twip*.
- `borderLeftColor`. Border left color.
- `borderLeftSize`. Border left size in *twip*.
- `borderRightColor`. Border right color.
- `borderRightSize`. Border right size in *twip*.
- `borderTopColor`. Border top color.
- `borderTopSize`. Border top size in *twip*.
- `breakType`. Section break type (`nextPage`, `nextColumn`, `continuous`, `evenPage`, `oddPage`).
- `colsNum`. Number of columns.
- `colsSpace`. Spacing between columns.
- `footerHeight`. Spacing to bottom of footer.
- `gutter`. Page gutter spacing.
- `headerHeight`. Spacing to top of header.

- `marginTop`. Page margin top in *twip*.
- `marginLeft`. Page margin left in *twip*.
- `marginRight`. Page margin right in *twip*.
- `marginBottom`. Page margin bottom in *twip*.
- **orientation**. Page orientation (**portrait**, which is default, or **landscape**). See `\PhpOffice\PhpWord\Style\Section::ORIENTATION_...` class constants for possible values
- `pageSizeH`. Page height in *twip*. Implicitly defined by `orientation` option. Any changes are discouraged.
- `pageSizeW`. Page width in *twip*. Implicitly defined by `orientation` option. Any changes are discouraged.
- **vAlign**. Vertical Page Alignment See `\PhpOffice\PhpWord\SimpleType\VerticalJc` for possible values

6.2 Font 字体

可用字体样式选项:

- `allCaps`. A 全大写, *true* or *false*.
- `bgColor`. 字体背景颜色, e.g. *FF0000*.
- `bold`. Bold, *true* or *false*.
- `color`. Font color, e.g. *FF0000*.
- `doubleStrikethrough`. Double strikethrough, *true* or *false*.
- **fgColor**. Font highlight color, e.g. *yellow, green, blue*. See `\PhpOffice\PhpWord\Style\Font::FGCOLOR_...` class constants for possible values
- `hint`. Font content type, *default*, *eastAsia*, or *cs*.
- `italic`. Italic, *true* or *false*.
- `name`. Font name, e.g. *Arial*.
- `rtl`. Right to Left language, *true* or *false*.
- `size`. Font size, e.g. *20, 22*.
- `smallCaps`. Small caps, *true* or *false*.
- `strikethrough`. Strikethrough, *true* or *false*.
- `subScript`. Subscript, *true* or *false*.
- `superScript`. Superscript, *true* or *false*.
- **underline**. Underline, *single, dash, dotted, etc*. See `\PhpOffice\PhpWord\Style\Font::UNDERLINE_...` class constants for possible values

- **lang.** Language, either a language code like *en-US*, *fr-BE*, etc. or an object (or as an array) if you need to set eastAsian or See `\PhpOffice\PhpWord\Style\Language` class for some language codes.
- **position.** The text position, raised or lowered, in half points
- **hidden.** 隐藏文本, *true* or *false*.

6.3 Paragraph 段落

可用段落样式:

- **alignment.** Supports all alignment modes since 1st Edition of ECMA-376 standard up till ISO/IEC 29500:2012. 参见 `\PhpOffice\PhpWord\SimpleType\Jc` 类常量找到可用的值.
- **basedOn.** Parent style.
- **hanging.** Hanging in *twip*.
- **indent.** Indent in *twip*.
- **keepLines.** Keep all lines on one page, *true* or *false*.
- **keepNext.** Keep paragraph with next paragraph, *true* or *false*.
- **lineHeight.** Text line height, e.g. *1.0*, *1.5*, etc.
- **next.** Style for next paragraph.
- **pageBreakBefore.** Start paragraph on next page, *true* or *false*.
- **spaceBefore.** Space before paragraph in *twip*.
- **spaceAfter.** Space after paragraph in *twip*.
- **spacing.** Space between lines in *twip*. If `spacingLineRule` is *auto*, 240 (height of 1 line) will be added, so if you want a double line height, set this to 240.
- **spacingLineRule.** Line Spacing Rule. *auto*, *exact*, *atLeast* See `\PhpOffice\PhpWord\SimpleType\LineSpacingRule` class constants for possible values.
- **suppressAutoHyphens.** Hyphenation for paragraph, *true* or *false*.
- **tabs.** Set of custom tab stops.
- **widowControl.** Allow first/last line to display on a separate page, *true* or *false*.
- **contextualSpacing.** Ignore Spacing Above and Below When Using Identical Styles, *true* or *false*.
- **bidirectional.** Right to Left Paragraph Layout, *true* or *false*.
- **shading.** Paragraph Shading.
- **textAlignment.** Vertical Character Alignment on Line. See `\PhpOffice\PhpWord\SimpleType\TextAlignment` class constants for possible values.

