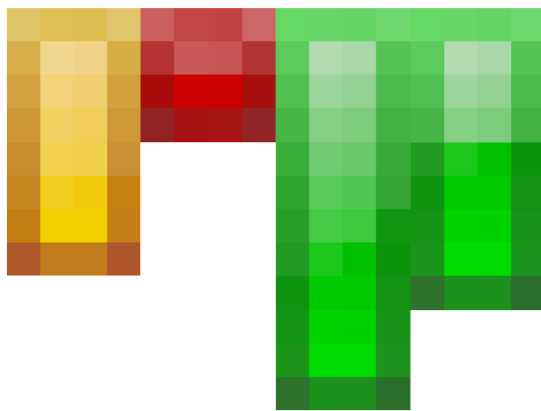# Auto grouping sitemap and menu

Version: 1.0.0, 2012-12-27

Extension Key: autositemap

Language: en

Keywords: sitemap, menu, grouping

Copyright 2012, Dirk Wildt, Die Netzmacher, <http://wildt.at.die-netzmacher.de>

This document is published under the Open Content License

available from http://www.opencontent.org/opl.shtml

The content of this document is related to TYPO3

- a GNU/GPL CMS/Framework available from www.typo3.org

Table of Contents

Screen Shots

Frontend

Illustration 1: Two main columns, one marginal column

Illustration 2: Four main columns, one marginal column

Illustration 3: Three main columns, one marginal column

Illustration 4: Four main columns, one marginal column

Backend

Illustration 5: TypoScript Constant Editor

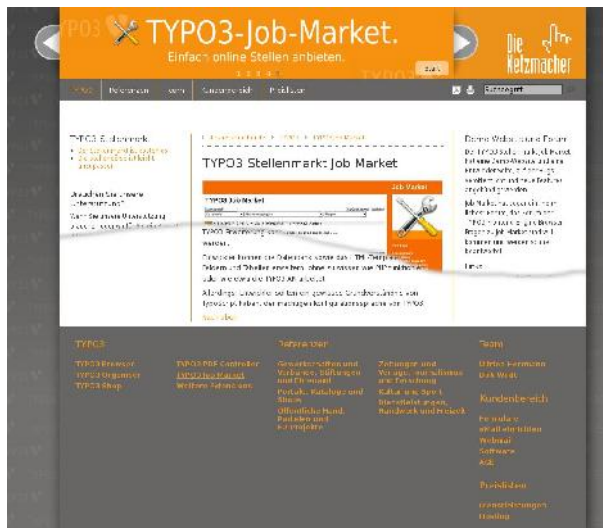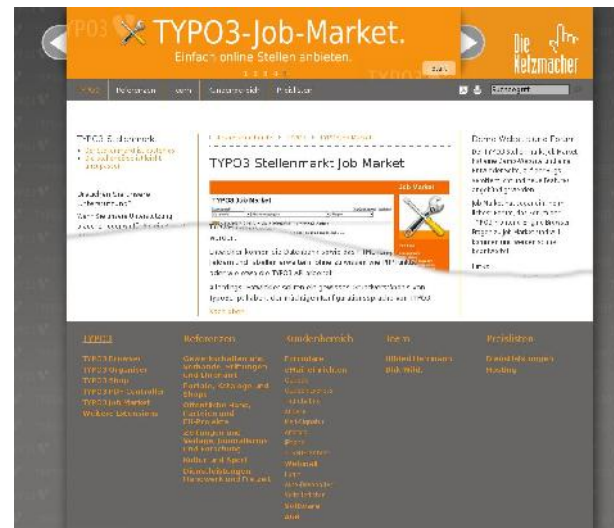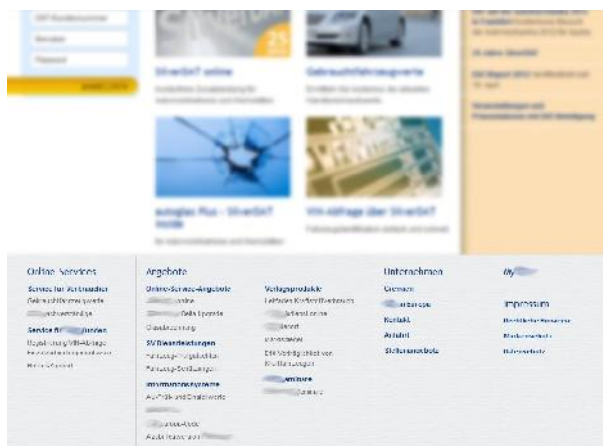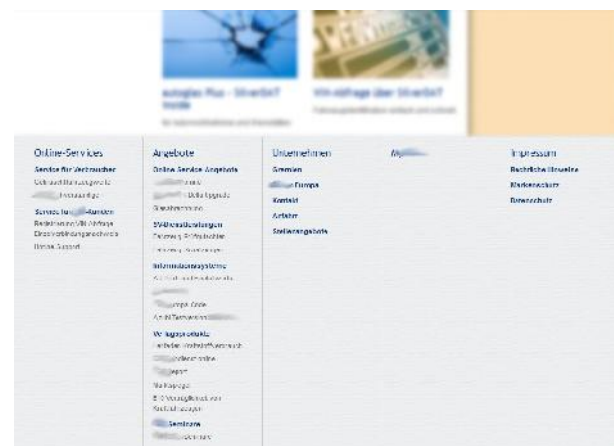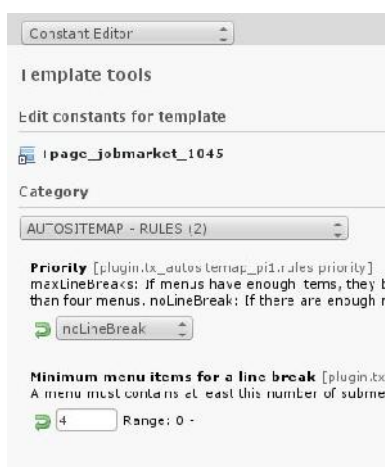## Layout Samples

