

# Coa, a GetSimple framework

Coa is a powerful framework to create lean and functional templates.  
It provides the Coa language and some useful out of the box features for fast and flexible development.

Coa is available for [GetSimple](#), the simplest Content Management System ever.

This documentation is related to Coa 1.6.

## Get started

Watch the [brief video overview](#)

[Download Coa](#) from GetSimple

Visit the [Coa project page](#)

[Drop me a mail](#) if you have any questions

## Do you like Coa and wish to support it?

[www.plue.me/coa/support-coa](http://www.plue.me/coa/support-coa)

Thanks a lot.

## Table of contents

1. Coa Script
2. Image Tags
3. Variables
4. Page Translation
5. License

# 1. Coa Script

A website developed with Coa needs at least two templates. These are 'constant.coa.php' and 'page.coa.php'. The constant template contains the website header and a bit of configuration, the page template contains the website body.

Create objects like 'menu', 'css', 'content' or 'image' to build your template. Assign attributes to them to specify their behaviour. You find all possible objects and attributes in the reference below.

You can use variables like '\$excerpt', '\$description' or '\$title' to print the data you need. Or generate basic galleries in seconds – with Image Tags.

```
1 menu
2   wrap = <nav><ul> | </ul></nav>
3   act = <li class="act"> | </li>
4
5 content
6   wrap = <article><h2>$title</h2> | </article>
7
8 content
9   get = sidebar
10  if = sub
```

## Facts

- Objects have to be separated with an empty line.
- Use '#' at the beginning of a line to comment attributes or even whole objects.
- Paths usually start in the theme folder, defined in 'template.php'.
- Wraps are always separated with a pipe, e.g. <div> | </div>.

## 1.1 Examples

Create a dynamic menu in a breath:

menu

wrap = <ul> | </ul>

act = <li class="act"> | </li>

Make a breadcrumb navigation out of it:

menu

mode = breadcrumb

wrap = <ul> | </ul>

cur = <li class="cur"> | </li>

Get content which is only shown on subpages:

content

get = sidebar

if = sub

You can place other files in your template, too:

place

file = footer.php

wrap = <footer> | </footer>

Use the constant template to include CSS and JavaScript files:

css

all = css/style.css

print = css/print.css

script

js = js/jquery.js

js1 = js/functions.js

## 1.2 Conditions

All objects can have the attributes 'only' and 'exclude'. Both can have comma separated page ids where to show or hide the object.

meta

description = This is the website description ...

only = index, about-us

#exclude = index

All objects can use 'if' with the values 'sub' or '!sub' to show or hide it on subpages.

text

value = <h1> \$title </h1>

if = !sub

and the values 'translation' or '!translation' to show or hide it for a certain language.

