

## EXT: beautyOfCode Syntax Highlighter

Extension Key: beautyofcode

Language: en

Copyright 2010, Felix Nagel, <info@felixnagel.com>

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: beautyOfCode Syntax Highlighter.....	1	<b>Available hooks.....</b>	<b>11</b>
<b>Introduction.....</b>	<b>3</b>	addMarker_getHTML.....	11
What does it do?.....	3	addJS_setHeaderData.....	11
Features.....	3	<b>Upgrade-Guide.....</b>	<b>12</b>
Live Demo.....	4	0.5 to 0.6.....	12
Syntax Highlighter 2.0 and v3.....	4	0.6 to 0.7.....	12
jQuery Plugin beautyOfCode.....	4	<b>FAQ.....</b>	<b>13</b>
Screenshots.....	4	No highlighting in FE.....	13
Frontend Plugin.....	4	I get a Java-Script error in FE.....	13
Constant Editor Settings.....	5	I'm using MooTools, ExtJS or similar.....	13
Backend FCE.....	5	Input is encoded and saved without "<", ">", etc.....	13
<b>Users manual.....</b>	<b>6</b>	How to change language strings.....	13
Tab: beautyOfCode.....	6	I don't like my users to overwrite my default TS configuration.....	13
Tab: Options.....	6	How to hide gutter, toolbar or collapse options for users.....	13
<b>Administration.....</b>	<b>7</b>	Where to post improvements or bugs.....	14
Installation.....	7	"Missing Ext. Manager configuration" error in FE.....	14
Settings: Extension Manager.....	7	"t3lib_div::array_merge_recursive_overrule()" error in BE.....	14
Which version to choose.....	7	Enable t3editor in BE.....	14
<b>Configuration.....</b>	<b>8</b>	<b>Known problems.....</b>	<b>15</b>
Typoscript.....	8	<b>To-Do list.....</b>	<b>16</b>
<b>Tutorial.....</b>	<b>10</b>	<b>ChangeLog.....</b>	<b>17</b>
Add missing brushes.....	10		
How to add a template marker with hooks.....	10		

# Introduction

## What does it do?

This plugin provides state-of-the-art syntax highlighting by using Java-Script by using SyntaxHighlighting 2.0 by Alex Gorbachev.

This is done by using the jQuery plugin beautyOfCode (jQuery version, SyntaxHighlighter v2) or without using jQuery (standalone version )by using new autoloader feature introduced with SyntaxHighlighter v3.

Use the constant editor or TS to define needed programming languages. These will be available in the FE plugin which needs to be added as a generic FE plugin ("insert plugin"). Each of this languages is a lazy-loaded (after the page has finished loading) Java-Script file. You can define different css styles (so called themes) and HTML templates to match your page design.

The following programming languages are available: Actionscript 3, Bash, Shell, ColdFusion", C, C++, C#, CSS, Delphi, Pas, Pascal, Diff, Patch, Erlang, Groovy, Java, Java FX, Java-Script, Perl, PHP, Power-Shell, Python, Ruby on Rails, Scala, SQL, Typoscript, MySQL, Virtual Basic, .Net, XML, XSLT, XHTML and HTML. There are even more available, please check out <http://alexgorbatchev.com/wiki/SyntaxHighlighter:Brushes>

The newly added Typoscript brush is based upon a [user language for Notepad++](#).

**Note:** This extension currently needs jQuery Java-Script framework (see Installation). I'm working in a non-jQuery option, see ToDo. Since version 0.7.0 its possible to use this extension without jQuery.