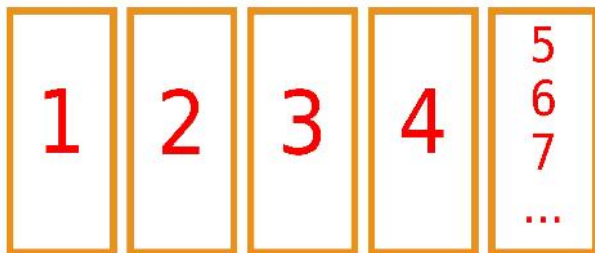Four examples. Ten layouts are possible.

Illustration 6: Four main columns, one marginal column
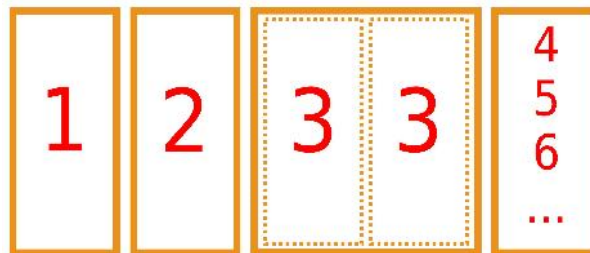
Illustration 7: Three main columns, one marginal column
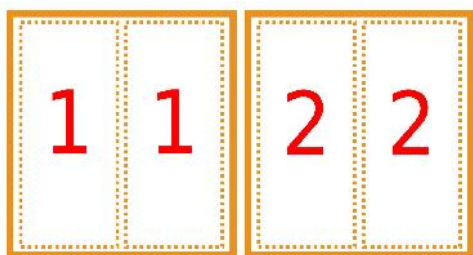
Illustration 8: Two main columns
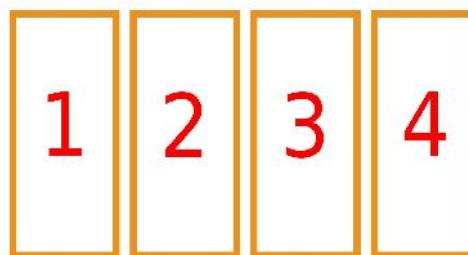
Illustration 9: Four main columns

Introduction

What does it do?

- autositemap generates a sitemap or menu in columns.

- autositemap tries to split menus for the best design. Ten designs are possible.

- For example: Big menus can get two columns.

- If you have more than four menus, the rest will placed in a fifth column.

- The menus in the first four columns has a depth of three levels by default.

- The menus in the fifth column has a depth of two levels by default.

- You can control the behaviour of placements with one of three priorities:

    - Maximum of line breaks

    - One line break only

    - No line break for any menu

    See "Priority" on page 7 below for more details.

- The menus are configured by TypoScript. They are based on the HMENU and TMENU. But you can't use all HMENU properties!

- The used HTML classes by the include static template "+Autositemap YAML (autositemap)" are based on the YAML CSS. But the JavaScript framework YAML isn't needed.

Installation

One page only

- Install the extension.

- Add the plugin to a page, where you like to use the autositemap.

- Add the TypoScript templates to the static templates of the TypoScript of this page:

  - Autositemap (autositemap)

  - +Autositemap YAML (autositemap)

- Configure autositemap supported by the TypoScript Constant Editor.
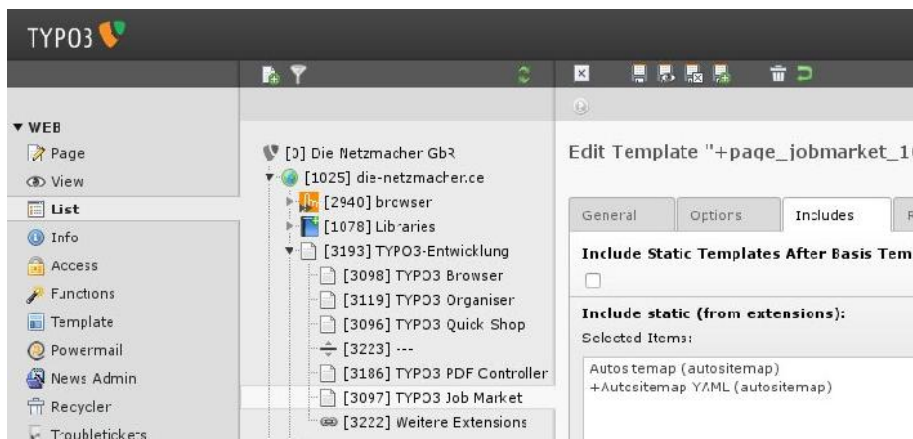
Illustration 10: Include static (from extensions)

Website wide Installation

- Install the extension.

