
TwigExcelBundle

Release

May 25, 2017

Contents

1	User Documentation	1
1.1	Installation	1
1.2	Getting started	2
1.3	Examples	3
1.4	Twig Functions	3
1.5	Twig Tags	4

Installation

Step 1: Download the Bundle

Open a command console, enter your project directory and execute the following command to download the latest stable version of this bundle:

```
$ composer require mewesk/twig-excel-bundle
```

Or add the following code to your `composer.json` file and run `composer update` afterwards:

```
{
  "require": {
    "mewesk/twig-excel-bundle": "1.3.*@dev"
  }
}
```

```
$ php composer.phar update mewesk/twig-excel-bundle
```

This requires you to have Composer installed globally, as explained in the [installation chapter](<https://getcomposer.org/doc/00-intro.md>) of the Composer documentation.

Step 2: Enable the Bundle

```
<?php
// app/AppKernel.php

// ...
class AppKernel extends Kernel
{
    public function registerBundles()
```

```
{
    $bundles = array(
        // ...

        new MewesK\TwigExcelBundle\MewesKTwigExcelBundle(),
    );

    // ...
}

// ...
}
```

Step 3: Configure the Bundle (optional)

Add the following configuration to your `config.yml` if you don't want to pre-calculate formulas. Disabling this option can improve the performance but the resulting documents won't show the result of any formulas when opened in an external spreadsheet software.

```
mewes_k_twig_excel:
    pre_calculate_formulas: false
```

Add the following configuration to your `config.yml` if you want to enable disk caching. Using disk caching can improve memory consumption by writing data to disk temporarily. Works only for `.XLSX` and `.ODS` documents.

```
mewes_k_twig_excel:
    disk_caching_directory: "%kernel.cache.dir%/phpexcel"
```

Getting started

Step 1: Create your controller

```
<?php
// src/Acme/HelloBundle/Controller/HelloController.php

namespace Acme\HelloBundle\Controller;

use Sensio\Bundle\FrameworkExtraBundle\Configuration\Route;
use Sensio\Bundle\FrameworkExtraBundle\Configuration\Template;
use Symfony\Component\HttpFoundation\Response;

class HelloController
{
    /**
     * @Route("/hello.{_format}", defaults={"_format"="xls"}, requirements={"_format"=
↪ "csv|xls|xlsx"})
     * @Template("AcmeHelloBundle:Hello:index.xls.twig")
     */
    public function indexAction($name)
    {
        return ['data' => ['La', 'Le', 'Lu']];
    }
}
```

Step 2: Create your template

```
{# src/Acme/HelloBundle/Resources/views/Hello/index.xls.twig #}

{% xlsdocument %}
  {% xlsheet 'Worksheet' %}
    {% xlsrow %}
      {% xlscell { style: { font: { size: '18' } } } %}Values{% endxlscell %}
    {% endxlsrow %}
    {% for value in data %}
      {% xlsrow %}
        {% xlscell %}{{ value }}{% endxlscell %}
      {% endxlsrow %}
    {% endfor %}
  {% endxlsheet %}
{% endxlsdocument %}
```

Examples

For more advanced examples check the unit test scenarios here:

<https://github.com/MewesK/TwigExcelBundle/tree/master/Tests/Resources/views>

Twig Functions

xlsmergestyles

```
xlsmergestyles([style1:array], [style2:array])
```

- Merges two style arrays recursively
- Returns a new array

Parameters

Name	Type	Optional	Description
style1	array		Standard PhpExcel style array
style2	array		Standard PhpExcel style array

Example

```
{% set mergedStyle = xlsmergestyles({ font: { name: 'Verdana' } }, { font: { size: '18' } }) %}
```

Twig Tags

xlsdocument

```
{% xlsdocument [properties:array] %}
...
{% endxlsdocument %}
```

- Must contain one or more 'xsheet' tags

Attributes

Name	Type	Optional	Description
properties	array	X	

Properties

Name	Type	XL\$	OD\$	Description
category	string	X		
company	string			
created	date-time	X	X	Can be null, timestamp or a strtotime compatible string
creator	string	X	X	
default-Style	array			Standard PhpExcel style array
description	string	X	X	
format	string	X		Possible formats are 'csv', 'html', 'pdf', 'xls', 'xlsx'
keywords	string	X		
lastModifiedBy	string	X		
manager	string			
modified	date-time	X	X	Can be null, timestamp or a strtotime compatible string
security	array			Cannot be tested - not supported by the reader
+ lockRevision	boolean			Cannot be tested - not supported by the reader
+ lockStructure	boolean			Cannot be tested - not supported by the reader
+ lockWindows	boolean			Cannot be tested - not supported by the reader
+ revision-Password	string			Cannot be tested - not supported by the reader
+ workbook-Password	string			Cannot be tested - not supported by the reader
subject	string	X	X	
template	string	X	X	The path can be an absolute system path or a Twig namespace prefixed path like '@AppBundle/...' or a custom one like '@Templates/...'. Works for CSV too.
title	string	X	X	