6.4 Table 表格

可用表格样式:

- **alignment. Supports all alignment modes since 1st Edition of ECMA-376 standard up till ISO/IEC 29500:2012.**
See `\PhpOffice\PhpWord\SimpleType\JcTable` and `\PhpOffice\PhpWord\SimpleType\Jc` class constants for possible values.
- `bgColor`. Background color, e.g. '9966CC' .
- `border (Top|Right|Bottom|Left) Color`. Border color, e.g. '9966CC' .
- `border (Top|Right|Bottom|Left) Size`. Border size in *twip*.
- `cellMargin (Top|Right|Bottom|Left)`. Cell margin in *twip*.
- `indent`. Table indent from leading margin. Must be an instance of `\PhpOffice\PhpWord\ComplexType\TblWidth`.
- `width`. Table width in Fiftieths of a Percent or Twentieths of a Point.
- `unit`. The unit to use for the width. One of `\PhpOffice\PhpWord\SimpleType\TblWidth`. Defaults to *auto*.
- `layout`. Table layout, either *fixed* or *autofit* See `\PhpOffice\PhpWord\Style\Table` for constants.
- `cellSpacing` Cell spacing in *twip*
- `position` Floating Table Positioning, see below for options
- `biDiVisual` Present table as Right-To-Left

表格位置浮动样式:

- `leftFromText` Distance From Left of Table to Text in *twip*
- `rightFromText` Distance From Right of Table to Text in *twip*
- `topFromText` Distance From Top of Table to Text in *twip*
- `bottomFromText` Distance From Top of Table to Text in *twip*
- `vertAnchor` Table Vertical Anchor, one of `\PhpOffice\PhpWord\Style\TablePosition::VANCHOR_*`
- `horzAnchor` Table Horizontal Anchor, one of `\PhpOffice\PhpWord\Style\TablePosition::HANCHOR_*`
- `tblpXSpec` Relative Horizontal Alignment From Anchor, one of `\PhpOffice\PhpWord\Style\TablePosition::XALIGN_*`
- `tblpX` Absolute Horizontal Distance From Anchor in *twip*
- `tblpYSpec` Relative Vertical Alignment From Anchor, one of `\PhpOffice\PhpWord\Style\TablePosition::YALIGN_*`

- `tblpY`. Absolute Vertical Distance From Anchorin *twip*

可用行样式选项:

- `cantSplit`. Table row cannot break across pages, *true* or *false*.
- `exactHeight`. Row height is exact or at least.
- `tblHeader`. Repeat table row on every new page, *true* or *false*.

