

## EXT: LESS Compiler

Extension Key: onm\_less

Language: en

Keywords: less, lessphp, leafo.ne, dynamic css, css processor

Copyright 2000-2010, Thomas Prangenberg, <tpb@onm.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](http://www.typo3.org)

## Table of Contents

EXT: LESS Compiler.....	1	<b>Configuration.....</b>	<b>4</b>
<b>Introduction.....</b>	<b>3</b>	Reference.....	4
What does it do?.....	3	<b>Known problems.....</b>	<b>6</b>
		<b>To-Do list.....</b>	<b>7</b>

# Introduction

## What does it do?

This extension adds the possibility to add LESS ( <http://www.lesscss.org> ) files via `PAGE.includeCSS` in TypoScript. The files are compiled with `lessphp` written by Leaf Corcoran ( <http://leafo.net/lessphp> ). It is influenced by existing solutions like `t3_less`, but with a slightly different approach.

## Features

- Files will be cached and only be re-compiled if modified (while keeping track of modifications in "@import"ed files)
- Output of `lessphp` can be configured in TypoScript
- You can pass functions and variables into LESS via TypoScript

## Configuration

- Install the extension from the TER
- add the static template in your template record - if you don't, nothing will happen at all
- Add your LESS Stylesheets on the PAGE object in your template. The extension will detect LESS files by their file extension (.less) and compile them automatically. The compiled CSS files will be rewritten every time any of the included LESS files (the referenced file as well as files included via @import) were modified (or the compiled CSS files have been deleted).
- Change the output paths and lessphp configuration, if necessary.
- Clear the cache
- You can pass functions and variables to LESS in your TypoScript setup

## Reference

### Constants (plugin.tx\_onmless.[property])

Property:	Data type:	Description:	Default:
path.output	string	Output path. The compiled css files will be written and referenced here.  Please note: File references in your CSS will not be modified by the compiler. You should change this to a directory next to the location of your .less files to preserve the references to your resources (url("../images/image.gif")). For example, if your path to the LESS files looks like the following:  fileadmin/template/less  You may want to set this value to something like this:  fileadmin/template/css	typo3temp/onm_less/cssout
path.cache	string	Cache path. The lessphp cache data will be stored here. Modification shouldn't be necessary.	typo3temp/onm_less/lesscache
lessphp.formatter	string	Can be set to "lessjs" (default), "compressed" or "classic"	lessjs
lessphp.force	boolean	If this is set, the file will always be recompiled, even if not modified	0
lessphp.preserveComments	boolean	Preserve comments in compiled file	0
addHash	boolean	By default, the path to less file will be added as a md5-hash to the output filename to prevent name conflicts. You may disable this here.  Please note that if you use the .variables property in your setup, setting this to FALSE will have no effect, because the variables and their values will always be appended as a hash.	1
enable	boolean	Enable/disable compiling. If this is set to FALSE, the extension won't do anything.	1

### Setup (plugin.tx\_onmless.[property])

(all of the properties in constants, in addition to the following)

Property:	Data type:	Description:	Default:
registerFunction	array	Set of name-callback pairs to be registered in LESS	
variables	array	Variables to pass into LESS.  This will only work if the variable is NOT redeclared in your LESS file. Values set in LESS itself will always have higher priority.	

## Examples

Just include your LESS files the way you would add a usual stylesheet

```
page = PAGE

page.includeCSS {

    #this will be compiled:

    file1 = fileadmin/templates/default/less/styles.less

    #this will not be touched:

    file2 = fileadmin/templates/default/css/justcss.css

}
```

this will result in following output:

```
<link rel="stylesheet" type="text/css" href="typo3temp/onm_less/cssout/styles-
e969c8f8e0ee20df1fb5e27d844495e4.css" media="all">

<link rel="stylesheet" type="text/css" href="fileadmin/templates/default/css/justcss.css" media="all">
```

## Setting variables

Set variables in LESS using TypoScript. The following configuration will set the value of the @awesomeColor variable to "#F00" in your LESS file.

```
plugin.tx_onmless {

    variables {

        awesomeColor = #F00

    }

}
```

## Register Functions

This example is taken from the lessphp documentation and adds a function called `double` that doubles any numeric argument:

### TypoScript

```
plugin.tx_onmless {

    registerFunction {

        double = user_AWESOMEUSERFUNC->lessphp_double

    }

}
```

### PHP

```
class user_AWESOMEUSERFUNC {

    public function lessphp_double($arg) {

        list($type, $value) = $arg;

        return array($type, $value*2);

    }

}
```

### LESS

```
div { width: double(400px); }
```

## Known problems

- None so far, please let me know if you're having any trouble using this extension.

## To-Do list

- Add functionality for combining and compressing JavaScript files as well