Example

```
{% xlsdocument {
    category: 'Test category',
    company: 'Test company',
    created: '2000/01/01',
    creator: 'Tester',
    defaultStyle: {
        font: {
            name: 'Verdana',
            size: 18.0
        }
    },
    description: 'Test document',
    format: 'xls',
    keywords: 'Test',
    lastModifiedBy: 'Tester',
    manager: 'Tester',
    modified: '2000/01/01',
    security: {
        lockRevision: true,
        lockStructure: true,
        lockWindows: true,
        revisionsPassword: 'test',
        workbookPassword: 'test'
    },
    subject: 'Test',
    title: 'Test'
} %}
{# ... #}
{% endxlsdocument %}
```

xlssheet

```
{% xlssheet [title:string] [properties:array] %}
...
{% endxlssheet %}
```

- May contain one or more ‘xlsheader’, ‘xlsfooter’, ‘xlsrow’ and ‘xlsdrawing’ tags

Attributes

Name	Type	Op-tional	Description
title	string	X	If no title is given the first existing sheet will be used. If no sheet exists a new one will be created.
proper-ties	ar-ray	X	

Properties

Name	Type	XLS	ODS	Description
autoFilter	string			The range like 'A1:E20'
columnDimension	array			Contains one or more arrays. Possible keys are 'default' or a valid column name
+ autoSize	boolean			
+ collapsed	boolean			Does not work in PhpExcel?
+ columnIndex	string			Does not work in PhpExcel?
+ outlineLevel	int			
+ visible	boolean			Does not work in PhpExcel?
+ width	double			
+ xfIndex	int			
pageMargins	array			
+ top	double			
+ bottom	double			
+ left	double			
+ right	double			
+ header	double			
+ footer	double			
pageSetup	array			
+ fitToHeight	int			
+ fitToPage	boolean			
+ fitToWidth	int			
+ horizontalCentered	boolean			
+ orientation	string			Possible orientations are 'default', 'landscape', 'portrait'
+ paperSize	int			Possible values are defined in PHPExcel_Worksheet_PageSetup
+ printArea	string			A range like 'A1:E20'
+ scale	int			
+ verticalCentered	boolean			
protection	array			
+ autoFilter	boolean			
+ deleteColumns	boolean			
+ deleteRows	boolean			
+ formatCells	boolean			
+ formatColumns	boolean			
+ formatRows	boolean			
+ insertColumns	boolean			
+ insertHyperlinks	boolean			
+ insertRows	boolean			
+ objects	boolean			
+ password	string			
+ pivotTables	boolean			
+ scenarios	boolean			
+ selectLockedCells	boolean			
+ selectUnlockedCells	boolean			
+ sheet	boolean			
+ sort	boolean			
printGridlines	boolean			
rightToLeft	boolean			
rowDimension	array			Contains one or more arrays. Possible keys are 'default' or a row index >=1
+ collapsed	boolean			Does not work in PhpExcel?
+ outlineLevel	int			
+ rowHeight	double			

Continued on next page

Table 1.1 – continued from previous page

Name	Type	XLS	ODS	Description
+ rowIndex	int			Does not work in PhpExcel?
+ visible	boolean			Does not work in PhpExcel?
+ xfIndex	int			
+ zeroHeight	boolean			Does not work in PhpExcel?
sheetState	string			
showGridlines	boolean			Cannot be tested - not supported by the reader
tabColor	string			
zoomScale	int			