## Features

- Highlight specific lines
- Show Gutter
- Customizable tab size
- Configurable label
- Collapse code
- Wrap lines
- Preview and t3editor support in BE
- Use HTML templating
- Different visual styles
- Configurable JS language strings in FE \*
- Show Toolbar with \*
  - Print in FE view
  - Copy to directly to clipboard (via swf) in FE view
  - Copy manually out of an pop-up in FE view

\* if using jQuery version aka Syntax Highlighter v2

### Code is embedded as:

jQuery Version (Syntax Highligther v2)

```
<pre class="code">
  <code class="php">
    CODE GOES HERE
  </code>
</pre>
```

Standalone Version (Syntax Highligther v3)

```
<pre class="php">
  CODE GOES HERE
</pre>
```

## Live Demo

- [FelixNagel.com/Blog/](http://FelixNagel.com/Blog/)

Please help me to add more sites here: [contact me!](#)

## Syntax Highlighter 2.0 and v3

Syntax Highlighter 2.0 by Alex Gorbachev, which does the actually highlighting, is the nicest syntax highlighter i've seen so far and it's also used by Freshbooks, ASP .Net Forums, wordpress.com, Aptana, Mozilla Developer Center, SitePoint and Yahoo Developer Network. It is coded in native Java-Script and is under active development.

Please see <http://alexgorbatchev.com/wiki/SyntaxHighlighter> for more information.

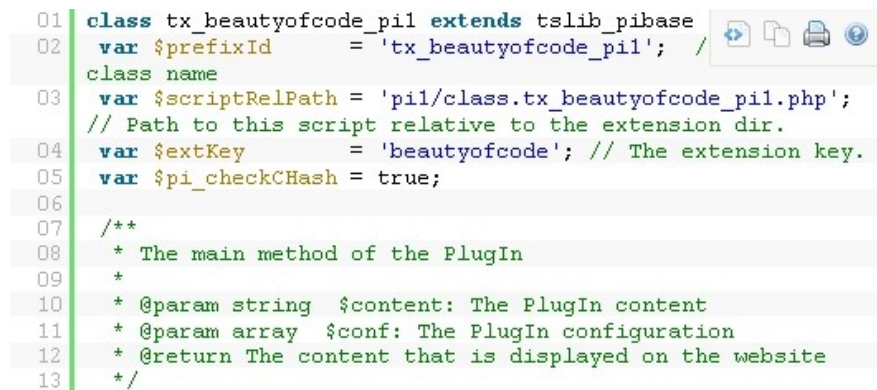
## jQuery Plugin beautyOfCode

Beauty of Code is a jQuery plugin written by Lars Corneliusen. This extension uses beautyOfCode to implement SyntaxHghlighter 2.0 in a xhtml 1.0 compliant way (see above) and to add the desired js files by [lazy loading](#).

See <http://startbigthinksmall.wordpress.com/2009/05/28/beautyofcode-jquery-plugin-for-syntax-highlighter-2-0-by-alex-gorbachev/> for more information.

## Screenshots

### Frontend Plugin



```

01 class tx_beautyofcode_pi1 extends tslib_pibase
02 var $prefixId      = 'tx_beautyofcode_pi1'; //
   class name
03 var $scriptRelPath = 'pi1/class.tx_beautyofcode_pi1.php';
   // Path to this script relative to the extension dir.
04 var $extKey        = 'beautyofcode'; // The extension key.
05 var $pi_checkCHash = true;
06
07 /**
08  * The main method of the PlugIn
09  *
10  * @param string $content: The PlugIn content
11  * @param array $conf: The PlugIn configuration
12  * @return The content that is displayed on the website
13  */

```

This is the default theme when using jQuery version.