单元格可用演示选项:

- `bgColor`. Background color, e.g. '9966CC' .
- `border (Top|Right|Bottom|Left) Color`. Border color, e.g. '9966CC' .
- `border (Top|Right|Bottom|Left) Size`. Border size in *twip*.
- `gridSpan`. Number of columns spanned.
- **`textDirection (btLr|tbRl)`. Direction of text.** You can use constants `\PhpOffice\PhpWord\Style\Cell::TEXT_DIR_BT_LR` and `\PhpOffice\PhpWord\Style\Cell::TEXT_DIR_TB_RL`
- `valign`. Vertical alignment, *top*, *center*, *both*, *bottom*.
- `vMerge`. *restart* or *continue*.
- `width`. 以 *twip* 为单位的单元格宽度。

6.5 Image 图像

可用图像样式:

- `alignment`. See `\PhpOffice\PhpWord\SimpleType\Jc` class for the details.
- `height`. Height in *pt*.
- `marginLeft`. Left margin in inches, can be negative.
- `marginTop`. Top margin in inches, can be negative.
- `width`. Width in *pt*.
- `wrappingStyle`. Wrapping style, *inline*, *square*, *tight*, *behind*, or *infront*.
- `wrapDistanceTop`. Top text wrapping in pixels.
- `wrapDistanceBottom`. Bottom text wrapping in pixels.
- `wrapDistanceLeft`. Left text wrapping in pixels.
- `wrapDistanceRight`. Right text wrapping in pixels.

6.6 Numbering level 编号级别

可用编号级别样式:

- **alignment.** Supports all alignment modes since 1st Edition of ECMA-376 standard up till ISO/IEC 29500:2012.
See `\PhpOffice\PhpWord\SimpleType\Jc` class constants for possible values.
- `font`. Font name.
- `format`. Numbering format `bullet|decimallupperRoman|lowerRoman|upperLetter|lowerLetter`.
- `hanging`. See paragraph style.
- `hint`. See font style.
- `left`. See paragraph style.
- `restart`. Restart numbering level symbol.
- `start`. Starting value.
- `suffix`. Content between numbering symbol and paragraph text `tab|space|nothing`.
- `tabPos`. See paragraph style.
- `text`. Numbering level text e.g. `%1` for nonbullet or bullet character.

6.7 Chart 图表

可用图表样式选项:

- `width`. Width (in EMU).
- `height`. Height (in EMU).
- `3d`. Is 3D; applies to pie, bar, line, area, *true* or *false*.
- `colors`. A list of colors to use in the chart.
- `title`. The title for the chart.
- `showLegend`. Show legend, *true* or *false*.
- `categoryLabelPosition`. Label position for categories, *nextTo* (default), *low* or *high*.
- `valueLabelPosition`. Label position for values, *nextTo* (default), *low* or *high*.
- `categoryAxisTitle`. The title for the category axis.
- `valueAxisTitle`. The title for the values axis.
- `majorTickMarkPos`. The position for major tick marks, *in*, *out*, *cross*, *none* (default).
- `showAxisLabels`. Show labels for axis, *true* or *false*.

- `gridX`. Show Gridlines for X-Axis, *true* or *false*.
- `gridY`. Show Gridlines for Y-Axis, *true* or *false*.

您可以使用包含的搜索模式 (宏) 创建 OOXML 文档模板，该模板可以替换为您想要的任何值。只能替换单行值。宏的定义如下: `${search-pattern}`。要加载模板文件，请创建模板处理器的新实例。

```
$templateProcessor = new TemplateProcessor('Template.docx');
```

7.1 设单值

给模板包含的一个替换

```
Hello ${firstname} ${lastname}!
```

The following will replace `${firstname}` with John, and `${lastname}` with Doe . The resulting document will now contain Hello John Doe!

```
$templateProcessor->setValue('firstname', 'John');  
$templateProcessor->setValue('lastname', 'Doe');
```

7.2 设置一组值

你还可以通过在数组中传递所有值来设置多个值。

```
$templateProcessor->setValues(array('firstname' => 'John', 'lastname' => 'Doe'));
```

7.3 setImageValue 设置图像值

要寻找的图像模式可能像下面的模式:

- `${search-image-pattern}`
- `${search-image-pattern:[width]:[height]:[ratio]}`
- `${search-image-pattern:[width]x[height]}`
- `${search-image-pattern:size=[width]x[height]}`
- `${search-image-pattern:width=[width]:height=[height]:ratio=false}`

Where:

- `[width]` and `[height]` can be just numbers or numbers with measure, which supported by Word (cm, mm, in, pt, pc, px, %, em, ex)
- `[ratio]` uses only for `false`, `-` or `f` to turn off respect aspect ration of image. By default template image size uses as 'container' size.