- Add a sysfolder "libraries" to your website.

- Add a page "sitemap" into the page "libraries".

- Add the plugin into the page "sitemap"

- Add the TypoScript templates to the static templates of the TypoScript of your root page:

  - Autositemap (autositemap)

  - +Autositemap YAML (autositemap)

- Include the content of the page "sitemap" website wide by TypoScript or TemplaVoilá.

- Configure autositemap supported by the TypoScript Constant Editor.
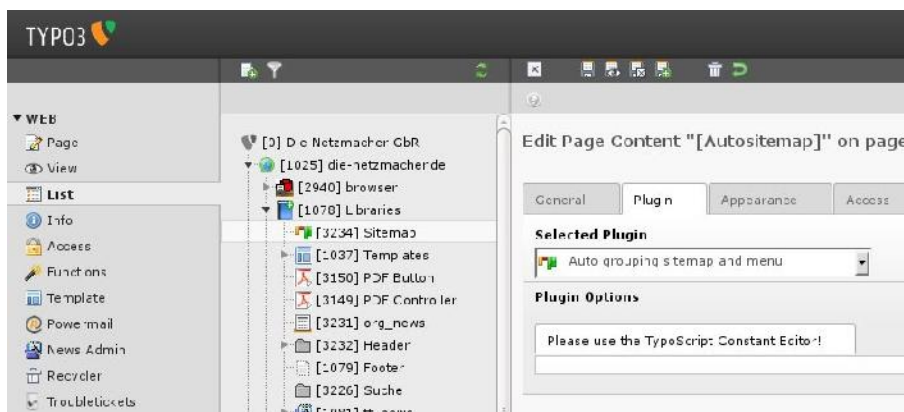
Illustration 11: The autositemap plugin

Priority

First of all

The marginal Column

The fifth column is a marginal column. It contains menus, which aren't contained by the first four columns. The fifth column doesn't place any line break in a menu.

The Competition of big Menus

There are some cases imaginable with menus which compete against other menus for a line break. autositemap tries to find the best solution in this cases.

Maximum of line breaks

Explanation

autositemap gives priority to the maximum of line breaks.

If first two menus have enough items for a line break, they will fill the first four columns. All other menus will fill the fifth column.

TypoScript Constant Editor

Illustration 12: TypoScript Constant Editor - maxLineBreaks

1. Web Modul: Template
2. Page tree: Root page or page of your choice
3. Edit area: [Constant Editor]
4.   Categorie: [AUTOSITEMAP – RULES]
5.   Priority: [maxLineBreaks]

Possible cases

Illustration 13: Any menu hasn't enough items for a line break.

Illustration 14: The 1st menu has enough items for a line break.

Illustration 15: The 2nd menu has enough items for a line break.

Illustration 16: The 3rd menu has enough items for a line break.

Illustration 17: The 1st and 2nd menu have enough items for a line break.

If you haven't enough menus, there won't be any marginal column.

One line break only

Explanation

Only one menu will get a line break at maximum.

If there are only three menus, one of the menus must have enough items for a line break.

If there are more than three menus, one of the first two menus must have enough items for a line break.