## Constant Editor Settings

<h3>Template-Werkzeuge</h3> <p>Konstanten bearbeiten für Template:</p> <p><b>Main TypoScript</b></p> <p>Kategorie: <input type="text" value="BEAUTYOF CODE (12)"/></p> <p><b>Enable features</b></p> <p><b>Loaded programming languages</b> [plugin.tx_beautyofcode_pi1.brushes] Define which programming languages should be available. Less is more: every brush is Erlang, Groovy, Java, JavaFX, JScript, Perl, Php, PowerShell, Python, Ruby, Scala, Typo</p> <p><input type="text" value="Xml,JScript,Php,Css,Typoscript"/></p> <p><b>Possible themes</b> [plugin.tx_beautyofcode_pi1.theme] Midnight, RDark, Default, Django, Eclipse, Emacs, FadeToGrey, FelixNagelv3 (which is d</p> <p><input type="text" value="FelixNagelv3"/></p> <p><b>Show label</b> [plugin.tx_beautyofcode_pi1.showLabel] If set to false the label is hidden.</p> <p><input type="checkbox"/></p> <p><b>Files</b></p> <p><b>jQuery noConflict</b> [plugin.tx_beautyofcode_pi1.noConflict] If set to true all embedded Java-Script code will be wrapped in jQuery's noConflict() fun</p> <p><input type="checkbox"/> [Empty]</p> <p><b>Template file</b> [plugin.tx_beautyofcode_pi1.templateFile] Default: EXT:beautyofcode/res/template.html</p> <p><input type="text" value="EXT:beautyofcode/res/template.html"/></p> <p><b>Add jQuery Library</b> [plugin.tx_beautyofcode_pi1.addjQuery] This option adds jQuery Framework file if the extension t3jquery is NOT installed. See d</p> <p><input type="checkbox"/></p> <p><b>jQuery selector</b> [plugin.tx_beautyofcode_pi1.jQuerySelector] Define a jQuery selector to improve frontend performance (example: #myMainContent)</p> <p><input type="text" value="#left"/></p>	<h3>Others</h3> <p><b>Size of tabulator</b> [plugin.tx_beautyofcode_pi1.defaults.tab-size] Tabulator chars will be changed to spaces. Default is 4 (spaces).</p> <p><input type="text" value="2"/> Range: 0 -</p> <p><b>Toolbar</b> [plugin.tx_beautyofcode_pi1.defaults.toolbar] Show toolbar when mouseover (with following functions: code in pop-up, copy to cliboar</p> <p><input type="checkbox"/></p> <p><b>Gutter</b> [plugin.tx_beautyofcode_pi1.defaults.gutter] Show or hide gutter. Helps user to recognize correct line with numbers.</p> <p><input type="checkbox"/></p> <p><b>Collapse</b> [plugin.tx_beautyofcode_pi1.defaults.collapse] Allows you to force highlighted elements on the page to be collapsed. A link "show source</p> <p><input type="checkbox"/></p> <p><b>Wrap lines</b> [plugin.tx_beautyofcode_pi1.defaults.wrap-lines] Allows you to turn line wrapping feature on and off. Recommended to to be enabled.</p> <p><input type="checkbox"/></p>
---	---

## Backend FCE

<p>Allgemein Plug-In Zugriff</p> <p><b>Plug-In:</b></p> <p><input type="text" value="Quellcode (beautyOfCode)"/></p> <p><b>Erweiterungsoptionen:</b></p> <p><b>DEF:</b></p> <p><input type="text" value="beautyOfCode"/> Optionen</p> <p>Beschriftung</p> <p><input type="text" value="123 test"/></p> <p>Programmiersprache</p> <p><input type="text" value="Typoscript"/></p> <p>Quellcode</p> <pre>[browser = msie]   page.headerData {     20 = TEXT     20.value = &lt;link rel="shortcut icon" href="fileadmin/templates/favicon.ico"   } [else]   page.headerData {     20 = TEXT     20.value = &lt;link rel="icon" href="fileadmin/templates/favicon.gif" type="image/gif" /&gt;   } [global]</pre>	<p>Allgemein Plug-In Zugriff</p> <p><b>Plug-In:</b></p> <p><input type="text" value="Quellcode (beautyOfCode)"/></p> <p><b>Erweiterungsoptionen:</b></p> <p><b>DEF:</b></p> <p><input type="text" value="beautyOfCode"/> Optionen</p> <p>Markierung für bestimmte Zeilen (1,2,3,4-6...)</p> <p><input type="text" value="4,8"/></p> <p>Quellcode Elemente einklappen</p> <p><input type="text" value="Ja"/></p> <p>Zeilennummerierung anzeigen</p> <p><input type="text" value="Nein"/></p> <p>Toolbar anzeigen</p> <p><input type="text" value="Standard"/></p>
--	--