Example

```
{% xlsxheet 'Worksheet' {
    columnDimension: {
        'default': {
            autoSize: false,
            collapsed: false,
            outlineLevel: 0,
            visible: true,
            width: -1,
            xfIndex: 0
        },
        'D': {
            columnIndex: 2,
            visible: false
        }
    },
    pageMargins: {
        top: 1,
        bottom: 1,
        left: 0.75,
        right: 0.75,
        header: 0.5,
        footer: 0.5
    },
    pageSetup: {
        fitToHeight: 1,
        fitToPage: false,
        fitToWidth: 1,
        horizontalCentered: false,
        orientation: 'landscape',
        paperSize: 9,
        printArea: 'A1:B1',
        scale: 100,
        verticalCentered: false
    },
    protection: {
        autoFilter: true,
        deleteColumns: true,
        deleteRows: true,
        formatCells: true,
        formatColumns: true,
        formatRows: true,
        insertColumns: true,
        insertHyperlinks: true,
```

```

        insertRows: true,
        objects: true,
        pivotTables: true,
        scenarios: true,
        selectLockedCells: true,
        selectUnlockedCells: true,
        sheet: true,
        sort: true
    },
    printGridlines: true,
    rightToLeft: false,
    rowDimension: {
        'default': {
            collapsed: false,
            outlineLevel: 0,
            rowHeight: -1,
            rowIndex: '1',
            visible: true,
            xfIndex: 0,
            zeroHeight: false
        },
        '2': {
            visible: false
        }
    },
    sheetState: 'visible',
    showGridlines: true,
    tabColor: 'c0c0c0',
    zoomScale: 75
}%;
    {# ... #}
{% endxlsxsheet %}

```

xlshheader

```

{% xlsxheader [type:string] [properties:array] %}
    ...
{% endxlsxheader %}

```

- May contain one 'xlsleft', 'xlscenter' and 'xlsright' tag
- Not supported by the OpenDocument writer

Attributes

Name	Type	Optional	Description
type	string	X	Possible types are 'header' (default), 'oddHeader' (xlsx), 'evenHeader' (xlsx), 'firstHeader' (xlsx)
properties	array	X	

Properties

Name	Type	XLS	ODS	Description
scaleWithDocument	boolean			
alignWithMargins	boolean			

Example

```
{% xlsheader 'firstHeader' %}
  {# ... #}
{% endxlsheader %}
```

xlsfooter

```
{% xlsfooter [type:string] [properties:array] %}
  ...
{% endxlsfooter %}
```

- May contain one 'xlsleft', 'xlscenter' and 'xlsright' tag
- Not supported by the OpenDocument writer

Attributes

Name	Type	Optional	Description
type	string	X	Possible types are 'footer' (default), 'oddFooter' (xlsx), 'evenFooter' (xlsx), 'firstFooter' (xlsx)
properties	array	X	

Properties

Name	Type	XLS	ODS	Description
scaleWithDocument	boolean			
alignWithMargins	boolean			

Example

```
{% xlsfooter 'firstFooter' %}
  {# ... #}
{% endxlsfooter %}
```

xlsleft, xlscenter, xlsright

```
{% xlsleft %}
...
{% endxlsleft %}

{% xlscenter %}
...
{% endxlscenter %}

{% xlsright %}
...
{% endxlsright %}
```

- May contain one 'xlsdrawing' tag (not supported by the Excel5 and OpenDocument writer)
- Not supported by the OpenDocument writer
- These tags replace the &L, &C and &R format codes.
- All other codes can be found at: <https://github.com/PHPOffice/PHPExcel/blob/develop/Documentation/markdown/Overview/08-Recipes.md#setting-the-print-header-and-footer-of-a-worksheet>

Example

```
{% xlsheader %}
  {% xlsleft %}
    Left part of the header
  {% endxlsleft %}
  {% xlscenter %}
    Center part of the header
  {% endxlscenter %}
  {% xlsright %}
    Right part of the header
  {% endxlsright %}
{% endxlsheader %}
```

xlsrow

```
{% xlsrow [index:int] %}
...
{% endxlsrow %}
```

- May contain one or more 'xlscell' tags
- If 'index' is not defined it will default to 1 for the first usage per sheet
- For each further usage it will increase the index by 1 automatically (1, 2, 3, ...)