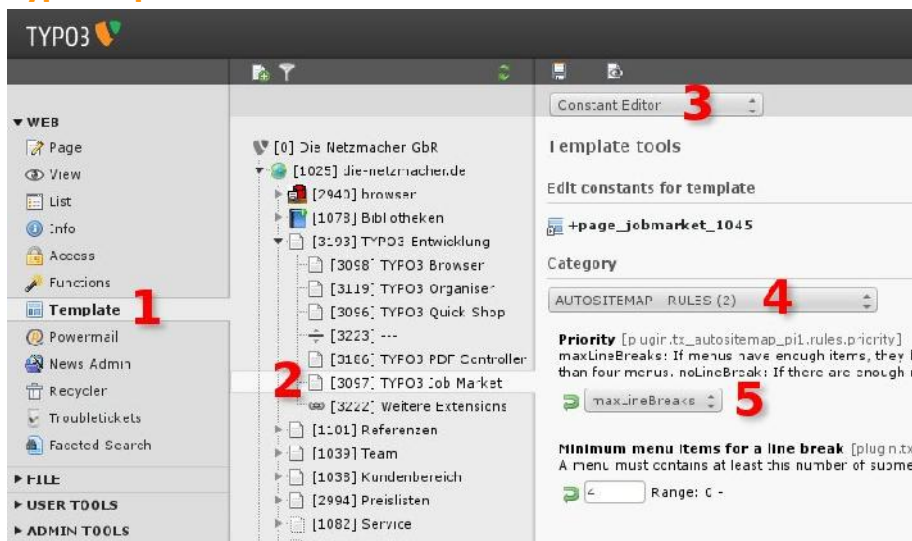## TypoScript Constant Editor
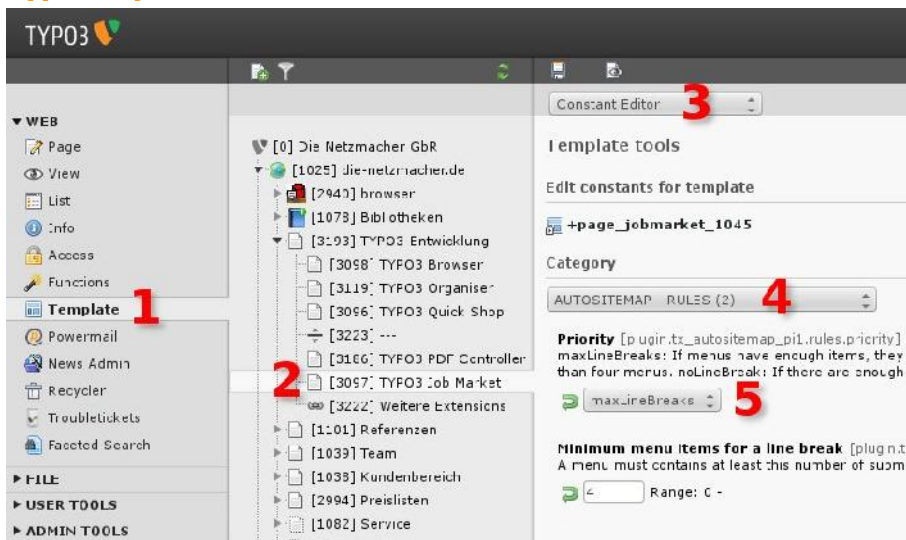
Illustration 18: TypoScript Constant Editor - maxLineBreaks

6.   Web Modul: Template

7.   Page tree: Root page or page of your choice

8.   Edit area: [Constant Editor]

9.   Categorie: [AUTOSITEMAP – RULES]
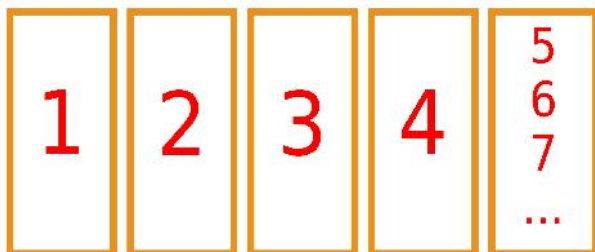
10.   Priority: [oneLineBreak]

Possible cases
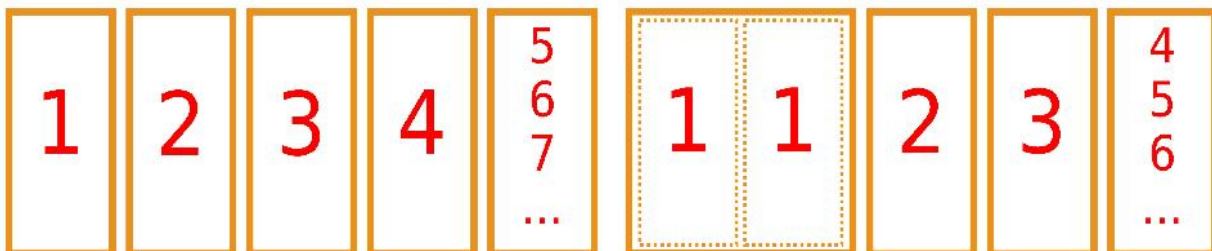
Illustration 19: Any menu hasn't enough items for a line break.
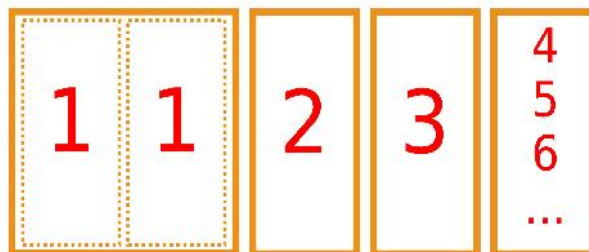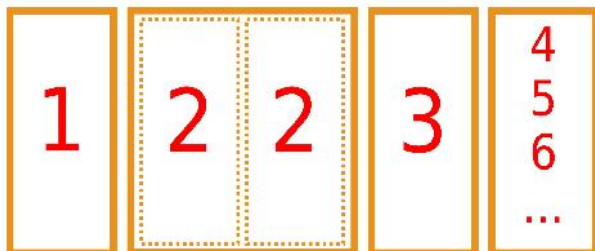