Screenshot Backend Standard Tab

Screenshot Backend Overwrite Configuration Tab

# Users manual

Add the beautyofcode frontend plugin via the FCE wizard or manually via a generic plugin. After adding the plugin there are 2 tabs available:

## Tab: beautyOfCode

These fields should be self explanatory.

- Description (not necessary, if empty no description is displayed)
- Programming language (available languages configured by your admin)
- Code

## Tab: Options

Within this tab its possible:

- to highlight some specific lines. Use a syntax like: *1,2,3,4,5 or 1-5*
- to overwrite the default settings configured by your admin.
  - Gutter (show line numbers)
  - Toolbar (shows the toolbar with copy to clipboard, show in pop-up, print and help)
  - Collapse (displays code element collapsed)

Please note: Configuration overwrite via the Option tab could be disabled by your admin.

# Administration

## Installation

- Install the extension via extension manager.
- Add the static template in your main typoscript (go to: list module, root page, edit your root TS template, tab: includes, add "beautyOfCode Syntax Highlighter").
- You could add the jQuery core file yourself, via this extension by enable *addjQuery* configuration or by using the t3jquery extension. Version 1.3.x and 1.4.x of jQuery will work. Since version 0.7.0 it's possible to use SyntaxHighlighter standalone without using jQuery.

**Please note:** It's strongly recommended to add one line in your localconf.php to force CDATA escaping within the flexforms in TYPO3. Please see know FAQ section.

## Settings: Extension Manager

Enable syntax highlighting (install EXT: t3editor) in BE by using enable\_t3editor option.

**Please note:** Some extension manager settings have been removed in version 0.7.0. Use TypoScript instead.

## Which version to choose

This extension has two versions of SyntaxHighlighter to choose. Both lazy load the necessary CSS and JS files after the DOM ready event has fired. Version 2 uses a jQuery plugin "beautyOfCode" by [Lars Corneliusen](#) and version 3 wich runs standalone using the autoloader feature introduced with version v3. There are some benefits and disadvantages by using each version:

Version v2 ("jquery"):	Version v3 ("standalone"):
<ul style="list-style-type: none"> <li>• needs jQuery</li> <li>• configurable toolbar</li> <li>• runs within a domReady event (runs very smooth when merging scripts automatically)</li> <li>• less code needs to be injected in your HTML</li> <li>• wrap-lines feature (line break)</li> <li>• more feature rich</li> </ul>	<ul style="list-style-type: none"> <li>• runs without using a Java-Script framework</li> <li>• no toolbar available as it has been removed in v3</li> <li>• uses latest version of SyntaxHighlighter</li> <li>• slim and less feature rich (but with almost the same functionality)</li> </ul>

# Configuration

## Typoscript

Feel free to use the constant editor (go to: module template, root page, constant editor, beautyofcode) to edit this settings. Add a Typoscript template to overwrite these settings at any point in your sitetree(s).