text

value = <h1 class="french"> \$title </h1>

if = translation

## 1.3 Constant Objects

### config

**loginReq** = if set to 1, login is required to see the website

**minify** = if set to 1, source code is compressed

**cache** = if set to 1, cache is enabled (disabled if logged in)

**htmlTag** = enter custom string, e.g. <html id="foo">

**bodyTag** = use 'basic' or 'extended' for IE and no js classes or enter custom string

**extFiles** = set to 'none' to exclude external files (e.g. lightbox CSS)

**protMail** = use 1 to protect email strings with %m@d.com%, 2 to protect simple mailto links as well

**wrapPage** = wraps the whole page template, e.g. <div id="wrap"> | </div>

### main

**title** = defines the website title tag

**charset** = website charset, default is utf-8

**language** = set language code for <html> tag, default is 'en' (comma separated for translation)

### meta

**viewport** = defines the content of the viewport meta tag

**robots** = defines the content of the robots meta tag

**description** = website meta description

**keywords** = website meta keywords

**headerGS** = set this to 'none' to exclude the default GetSimple header

### CSS

**wrap** = wraps each <link> tag, e.g. for conditional comments

**all** = path to css file without media tag, also numeric (all1 - all9)

**screen** = path to css file with media="screen", also numeric (screen1 - screen9)

**print** = path to css file with media="print", also numeric (print1 - print9)

**handheld** = path to css file with media="handheld", also numeric (handheld1 - handheld9)

### script

**wrap** = wraps each script tag, e.g. for conditional comments

**js** = path to JavaScript file, also numeric (js1 - js9)

### text

**value** = string to echo into the website header

**wrap** = wraps the whole object

## 1.4 Page Objects

### text

**value** = string to echo

**wrap** = wraps the whole object, e.g. <p> | </p>

**link** = define a link, optional target set after comma

### content

**get** = default is 'one' and gets the page content, also components work e.g. 'sidebar'.

use 'page: page\_slug' to get the content of the given page

**wrap** = wraps the whole object, e.g. <article> | </article>

### image

**file** = path to the image

**alt** = alternative title

**title** = image title

**wrap** = wraps the whole object

**size** = define image width and height, separate with comma (use 'auto' for no attribute)

**link** = define a link, optional target set after comma

### menu

**mode** = use 'breadcrumb' or 'default' (native GetSimple menu)

**list** = comma separated page ids, if set only these pages are shown

**language** = comma separated menu texts for language menu (active if set)

**title** = use 1 to show the title attribute of the <a> tags

**wrap** = wraps the whole menu, default is <ul> | </ul>

**item** = wraps each menu item, default is <li> | </li>

**act** = wraps each active menu item, means current page and its parents

**cur** = wraps the current menu item, e.g. <li id="cur"> | </li>

### place

**file** = path to file that shall be included, allowed are php, html, txt and coa files

**wrap** = wraps the whole object

### script

**js** = path to JavaScript file to fire in the footer, also numeric (js1 - js9)

**wrap** = wraps the script tag, e.g. for conditional comments

## 2. Image Tags

These tags basically fetch images from a given directory. Display them as thumbnails, an image list or in a minimalist lightbox.

- Image captions, titles and alternatives from file title.
- Paths always start at 'data/uploads/'.
- Image Tags are limited to the body.

### Lightbox

- Use [lightbox: path\_to\_folder] to insert the Coa Lightbox.
- Depends on jQuery and a lightbox script. The latter will be included automatically unless external files are deactivated with the config object ('extFiles = none').

### Images

- Use [images: path\_to\_folder] to insert an image list.

### Thumbnails

- Use [thumbs: path\_to\_folder] to insert a thumbnail list

### 3. Variables

There are several variables available you can use universally.

`$title` -> echo page title

`$cleanTitle` -> echo unformatted page title

`$slug` -> echo page slug

`$keywords` -> echo page meta keywords

`$description` -> echo page meta description

`$date / $date{Y}` -> echo page date (last change) with optional format in brackets, default is 'j. M Y'

`$excerpt / $excerpt{50}` -> echo content excerpt with optional length in brackets, default is 100

`$parent` -> echo page parent

`$siteName` -> echo website name

`$theme` -> echo theme name



## 4. Page Translation

Coa adds the ability to translate pages really smoothly. With a click on 'Translate Page' in the sidebar of Pages you can easily translate the page title, menu text and content of every page.

All Coa Objects can be visible or invisible for a certain language. See 'Conditions' for details.

Example language menu (with custom menu text)

menu

language = English, German

act = <li class="act"> | </li>

Settings

With the constant TRANSLATIONURL in 'template.php' you can set the translated language name for the language parameter or pretty url path segment.

With the main object you can set the language codes for the <html> tag:

main

language = en, de

Pretty Urls

To rewrite translated urls with a virtual path segment, you need to set a custom permalink structure like '%slug%.html' or '%parent%/%slug%.html' and your .htaccess file needs to be updated. Replace the default RewriteRule with:

```
RewriteRule ^([A-Za-z0-9_-]+).html index.php?id=$1 [QSA,L]
RewriteRule ^([A-Za-z0-9_-]+)/([A-Za-z0-9_-]+).html index.php?id=$2&L=$1 [QSA,L]
RewriteRule ^([A-Za-z0-9_-]+)/([A-Za-z0-9_-]+)/([A-Za-z0-9_-]+).html index.php?id=$3&L=$1 [QSA,L]
```

If your website is in a subdirectory you need to set the constant SUBDIRPATH in 'template.php' and update the RewriteBase in your .htaccess file as well.

## 5. License

Modified MIT License

Copyright © 2012 Benjamin Scherer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, and/or sublicense, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Feel free to contact me if you have any further questions or feedback.

[www.plue.me](http://www.plue.me) | [mail@plue.me](mailto:mail@plue.me)