Attributes

Name	Type	Optional	Description
index	int		A row index >=1

Example

```
{% xlsrow 1 %}
    {# ... #}
{% endxlsrow %}
```

xlscell

```
{% xlscell [index:string] [properties:array] %}
    ...
{% endxlscell %}
```

- If 'index' is not defined it will default to 0 for the first usage per row
- For each further usage it will increase the index by 1 automatically (0, 1, 2, ...)
- Formulas are supported (e.g. =SUM(A1:F1) or =A1+B1)

Attributes

Name	Type	Optional	Description
index	int		A column index >=0
properties	array	X	

Properties

Name	Type	XLS	ODS	Description
break	int	X		Possible values are defined in PHPExcel_Worksheet
dataType	string	X	X	Possible values are defined in PHPExcel_Cell_DataType
dataValidation	array			
+ allowBlank	boolean			
+ error	string			
+ errorStyle	string			Possible values are defined in PHPExcel_Cell_DataValidation
+ errorTitle	string			
+ formula1	string			
+ formula2	string			
+ operator	string			Possible values are defined in PHPExcel_Cell_DataValidation
+ prompt	string			
+ promptTitle	string			
+ showDropDown	boolean			
+ showErrorMessage	boolean			
+ showInputMessage	boolean			
+ type	string			Possible values are defined in PHPExcel_Cell_DataValidation
explicitValue	boolean	X		If true cell is rendered as an explicit value (useful to show leading zeros using strings)
merge	int string	X		Merge a cell range. Allows zero-based cell index or cell coordinates like 'A3'
style	array	X		Standard PhpExcel style array
url	string	X		

Example

```
{% xlscell 0 {
    break: 1,
    dataValidation: {
        allowBlank: false,
        error: '',
        errorStyle: 'stop',
        errorTitle: '',
        formula1: '',
        formula2: '',
        operator: '',
        prompt: ''
        promptTitle: '',
        showDropDown: false,
        showErrorMessage: false,
        showInputMessage: false,
        type: 'none',
    },
    merge: 2,
    style: {
        borders: {
            bottom: {
                style: 'thin',
                color: {
```



```

        rgb: '000000'
    }
}
},
url: 'http://www.example.com'
} %}
{# ... #}
{% endxlsxcell %}

```

xlsdrawing

```
{% xlsdrawing [path:string] [properties:array] %}
```

- If the xlsdrawing is used in a header/footer it automatically adds the &G code to be displayed
- Not supported by the OpenDocument writer

Attributes

Name	Type	Optional	Description
path	string		
properties	array	X	

Properties

Name	Type	XLS	ODS	Description
coordinates	string	X		Cell coordinates like 'A1'
description	string			
height	int	X		
name	string			
offsetX	int			
offsetY	int			
resizeProportional	boolean	X		
rotation	int			
shadow	array			
+ alignment	string			Possible values are defined in PHPExcel_Worksheet_Drawing_Shadow
+ alpha	int			
+ blurRadius	int			
+ color	string			A hexadecimal color string like '000000' (without #)
+ direction	int			
+ distance	int			
+ visible	boolean			
width	int	X		

Example

```
{% xlsdrawing '/test.png' {
    coordinates: 'A1',
    description: 'Test',
    height: 0,
    name: '',
    offsetX: 0,
    offsetY: 0,
    resizeProportional: true,
    rotation: 0,
    shadow: {
        alignment: 'br',
        alpha: 50,
        blurRadius: 6,
        color: '000000',
        direction: 0,
        distance: 2,
        visible: false
    },
    width: 0
} %}
```

xlsblock

```
{% xlsblock ... %}
    {# ... #}
{% endxlsblock %}
```

- Works the same way as the standard block tag does
- Attention: the standard block tag does not with tags provided by this bundle

Example

```
{# parent.xls.twig #}
```

```
{% xlsblock test %}
    {# ... #}
{% endxlsblock %}
```

```
{# child.xls.twig #}
```

```
{% extends 'parent.xls.twig' %}
```

```
{% xlsblock test %}
    {# ... #}
{% endxlsblock %}
```

xlsmacro

```
{% xlsmacro ... %}
    {# ... #}
{% endxlsmacro %}
```

- Works the same way as the standard macro tag does
- Attention: the standard macro tag does not work with tags provided by this bundle

Example

```
{# macros.xls.twig #}

{% xlsmacro test() %}
    {# ... #}
{% endxlsmacro %}
```

```
{# test.xls.twig #}

{% import 'macros.xls.twig' as macros %}

{# ... #}
{{ macros.test() }}
{# ... #}
```

xlsinclude

```
{% xlsinclude %}
    {# ... #}
{% endxlsinclude %}
```

- Attention: without this wrapper tag templates containing tags provided by this bundle can not be used

Example

```
{# include.xls.twig #}

{% xlsinclude %}
    {# ... #}
{% endxlsinclude %}
```

```
{# test.xls.twig #}

{% include 'include.xls.twig' %}
```