
TwigSpreadsheetBundle

Jul 29, 2018

Contents

1	Installation	1
1.1	Step 1: Download the Bundle	1
1.2	Step 2: Enable the Bundle	1
2	Getting started	3
2.1	Step 1: Create your controller	3
2.2	Step 2: Create your template	3
3	Configuration	5
4	Examples	7
4.1	Formulas	7
4.2	Templates	7
4.3	More examples	8
5	Twig Functions	9
5.1	xlsmergestyles	9
5.2	xlscellindex	9
5.3	xlsrowindex	10
6	Twig Tags	11
6.1	xlsdocument	11
6.2	xlssheet	13
6.3	xlsheader, xlsfooter	16
6.4	xlsleft, xlscenter, xlsright	17
6.5	xlsrow	18
6.6	xlscell	18
6.7	xlsdrawing	20

1.1 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-spreadsheet-bundle
```

This command requires you to have Composer installed globally, as explained in the ‘**installation chapter**’ of the Composer documentation.

1.2 Step 2: Enable the Bundle

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

// ...
class AppKernel extends Kernel
{
    public function registerBundles()
    {
        $bundles = array(
            // ...

            new MewesK\TwigSpreadsheetBundle\MewesKTwigSpreadsheetBundle(),
        );

        // ...
    }

    // ...
}
```


CHAPTER 2

Getting started

2.1 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:spreadsheet.twig")
     */
    public function indexAction($name)
    {
        return ['data' => ['F', 'o', 'o']];
    }
}
```

2.2 Step 2: Create your template

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

{% xlsdocument %}
    {% xlssheet 'Worksheet' %}
```

(continues on next page)

(continued from previous page)

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

CHAPTER 3

Configuration

```
# Default configuration for "MewesKTwigSpreadsheetBundle"
mewes_k_twig_spreadsheet:

    # Disabling formula calculations can improve the performance but the resulting_
↪documents
    # won't immediately show formula results in external programs.
    pre_calculate_formulas: true

    cache:

        # Using a bitmap cache is necessary, PhpSpreadsheet supports only local files.
        bitmap:                "%kernel.cache_dir%/spreadsheet/bitmap"

        # Using XML caching can improve memory consumption by writing data to disk.
        # Works only for .xlsx and .ods documents.
        xml:                    false # Example: "%kernel.cache_dir%/spreadsheet/xml"

    # See PhpOffice\PhpSpreadsheet\Writer\Csv.php for more information.
    csv_writer:
        delimiter:              ",",
        enclosure:               "\"\"",
        excel_compatibility:     false
        include_separator_line:  false
        line_ending:             "\n" # Depends on the OS since it is actually using_
↪PHP_EOL
        sheet_index:            0
        use_bom:                 false
```


4.1 Formulas

```
{% xlsdocument %}
  {% xlssheet %}
    {% xlsrow %}
      {% xlscell %}667.5{% endxlscell %}{# A1 #}
      {% xlscell %}2{% endxlscell %}{# B1 #}
    {% endxlsrow %}
    {% xlsrow %}
      {% xlscell %}=A1*B1+2{% endxlscell %}
    {% endxlsrow %}
    {% xlsrow %}
      {% xlscell %}=SUM(A1:B1){% endxlscell %}
    {% endxlsrow %}
  {% endxlssheet %}
{% endxlsdocument %}
```

Note: It can be faster to calculate formulas in the template or the controller.

4.2 Templates

```
{% xlsdocument { 'template': '@Hello/templates/template.xlsx' } %}
  {% xlssheet %}
    {# ... #}
  {% endxlssheet %}
{% endxlsdocument %}
```

Note: If you want to save your templates outside of your *views* folder consider using your own Twig `namespace`.

4.3 More examples

For more advanced examples check the unit test scenarios here:

<https://github.com/MewesK/TwigSpreadsheetBundle/tree/master/tests/Twig/Fixtures/views>

5.1 xlsmergestyles

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

- Merges two style arrays recursively
- Returns a new array

5.1.1 Parameters

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

5.1.2 Example

```
{% set mergedStyle = xlsmergestyles({ font: { name: 'Verdana' } }, { font: { size: 18.  
↪0 } }) %}
```

5.2 xlscellindex

```
xlscellindex()
```

- Returns the current cell index or null if no cell is initialized

5.2.1 Example

```
{% xlsrow %}
  {% set cellIndex = xlscellindex() %}
  {% xlscell %}
    {# cell index is null, because it was read before the first cell was_
↪initialized #}
    {{ cellIndex }}
    {# cell index is 1, because it was read after the first cell was initialized
↪#}
    {{ xlscellindex() }}
  {% endxlscell %}
{% endxlsrow %}
```

5.3 xlsrowindex

```
xlsrowindex()
```

- Returns the current row index or null if no row is initialized

5.3.1 Example

```
{% xlssheet 'Test' %}
  {% set rowIndex = xlsrowindex() %}
  {% xlsrow %}
    {% xlscell %}
      {# row index is null, because it was read before the first row was_
↪initialized #}
      {{ rowIndex }}
      {# row index is 1, because it was read after the first row was_
↪initialized #}
      {{ xlsrowindex() }}
    {% endxlscell %}
  {% endxlsrow %}
{% endxlssheet %}
```

6.1 xlsdocument

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

- Must contain one or more ‘xlssheet’ tags

6.1.1 Parameters

Name	Type	Optional	Description
properties	array	X	

6.1.2 Properties

Name	Type	XLS	ODS	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 PhpSpreadsheet 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
+ workbookPassword	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	