Example:

```
${CompanyLogo}
${UserLogo:50:50} ${Name} - ${City} - ${Street}
```

```
$templateProcessor = new TemplateProcessor('Template.docx');
$templateProcessor->setValue('Name', 'John Doe');
$templateProcessor->setValue(array('City', 'Street'), array('Detroit', '12th Street
↳'));

$templateProcessor->setImageValue('CompanyLogo', 'path/to/company/logo.png');
$templateProcessor->setImageValue('UserLogo', array('path' => 'path/to/logo.png',
↳'width' => 100, 'height' => 100, 'ratio' => false));
```


7.4 克隆块

模板中包含参考 `Sample_23_TemplateBlock.php` 为示例。

```

${block_name}
Customer: ${customer_name}
Address: ${customer_address}
${/block_name}

```

以下内容将复制 `${block_name}` 和 `${/block_name}` 之间的所有内容 3 次。

```

$templateProcessor->cloneBlock('block_name', 3, true, true);

```

最后一个参数将重命名块中定义的任何宏，并添加 #1、#2、#3... 到宏名称。结果将是

```

Customer: ${customer_name#1}
Address: ${customer_address#1}

Customer: ${customer_name#2}
Address: ${customer_address#2}

Customer: ${customer_name#3}
Address: ${customer_address#3}

```

也可以传递带有值的数组来替换 macros。如果传递了带有替换项的数组，则忽略 `count` 参数，它是计数数组的大小。

```

$replacements = array(
    array('customer_name' => 'Batman', 'customer_address' => 'Gotham City'),
    array('customer_name' => 'Superman', 'customer_address' => 'Metropolis'),
);
$templateProcessor->cloneBlock('block_name', 0, true, false, $replacements);

```

结果将是

```

Customer: Batman
Address: Gotham City

Customer: Superman
Address: Metropolis

```

7.5 replaceBlock 替换块

模板内容包含.. code-block:: clean

```
{block_name} This block content will be replaced {/block_name}
```

以下内容将 `{block_name}` 和 `{/block_name}` 之间的所有内容替换为传递的值。

```
$templateProcessor->replaceBlock('block_name', 'This is the replacement text.');
```

7.6 deleteBlock 删除块

和之前类似，但是删除块

```
$templateProcessor->deleteBlock('block_name');
```

7.7 cloneRow 克隆行

模板文档中克隆表格的一行参考 `Sample_07_TemplateCloneRow.php` 示例.

```
+-----+-----+
| {userId} | {userName} |
|          |-----+
|          | {userAddress} |
+-----+-----+
```

```
$templateProcessor->cloneRow('userId', 2);
```

结果将为

```
+-----+-----+
| {userId#1} | {userName#1} |
|          |-----+
|          | {userAddress#1} |
+-----+-----+
| {userId#2} | {userName#2} |
|          |-----+
|          | {userAddress#2} |
+-----+-----+
```

7.8 cloneRowAndSetValues 克隆并赋值

在由 `$search` 参数标识的表行中查找行，并将其克隆到 ‘\$values’ 中的条目的次数。

```
+-----+-----+
| ${userId} | ${userName} |
|           |-----+
|           | ${userAddress} |
+-----+-----+
```

```
$values = [
    ['userId' => 1, 'userName' => 'Batman', 'userAddress' => 'Gotham City'],
    ['userId' => 2, 'userName' => 'Superman', 'userAddress' => 'Metropolis'],
];
$templateProcessor->cloneRowAndSetValues('userId', );
```