Illustration 20: The 1st menu has enough items for a line break.
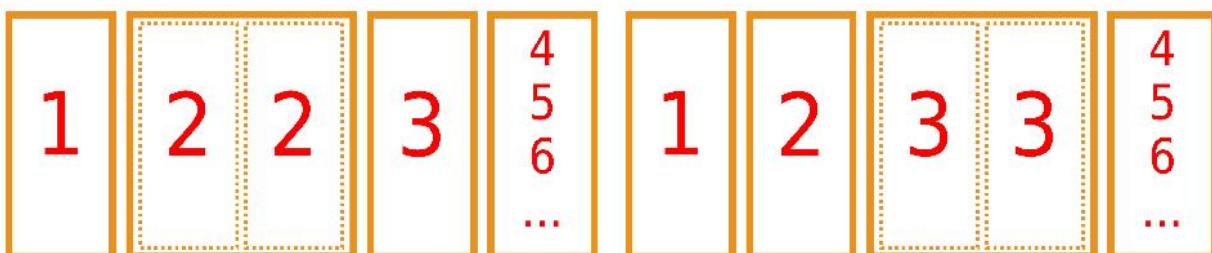
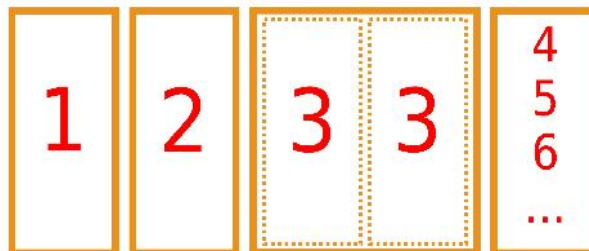Illustration 21: The 2nd menu has enough items for a line break.

Illustration 22: The 3rd menu has enough items for a line break.
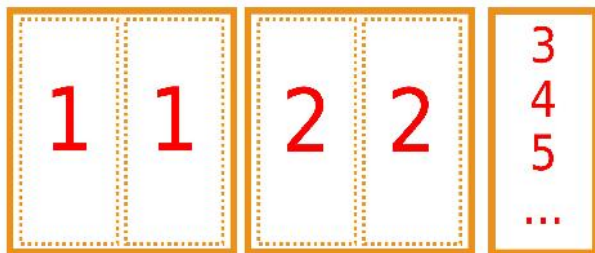
If you haven't enough menus, there won't be any marginal column.

No line break for any menu

Explanation

Any menu won't get any line break.

TypoScript Constant Editor
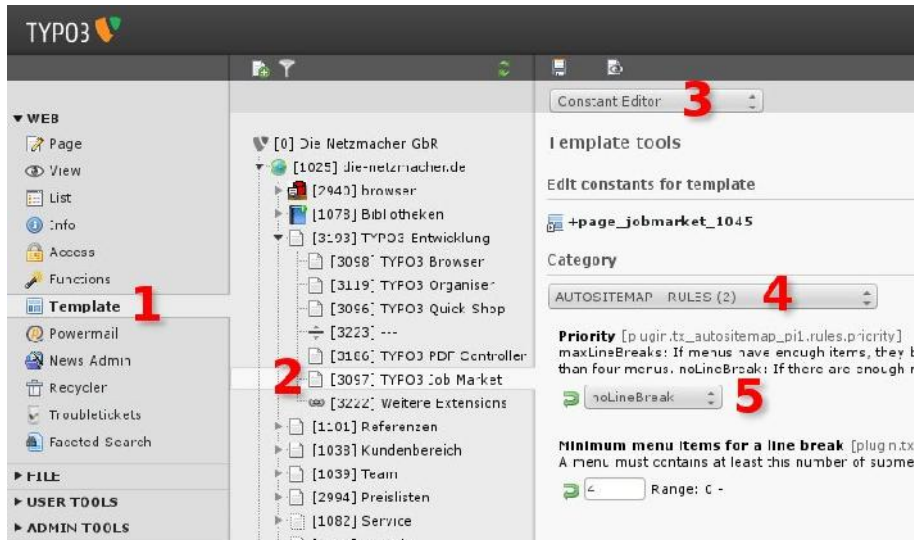
Illustration 23: TypoScript Constant Editor - maxLineBreaks

1. Web Modul: Template
2. Page tree: Root page or page of your choice
3. Edit area: [Constant Editor]
4.   Categorie: [AUTOSITEMAP – RULES]
5.   Priority: [noLineBreak]

Possible cases

Illustration 24: Any menu hasn't a line break.   Illustration 25: Same case with four menus

If you have less than four menus, you will receive less columns.

TypoScript Constant-Editor

AUTOSITEMAP – MENU CONFIGURATION

Root page id (this is a must!)

Uid of the root page. Menu / sitemap will calculated from this startingpoint. You will get an error, if root page id is empty!

Doktypes

You can exclude doktypes from the sitemap.