Property:	Data type:	Description:	Default:
templateFile	String	Path to the template file.	EXT:beautyofcode/res/template.html
jquery.baseUrl standalone.baseUrl	String / TS object	Enter path to the res directory by using EXT: or FILE: or absolute path (http://your.domain.de/fileadmin/.../res/). Make sure the relative paths res/ and styles/ exists.  Leave empty to use online repository. If you use online repository you shall not edit settings <i>styles</i> and <i>scripts</i> . More info about online hosting see: <a href="http://alexgorbatchev.com/wiki/SyntaxHighlighter:Hosting">http://alexgorbatchev.com/wiki/SyntaxHighlighter:Hosting</a>	EXT:beautyofcode/res/
jquery.scripts standalone.scripts	String	Path to syntax highlighter core file and to the brushes (Java-Script files), relative to the <i>baseUrl</i> . Leave empty or default when using online repository.	scripts/
jquery.styles standalone.styles	String	Path to syntax highlighter css files (css themes), relative to the <i>baseUrl</i> . Leave empty or default when using online repository.	styles/
standalone.includeAsDomReady	String	If using standalone version it's possible to add a JS domReady instaed of injecting the code at the bottom of the body. Useful when using minification scripts. Possible options: false, jquery or standalone	false
jquery.scriptUrl	String / TS object	Path to jquery.beautyOfCode.js file (use FILE: and EXT: or absolute path http://your.domain.de/...).	EXT:beautyofcode/res/jquery.beautyOfCode.js
jquery.addjQuery	Boolean	This option adds jQuery Framework file if the extension t3jquery is NOT installed. All other Java-Script files are added manually (see Extension Manager configuration).	1
jquery.selector	String	Please provide a jQuery selector (like <i>#myMainContent</i> ) to improve frontend performance. Is set beautyOfCode will only search for code snippets within the given id.	
jQueryNoConflict	Boolean	If set to true all embedded jQuery Java-Script code will be wrapped in jQuery's noConflict() function. This helps you to use jQuery together with other JS Frameworks which use \$ as a function or variable name, too. Useful if you using Mootools (or similar) and want to use beautyOfCode (and therefor jQuery) only to highlight your code.  See <a href="http://api.jquery.com/jquery.noConflict/">http://api.jquery.com/jquery.noConflict/</a> for more information.	0
brushes	String	Loaded programming languages: Define which programming languages should be available. Less is more: every brush is lazy loaded a single js file. Add a seperated list out of: <i>AS3, Bash, ColdFusion, Cpp, CSharp, Css, Delphi, Diff, Erlang, Groovy, Java, JavaFX, JScript, Perl, Php, PowerShell, Python, Ruby, Scala, Typoscript, Sql, Vb, Xml</i> .	Xml,JScript,CSharp,Plain
theme	String	Define a theme (which is basically a CSS file). Following themes are avaiable: Midnight, RDark, Default, Django, Eclipse, Emacs, FadeToGrey, FelixNagelv3 (which is dark minimal).  Add your own in EXT:res/styles/	Default
showLabel	Boolean	If set to false the label is hidden.	1
defaults.tab-size	Int	Specify a tabulator size. Tabulator chars will be changed to spaces.	4
defaults.toolbar	Boolean	Show toolbar when mouseover the code in FE (with following functions: code in pop-up, copy to cliboard via swf, print, info about SyntaxHighlighter 2.0).	1
defaults.gutter	Boolean	Show or hide gutter. Helps user to recognize correct line.	1
defaults.collapse	Boolean	Allows you to force highlighted elements on the page to be collapsed. A link "show source" is displayed instead (not customizable yet).	0
defaults.wrap-lines	Boolean	Allows you to turn line wrapping feature on and off. Recomendend to to be enabled.	1



Property:	Data type:	Description:	Default:
config.strings	object	<p>Configure SyntaxHighlighter language strings in frontend. See <a href="http://alexgorbatchev.com/SyntaxHighlighter/manual/configuration/strings.html">http://alexgorbatchev.com/SyntaxHighlighter/manual/configuration/strings.html</a> for possible strings. Please see pi1/static/setup.txt for an extended example how to use.</p> <p>For multilanguage installations please use globalVar conditons:  [globalVar = GP:L = 1]  config.strings.viewSource = custom string  [global]</p>	

[tsref:plugin.tx\_beautyofcode\_pi1]

# Tutorial

## Add missing brushes

Please see <http://alexgorbatchev.com/SyntaxHighlighter/manual/brushes/> for all available brushes. You need to edit the EXT:/pi1/class.tx\_beautyofcode\_addFields.php: add a case within the switch command to use your custom brush. If you're using standalone (aka SyntaxHighlighter v3) you need to extend function chooseCssTag in filelib/class.tx\_beautyofcode\_standalone.php, too.