结果将是

```
+---+-----+
| 1 | Batman |
|   |-----+
|   | Gotham City |
+---+-----+
| 2 | Superman |
|   |-----+
|   | Metropolis |
+---+-----+
```

7.9 应用 xsl 样式表

应用传递给页眉部分、页脚部分和主要部分的 XSL 样式表

```
$xslDomDocument = new \DOMDocument();
$xslDomDocument->load('/path/to/my/stylesheet.xsl');
$templateProcessor->applyXslStyleSheet($xslDomDocument);
```

7.10 设置复杂值

将 `{macro}` 删除，并传递复杂类型。参见“Sample_40_TemplateSetComplexValue.php”示例。

```
$inline = new TextRun();
$inline->addText('by a red italic text', array('italic' => true, 'color' => 'red'));
$templateProcessor->setComplexValue('inline', $inline);
```

7.11 设置复杂块

将 `{macro}` 删除，并传递复杂类型。参见 Sample_40_TemplateSetComplexValue.php 示例。

```
$table = new Table(array('borderSize' => 12, 'borderColor' => 'green', 'width' => 6000, 'unit' => TblWidth::TWIP));
$table->addRow();
$table->addCell(150)->addText('Cell A1');
$table->addCell(150)->addText('Cell A2');
$table->addCell(150)->addText('Cell A3');
$table->addRow();
$table->addCell(150)->addText('Cell B1');
$table->addCell(150)->addText('Cell B2');
$table->addCell(150)->addText('Cell B3');
$templateProcessor->setComplexBlock('table', $table);
```

8.1 OOXML

OOXML 文档包由下文件构成。

- _rels/
 - .rels
- docProps/
 - app.xml
 - core.xml
 - custom.xml
- word/
 - rels/
 - * document.rels.xml
 - media/
 - theme/
 - * theme1.xml
 - document.xml
 - fontTable.xml

- numbering.xml
- settings.xml
- styles.xml
- webSettings.xml
- [Content_Types].xml

8.2 OpenDocument

8.2.1 Package

OpenDocument 文档包由下文件构成。

- META-INF/
 - manifest.xml
- Pictures/
- content.xml
- meta.xml
- styles.xml

8.2.2 content.xml

content.xml 的结构如下所示。

- office:document-content
 - office:font-facedecls
 - office:automatic-styles
 - office:body
 - * office:text
 - draw:*
 - office:forms
 - table:table
 - text:list
 - text:numbered-paragraph
 - text:p

- text:table-of-contents
- text:section
- * office:chart
- * office:image
- * office:drawing

8.2.3 styles.xml

styles.xml 的结构如下所示。

- office:document-styles
 - office:styles
 - office:automatic-styles
 - office:master-styles
 - * office:master-page

8.3 RTF

待完成。

8.4 HTML

待完成。

8.5 PDF

待完成。

9.1 创建左浮动图像

使用相对于水平边距和垂直线的绝对定位。

```
$imageStyle = array(
    'width' => 40,
    'height' => 40,
    'wrappingStyle' => 'square',
    'positioning' => 'absolute',
    'posHorizontalRel' => 'margin',
    'posVerticalRel' => 'line',
);
$textrun->addImage('resources/_earth.jpg', $imageStyle);
$textrun->addText($lipsumText);
```

9.2 自动下载生成的文件

使用 `php://output` 作为文件名。

```
$phpWord = new \PhpOffice\PhpWord\PhpWord();
$section = $phpWord->createSection();
$section->addText('Hello World!');
```

(下页继续)

```
$file = 'HelloWorld.docx';
header("Content-Description: File Transfer");
header('Content-Disposition: attachment; filename="' . $file . '"');
header('Content-Type: application/vnd.openxmlformats-officedocument.wordprocessingml.
↳document');
header('Content-Transfer-Encoding: binary');
header('Cache-Control: must-revalidate, post-check=0, pre-check=0');
header('Expires: 0');
$xmlWriter = \PhpOffice\PhpWord\IOFactory::createWriter($phpWord, 'Word2007');
$xmlWriter->save("php://output");
```