The doktypes 3, 6, 7, 199, 254, 255 (see below) are excluded by default. That means: the doktypes page and short cut will displayed by default.
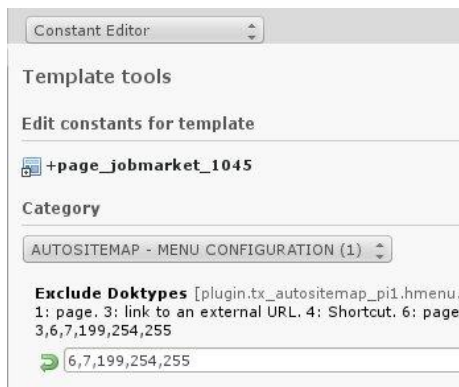
Illustration 26: Configuring doktypes

Defined doktypes

1:	page
3:	link to an external URL
4:	Shortcut
6:	page for backend user
7:	mount point
199:	menu seperator
254:	folder
255:	recycler

Exclude uids

Comma separated list of ids of pages, which should ignored. All subpages of an excluded page will ignored too.

AUTOSITEMAP – RULES

Priority
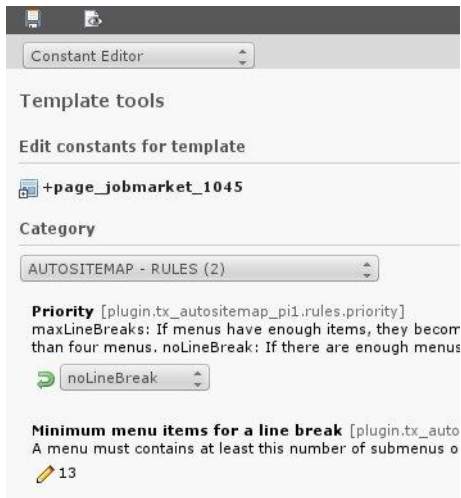
Illustration 27: Rule for a line break

- maxLineBreaks: If menus have enough items, they become double columns.

- oneLineBreak: The greeatest menu will get a double column, if there are less than four menus.

- noLineBreak: If there are enough menus, each menu becomes one column only.

See details at "Priority" on page 7 above.

Minimum menu items for a line break

A menu must contains at least this number of submenus or and items, if it should get a line break.

AUTOSITEMAP – DEBUGGING

Enable Debugging

HTML comments will displayed.

See details at "HTML Code" on page 15 below.

TypoScript Object-Browser

CSS

autositemap includes a CSS file while runtime.

```
page.includeCSS {
  tx_autositemap_pi1_yaml = EXT:autositemap/res/pi1/yaml/autositemap_yaml.css
}
```

If you don't need it, please configure:

```
page.includeCSS {
  tx_autositemap_pi1_yaml >
}
```

If you need an own CSS file please configure something like this:

```
page.includeCSS {
  tx_autositemap_pi1_yaml >
  tx_autositemap_pi1_yaml = fileadmin/my_files/my_own.css
}
```

Master Template

The TypoScript properties are configured by a master template in general.

The parts of the master template are allocated to the autositemap plugin like in the snippet below:

```
plugin.tx_autositemap_pi1 {
  css            < plugin.tx_autositemap_pi1.master_templates.yaml.css
  html           < plugin.tx_autositemap_pi1.master_templates.yaml.html
  menu_main      < plugin.tx_autositemap_pi1.master_templates.yaml.menus.menu_main
  menu_main_last < plugin.tx_autositemap_pi1.master_templates.yaml.menus.menu_main_last
  menu_margin    < plugin.tx_autositemap_pi1.master_templates.yaml.menus.menu_margin
}
```

If you like to configure your own layout, please configure your own master template. A good example is the TypoScript of "+Autositemap YAML (autositemap)". See the files:

```
autositemap/static/pi1/yaml
     •   constants.txt
     •   setup.txt
```

More Configuration in Depth

There are a lot of properties for customising.

Sorry, that these properties aren't a subject of this manual.

You have the option to inspect the TypoScript only.

Debugging

Follow the Workflow

You can follow the workflow with the DRS – the Development Reporting System.

You need the extension devlog.

DRS – Development Reporting System

The DRS – Development Reporting System – prompts the whole workflow into the devlog.

If you have any unexpected result, please enable the DRS by the extension manager.