6.1.3 Example

```
{% xlsdocument {
    category: 'Test category',
    company: 'Test company',
    created: '2000/01/01',
    creator: 'Tester',
    defaultStyle: {
        font: {
            name: 'Verdana',
            size: 18.0
        }
    },
    description: 'Test document',
```

(continues on next page)

(continued from previous page)

```

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 %}

```

6.2 xlsxsheet

```

{% xlsxsheet [title:string] [properties:array] %}
...
{% endxlsxsheet %}

```

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

6.2.1 Parameters

Name	Type	Optional	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.
properties	array	X	

6.2.2 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 PhpSpreadsheet?
+ columnIndex	string			Does not work in PhpSpreadsheet?
+ outlineLevel	int			
+ visible	boolean			Does not work in PhpSpreadsheet?
+ width	double			
+ xfIndex	int			

Continued on next page

Table 1 – continued from previous page

Name	Type	XLS	ODS	Description
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 <code>PhpOfficePhpSpreadsheetWorksheetPageSetup</code>
+ 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 PhpSpreadsheet?
+ outlineLevel	int			
+ rowHeight	double			
+ rowIndex	int			Does not work in PhpSpreadsheet?
+ visible	boolean			Does not work in PhpSpreadsheet?
+ xfIndex	int			
+ zeroHeight	boolean			Does not work in PhpSpreadsheet?
sheetState	string			
showGridlines	boolean			Cannot be tested - not supported by the reader
tabColor	string			
zoomScale	int			

6.2.3 Example

```
{% xlsxsheet '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: {
```

(continues on next page)

(continued from previous page)

```

        '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
} %}
{# ... #}
{% endxlsheet %}

```

6.3 xlsheader, xlsfooter

```

{% xlsheader [type:string] [properties:array] %}
...
{% endxlsheader %}

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

```

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

6.3.1 Parameters

Name	Type	Optional	Description
type	string	X	Possible types are null (default), 'odd' (xlsx), 'even' (xlsx), 'first' (xlsx)
properties	array	X	

6.3.2 Properties

Name	Type	XLS	ODS	Description
scaleWithDocument	boolean			
alignWithMargins	boolean			

6.3.3 Example

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

{% xlsheader 'odd' %}
    {# ... #}
{% endxlsheader %}

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

6.4 xlsleft, xlscenter, xlsright

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

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

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

- May contain one 'xlsdrawing' tag (not supported by the XLS and ODS writer)
- Not supported by the ODS writer
- These tags replace the &L, &C and &R format codes. All other codes can be found in `PhpOfficePhpSpreadsheetWorksheetHeaderFooter`

6.4.1 Example

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

6.5 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, ...)

6.5.1 Parameters

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

6.5.2 Example

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

6.6 xlscell

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

- If ‘index’ is not defined it will default to 1 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)

6.6.1 Parameters

Name	Type	Optional	Description
index	int	X	A column index >=1
properties	array	X	

6.6.2 Properties

Name	Type	XLS	ODS	Description
break	int	X		Possible values are defined in <code>PhpOfficePhpSpreadsheetSpreadsheet</code>
dataType	string	X	X	If set cell is rendered as an explicit value (prevents PHP type casting). Possible values are defined in <code>PhpOfficePhpSpreadsheetCellDataType</code>
dataValidation	array			
+ allowBlank	boolean			
+ error	string			
+ errorStyle	string			Does not work in <code>PhpSpreadsheet</code> ? Possible values are defined in <code>PhpOfficePhpSpreadsheetCellDataValidation</code>
+ errorTitle	string			
+ formula1	string			
+ formula2	string			
+ operator	string			Possible values are defined in <code>PhpOfficePhpSpreadsheetCellDataValidation</code>
+ prompt	string			
+ promptTitle	string			
+ showDropDown	boolean			
+ showErrorMessage	boolean			
+ showInputMessage	boolean			
+ type	string			Does not work in <code>PhpSpreadsheet</code> ? Possible values are defined in <code>PhpOfficePhpSpreadsheetCellDataValidation</code>
merge	int string	X		Merge a cell range. Allows cell index ≥ 1 or cell coordinates like 'A3'
style	array	X		Standard <code>PhpSpreadsheet</code> style array
url	string	X		

6.6.3 Example

```
{% xlsxcell 1 {
    break: 1,
    dataValidation: {
        allowBlank: false,
        error: '',
        errorStyle: 'stop',
        errorTitle: '',
        formula1: '',
        formula2: '',
        operator: '',
        prompt: ''
        promptTitle: '',
        showDropDown: false,
        showErrorMessage: false,
        showInputMessage: false,
        type: 'none',
    }
}
```

(continues on next page)

(continued from previous page)

```
    },
    merge: 3,
    style: {
        borders: {
            bottom: {
                style: 'thin',
                color: {
                    rgb: '000000'
                }
            }
        }
    },
    url: 'http://www.example.com'
} %}
{# ... #}
{% endxlscell %}
```

6.7 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

6.7.1 Parameters

Name	Type	Optional	Description
path	string		
properties	array	X	

6.7.2 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 <code>PhpOfficePhpSpreadsheetWorksheet-DrawingShadow</code>
+ alpha	int			
+ blurRadius	int			
+ color	string			A hexadecimal color string like '000000' (without #)
+ direction	int			
+ distance	int			
+ visible	boolean			
width	int	X		

6.7.3 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
} %}
```