



From phpdocx v3.7 to v5. Upgrading quick guide.

Document generated on: 2015-10-19

For technical support issues or questions, please visit:

<http://www.phpdocx.com/documentation>

For other inquiries, please contact details below:

contact@phpdocx.com

2mdc

C/ Matías Turrión 24, 1º

Cp 28043 Madrid, Spain

© Copyright 2015 Narcea Producciones Multimedia. All rights reserved.

Table of contents

Abstract	3
New Features	4
Samples and tips to upgrade from phpdocx 3 to 5	5
Equivalency guide	8
For more information	9

Abstract

The release of **phpdocx 5** was a big step forward regarding the functionality of the library. Albeit all its changes, the upgrade from **phpdocx 3** to **phpdocx 5** is really easy: you just need to change a few lines of the existing scripts to take advantage of the new features, classes and methods.

This quick guide is the main reference to know why and how to do the upgrading.

New Features

1. **Completely refactored CreateDocx** class and a completely new CreateDocxFromTemplate class that allows full control of documents based on custom templates.
2. **New WordFragment class** that greatly simplifies the process of nesting content within the Word document.
3. **A new method to replace a placeholder** by arbitrarily complex Word fragments in the main document, headers and footers.
4. **The MultiMerge class** that allows to merge an arbitrarily large number of Word documents with just one line of code. This class supports PDF documents too.
5. **Support for namespaces.** It greatly simplifies the integration with frameworks, CMS and other libraries by using namespaces as Symfony 2, Drupal 8, Laravel, Yii and so on.
6. **Both classic and namespaces packages** are available for the Corporate and Enterprise users.
7. **A new conversion plugin** that uses LibreOffice as the transformation tool, which doesn't need OdfConverter to transform the documents. This new conversion plugin allows to export several types of charts generates the TOC content and returns a better PDF output than using the OpenOffice plugin.
8. **Plenty of plugins for using phpdocx with CMS and frameworks** such as Drupal, WordPress, Symfony and any other framework or development that runs with Composer.
9. **The new XMLAPI** to work with **phpdocx** without PHP knowledge, based on the XML markup language.

Samples and tips to upgrade from **phpdocx 3** to **5**

The main differences when upgrading from **phpdocx 3** to **phpdocx 5** are the new classes `CreateDocxFromTemplate` and `WordFragment`. Both classes allow to easily call the already existing methods and use the new features the new versions include.

The `CreateDocxFromTemplate` class incorporates the methods to work with templates. Instead of the method `addTemplateVariable` of **phpdocx 3**, the new **phpdocx 5** version has a method for each kind of replacement:

- `replaceListVariable` (<http://www.phpdocx.com/api-documentation/templates/replace-list-variable-Word-document>)
- `replaceTableVariable` (<http://www.phpdocx.com/api-documentation/templates/replace-table-variable-Word-document>) and
- `replaceVariableByText` (<http://www.phpdocx.com/api-documentation/templates/replace-variable-text-Word-document>).

phpdocx 5 doesn't include the method `addTemplate` anymore. You have to instantiate the `CreateDocxFromTemplate` class to set the DOCX template.

For example, check this **phpdocx 3** code:

```
require_once 'classes/CreateDocx.inc';

$docx = new CreateDocx();

$docx->addTemplate('template.docx');

$docx->addTemplateVariable('NAME', 'John Smith');

$docx->createDocx('template_text.docx');
```

You have to write that code for **phpdocx 5** as:

```
require_once 'CreateDocx.inc';

$docx = new CreateDocxFromTemplate('template.docx');

$docx->replaceVariableByText(array('NAME' => 'John
Smith'));

$docx->createDocx('output');
```

From phpdocx v3.7 to v5. Upgrading quick guide.

To replace a placeholder by a complex content with the new method `replaceVariableByWordFragment` (<http://www.phpdocx.com/api-documentation/templates/replace-variable-word-fragment-Word-document>) is really simple. E.g.: You can try it with content with texts, links, charts, tables, lists and many others.

The other class mentioned above, `WordFragment`, replaces `createWordMLFragment`. The new class is more flexible and powerful than the previous method.

`WordFragment` needs to reference a new document or a template, where you'll add almost any Word content.

E.g.: This code creates an image and a link, adds them into a text array and then inserts all that content into the document:

```
require_once 'classes/CreateDocx.inc';

$docx = new CreateDocx();

$text = new WordFragment($docx);

$imageOptions = array(
    'src' => 'image.png',
    'scaling' => 50,
    'float' => 'right',
    'textWrap' => 1
);

$image->addImage($imageOptions);

$link = new WordFragment($docx);

$linkOptions = array('url'=> 'http://www.google.es',
    'color' => '0000FF',
    'underline' => 'single');

$link->addLink('link to Google', $linkOptions);

$text = array();
```

From phpdocx v3.7 to v5. Upgrading quick guide.

```
$text[] = $image;  
$text[] = $link;  
$docx->addText($text);  
$docx->createDocx('output');
```

Equivalency guide

This is a table to quickly see the main changes of **phpdocx 3** and which class or method offers the same functionality in **phpdocx 5**.

phpdocx 3	phpdocx 5
addFooter	The same method name, but adding a WordFragment object instead of using the createWordMLFragment method.
addHeader	The same method name, but adding a WordFragment object instead of using the createWordMLFragment method.
addBasicHTML	Removed. Use the embedHtml method to add HTML content.
addDocx	Replaced by addExternalFile.
addElement	Removed. Use the WordFragment class.
addHTML	Replaced by addExternalFile. Or use the embedHtml method.
addMathDocx	Replaced by addMathEquation.
addMathEq	Replaced by addMathEquation.
addMathML	Replaced by addMathEquation.
addMHT	Replaced by addExternalFile.
addParagraph	Removed. Use the addText method
addRTF	Replaced by addExternalFile.
createWordMLFragment	Removed. Use the WordFragment class.
addTemplate	Removed. Use the CreateDocxFromTemplate class.
addTemplateChart	Removed. Use the eplaveVariableByWordMethod method.
addTemplateImage	Removed. Use the replaceVariableByWordMethod method.
addTemplateVariable	Removed. Use one of these methods: replaceListVariable, replaceTableVariable, replaceVariableByText or replaceVariableByExternalFile.
cleanTemplateVariable	Removed. Use the removeTemplateVariable emthod.
replaceHeaderImage	Removed. Use removePlaceholder image method.
replaceTemplateVariableByHTML	Renamed to replaceVariableByHtml.
generatePDF	Removed. Use the conversion plugin.
generateXHTML	Removed. Use the conversion plugin.

From phpdocx v3.7 to v5. Upgrading quick guide.

For more information

The release of **phpdocx 5** brought new documentation that has improved in order to be useful to beginner and advanced users.

On the following links you may find the **phpdocx** introduction, a new practical guide, all necessary documentation to work with the conversion plugin and the full API documentation:

<http://www.phpdocx.com/documentation/introduction>

<http://www.phpdocx.com/documentation/practical>

<http://www.phpdocx.com/documentation/conversion-plugin>

<http://www.phpdocx.com/documentation/api-documentation>