Enabling

Illustration 28: Enable the DRS – Development Reporting System

1. Modul Admin Tools: Extension Manager
2. Edit area: Select autositemap
3.   [Configuration]
4.   DRS: [Enabled]
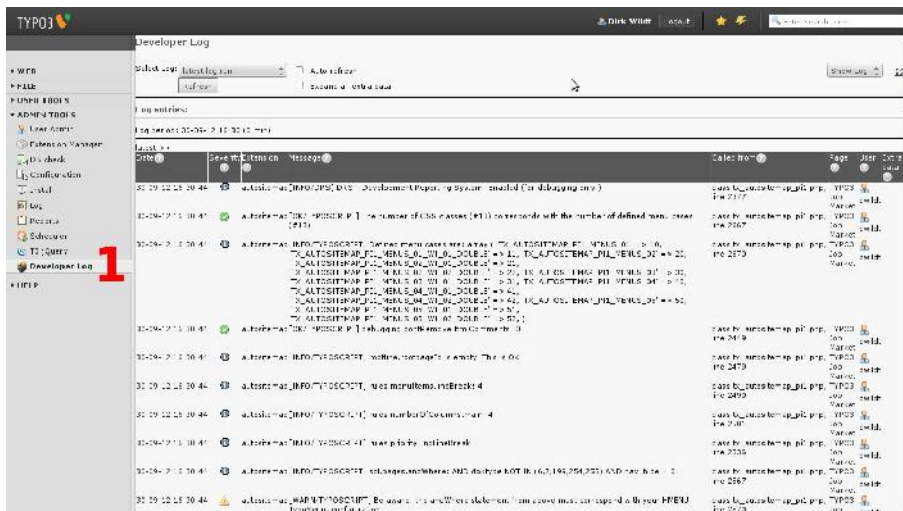5.   [Update]

Reload the page with the autositemap in the frontend.

Report

Illustration 29: Report of the DRS – Development Reporting System

Follow the workflow: read the report!

HTML Code

The autositemap uses some markers for replacement of HTML wraps.

It simplifies to fix bugs, if the markers won't replaced in the HTML code, if there is any unexpected result,

Debugging
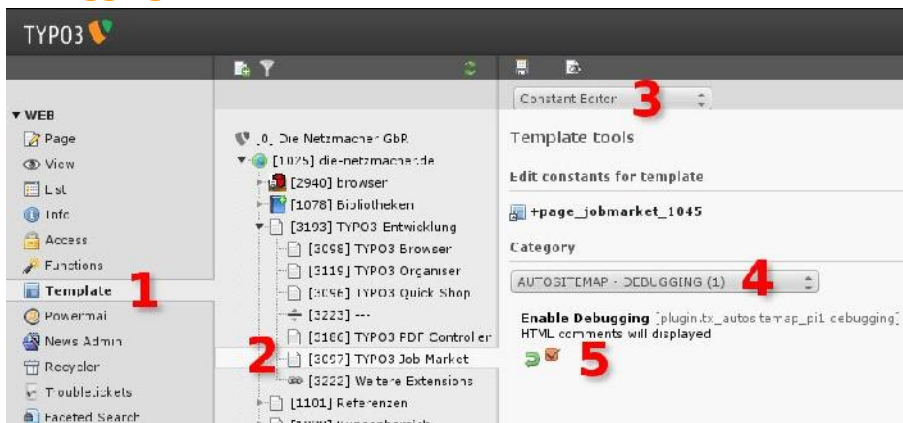

Illustration 30: Display HTML comments

Enable HTML comments (markers won't removed):

1. Modul Web: Template
2. Page tree: Select a page with the autositemap plugin
3. Edit area: [Constant Editor]
4.   Category: [AUTOSITEMAP - DEBUGGING]
5.   [X] Enable Debugging

Reload the page in the frontend.

Inspect the HTML source code.

```
<footer id="footer" class="no-print"><!--###footer### begin -->
    <div class="shadow"><!--###shadow### begin -->
    <!--###shadow### end --></div>
    <div id="footer-logo"><div id="       " class="csc-default"><div class="tx-autositem
        <div class="ym-grid ym-equalize">
  <!-- ###OUTER-UID-    ### begin -->                    <div class="ym-g20 ym-gl">
            <div class="ym-gbox"><!-- ###INNER-UID-3193### begin --><h1 class="cur"><a
        </div><!-- ###OUTER-UID-3193### end --><!-- ###OUTER-UID-1101### begin -->
            <div class="ym-gbox"><!-- ###INNER-UID-1101### begin --><h1 class="ifsub">
        </div><!-- ###OUTER-UID-1101### end --><!-- ###OUTER-UID-1038### begin -->
            <div class="ym-gbox"><!-- ###INNER-UID-1038### begin --><h1 class="ifsub">
        </div><!-- ###OUTER-UID-1038### end --><!-- ###OUTER-UID-1039### begin -->
            <div class="ym-gbox"><!-- ###INNER-UID-1039### begin --><h1 class="ifsub">
        </div><!-- ###OUTER-UID-1039### end --><div class="ym-g20 ym-gr">
            <div class="ym-gbox"><h1 class="ifsub"><a href="preislisten/" title="Preisli
        </div>
    </div>
</div>
```

Illustration 31: HTML source code with autositemap marker

Reference

Property:	Data type:	Description:	Default:

css	array	Variables for your CSS classes.
		
		Please take a look into TypoScript of "+Autositemap YAML (autositemap)" see "Master Template" on page 13 above.
		
		Example:
		
		``
		plugin.tx_autositemap_pi1 {
		  css {
		    // One menu in a single column
		    10 = tx_autositemap_pi1_menus_01
		    10 {
		      // Class for a single column
		      mainSingle = TEXT
		      mainSingle {
		        value = 100
		      }
		    }
		    // One menu in a double column
		    11 = tx_autositemap_pi1_menus_01_wi_01_double
		    11 {
		      // Class for a double column
		      mainDouble = TEXT
		      mainDouble {
		        value = 100
		      }
		    }
		    ...
		  }
		}
		``	

debugging	array	Support for debugging: HTML comments will removed by default (0).
		Please configure it by the TypoScript Constant Editor!
		