9.3 创建编号标题

定义编号样式和标题样式，并将这两种样式(带有 pStyle 和 numStyle)匹配，如下所示。

```
$phpWord->addNumberingStyle(
    'hNum',
    array('type' => 'multilevel', 'levels' => array(
        array('pStyle' => 'Heading1', 'format' => 'decimal', 'text' => '%1'),
        array('pStyle' => 'Heading2', 'format' => 'decimal', 'text' => '%1.%2'),
        array('pStyle' => 'Heading3', 'format' => 'decimal', 'text' => '%1.%2.%3'),
    )
);
$phpWord->addTitleStyle(1, array('size' => 16), array('numStyle' => 'hNum', 'numLevel
↳' => 0));
$phpWord->addTitleStyle(2, array('size' => 14), array('numStyle' => 'hNum', 'numLevel
↳' => 1));
$phpWord->addTitleStyle(3, array('size' => 12), array('numStyle' => 'hNum', 'numLevel
↳' => 2));

$section->addTitle('Heading 1', 1);
$section->addTitle('Heading 2', 2);
$section->addTitle('Heading 3', 3);
```

9.4 在标题中添加链接

将 ‘HeadingN’ 段落样式应用于 TextRun 或 Link。示例代码:

```
$phpWord = new \PhpOffice\PhpWord\PhpWord();
$phpWord->addTitleStyle(1, array('size' => 16, 'bold' => true));
$phpWord->addTitleStyle(2, array('size' => 14, 'bold' => true));
$phpWord->addFontStyle('Link', array('color' => '0000FF', 'underline' => 'single'));

$section = $phpWord->addSection();

// Textrun
$textrun = $section->addTextRun('Heading1');
$textrun->addText('The ');
$textrun->addLink('https://github.com/PHPOffice/PHPWord', 'PHPWord', 'Link');

// Link
$section->addLink('https://github.com/', 'GitHub', 'Link', 'Heading2');
```

9.5 删除 MS Word 标题栏中的 [Compatibility Mode] 文本

使用 `Metadata\Compatibility\setOoxmlVersion(n)` 方法其中 n 为 Office 版本 (14 = Office 2010, 15 = Office 2013)。

```
$phpWord->getCompatibility()->setOoxmlVersion(15);
```


10.1 如何贡献 PHPWord?

- 提高文档 (Sphinx Format)

10.2 这与我在 CodePlex 中找到的 PHPWord 相同吗 ?

不。这个更好, 有很多你不能使用的新功能在 PHPWord 0.6.3 中查找。CodePlex 的发展停止并且切换到 GitHub 以允许更多人群参与。更多最好的, 对吗?

10.3 我一直在完美地运行来自 CodePlex 的 PHPWord, 但是我不能使用来自 GitHub 的最新 PHPWord。为什么 ?

PHPWord 从 0.8 开始需要 PHP 5.3 +, 而 PHPWord 从 CodePlex 开始需要 0.6.3 可以使用 PHP 5.2 运行。我们可以从中获得许多新功能 PHP 5.3, 它已经在 2009 年左右了! 您应该升级您的 PHP 使用 PHPWord 0.8 + 的版本。

CHAPTER 11

Credits

12.1 ISO/IEC 29500, Third edition, 2012-09-01

- Part 1: Fundamentals and Markup Language Reference
- Part 2: Open Packaging Conventions
- Part 3: Markup Compatibility and Extensibility
- Part 4: Transitional Migration Features

12.2 正式规格

- Oasis OpenDocument Standard Version 1.2
- Rich Text Format (RTF) Specification, version 1.9.1

12.3 其他资源

- `DocumentFormat.OpenXml.Wordprocessing` Namespace on MSDN

CHAPTER 13

Indices and tables

- `genindex`
- `modindex`
- `search`