		Example:
		
		``
		plugin.tx_autositemap_pi1 {
		  debugging {
		    dontRemoveHtmlComments = TEXT
		    dontRemoveHtmlComments {
		      value = 0
		    }
		  }
		}
		``
			0

html	array	HTML snippets for wraps. The snippets must contain the variable %content%. %content% will replaced with the current main menu or submenu.
		
		Please take a look into TypoScript of "+Autositemap YAML (autositemap)" see "Master Template" on page 13 above.
		
		Example:
		
		``
		plugin.tx_autositemap_pi1 {
		  html {
		    sitemap = TEXT
		    sitemap {
		      ...
		    }
		    menuMainOuterWrap = TEXT
		    menuMainOuterWrap {
		      ...
		    }
		    menuMainInnerWrap = TEXT
		    menuMainInnerWrap {
		      ...
		    }
		    menuSubOuterWrap = TEXT
		    menuSubOuterWrap {
		      ...
		    }
		  }
		}
		``	

Property:	Data type:	Description:	Default:

master_templates	array	Master templates: it contains the whole configuration for the HMENU with TMENUs.
		
		Please take a look into TypoScript of "+Autositemap YAML (autositemap)" see "Master Template" on page 13 above.
		
		Example:
		
		``
		plugin.tx_autositemap_pi1 {
		  master_templates =
		  master_templates {
		    ...
		  }
		}
		``	

menu_main
menu_margin
	array	Configuration of the menu in the main section (first four columns) and of the menu in the margin section (fifth column).
		
		Please take a look into TypoScript of "+Autositemap YAML (autositemap)" see "Master Template" on page 13 above.
		
		Example:
		
		``
		plugin.tx_autositemap_pi1 {
		  menu_main =
		  menu_main_last =
		  menu_margin =
		}
		``	

rules	array	Rules for the number of columns and the line break position.
		Please configure it by the TypoScript Constant Editor!
		
		Example:
		
		``
		plugin.tx_autositemap_pi1 {
		   // Rules for number of columns and line break
		  rules =
		  rules {
		     // Minimum of submenus and items for a line break
		    menuItems =
		    menuItems {
		      ...
		    }
		     // Number of columns ...: DON'T TOUCH IT!
		    numberOfColumns =
		    numberOfColumns {
		      ...
		    }
		     // Priority
		    priority = TEXT
		    priority {
		      ...
		    }
		}}
		``	

Property:	Data type:	Description:	Default:

sql	array	SQL statements. Contains an andWhere for the pages table only. Configuration has to correspond with the configuration of the HMENU.
		
		You don't need to configure it.
		
		Example:
		
		``
		plugin.tx_autositemap_pi1 {
		  sql {
		      // Table
		    pages =
		    pages {
		      andWhere = TEXT
		      andWhere {
		        value = AND doktype NOT IN \
		({$plugin.tx_autositemap_pi1.hmenu.excludeDoktypes})\
		                      AND nav_hide = 0
		        noTrimWrap =
			||
			      }
			    }
			  }
			}
			``	

Whats new in ...

... version 1.0?

Exclude page Ids

[1.0.0]

You can exclude page ids from the the menu / sitemap.

Configurable rootpage ID

[1.0.0]

You can configure the rootpage / staringpoint of the menu / sitemap.

Update

Update from < 0.9.x

You have to allocate the root page id supported by the TypoScript Constant-Editor. See "Root page id (this is a must!)" on page 11 above.

Helpful suggestions

If you have helpful suggestions, feel free to send a question, a bug or a code snippet by e-mail. See http://wildt.at.die-netzmacher.de.

E-mails are welcome in English and German.

Known Bugs

The option "One Line Break" doesn't seem to work proper.

To-Do List

Check the option "One Line Break".

Change Log

1.0.0	**New Features**	25

1.0.0 New Features
* #43696: Exclude page ID
* #43695: Configurable rootpage ID

0.9.3 Bugfix
* static/pi1/yaml/setup.txt: ACRIFSUB > ACTIFSUB

0.9.2 Bugfix
* array_keys() expects parameter 1 to be array in pi1/class.tx_autositemap_pi1.php line 2729

0.9.1 Publishing
Update
* Manual

0.0.4 Improvements
* margin column get the class ym-gr (former: ym_gl)
* Configuring the TypoScript Constant Editor
Bugfix
* pi1/locallang.xml

0.0.3 New Feature
* multiple master_templates
Effected file: static/pi1/setup.txt
* priorities: plugin.tx_autositemap_pi1.rules.priority
    * maxLineBreaks
    * oneLineBreak
    * noLineBreak

0.0.2 Manual

0.0.1 Initial release
Dirk Wildt http://wildt.at.die-netzmacher.de

Illustration Index