Please contact me if you would like to see a brush in the next release, I will add it asap.

## How to add a template marker with hooks

You need a custom marker within your template? No problem:

Generate user\_ extension via kickstarter.

Add following code to your extension *ext\_localconf.php*:

```
$GLOBALS['TYP03_CONF_VARS']['EXTCONF']['beautyofcode']['addMarker_getHTML'][] =
'EXT:user_hooktest/class.tx_user_hooktest.php:tx_hooktest->addMarker';
```

Your class could look like this:

```
class tx_hooktest {
    function addMarker(&$content, $obj) {
        $content['###TEST###'] = "test123";
    }
}
```

Now you can add your new marker to the template:

```
###TEST###
```

## Available hooks

I added some hooks to enable you to extend my extension. Please contact me to add a hook for you! Please notice that you are able to use XCLASS as well.

### addMarker\_getHTML

This hook enables you to add or modify the processed template markers. Use following code in your ext\_localconf.php

```
$GLOBALS['TYPO3_CONF_VARS']['EXTCONF']['beautyofcode']['addJS_setHeaderData'][] =  
'EXT:user_myhook/class.tx_user_myhook.php:tx_myhook->myFunction';
```

### addJS\_setHeaderData

This hook enables you to add some JS into the JS code added for each FCE. Use following code in your ext\_localconf.php

```
$GLOBALS['TYPO3_CONF_VARS']['EXTCONF']['beautyofcode']['addJS_setHeaderData'][] =  
'EXT:user_myhook/class.tx_user_myhook.php:tx_myhook->myFunction';
```

# Upgrade-Guide

If there's no upgrade guide, there's no need to change anything. This guide does not cover extending via PHP (aka XCLASS or Hooks) – you need to check changes for yourself.

## 0.5 to 0.6

This version introduces HTML templating. That's why some TS options are not longer available or renamed:

- `plugin.tx_beautyofcode_pi1.label.wrap` is removed as it's now configured via template
- `plugin.tx_beautyofcode_pi1.label.show` is renamed to `showLabel`
- `plugin.tx_beautyofcode_pi1.wrap` is removed as it's now configured via template

## 0.6 to 0.7

Big update: finally its possible to use this extension without jQuery. Massive remanufacturing and code changes. Please consider this release as a beta version as I did test every feature the best I can but not sure if everything will work out on every TYPO3 installation.

What you need to know:

- You'll need to change your TypoScript code
  - paths for *base*, *scripts* and *styles*
  - *addjQuery* and *selector* have been moved to the *jquery*. *Array*
  - some variables have been slightly renamed
- Extension Manager settings has been removed, use TS configuration instead
- SyntaxHighlighter v3 changed some features (learn more: ["Which version to choose"](#))

Please see the changelog for all changes in detail.

## FAQ

### No highlighting in FE

Did you include a jQuery core file? You either need to install and configure the t3jquery extension, add the jQuery Core file manually or activate the addjQuery option.

### I get a Java-Script error in FE

See above.

Do you use other Java-Script frameworks? Please see down below.

### I'm using MooTools, ExtJS or similar

No problem. There are two possible solutions to this problem:

#### Using standalone version:

Just use the standalone version of this plugin. This is possible by using:

```
plugin.tx_beautyofcode_pil.version = standalone
```

#### Using JQuery Version:

Use the noConflict option via TS or Constant Editor. This way you can use your own Java-Script framework (which uses the \$ selector as well) together with beautyOfCode as it will work "standalone" with a incompatible Java-Script framework.

Please note and keep in mind: if jQuery must be loaded (minified about 56KB) which leads to more traffic and less performance for your users.

### Input is encoded and saved without "<", ">", etc.

This is a server side problem within PHP 5 respective in the libxml module (1.6.32 and higher) with an existing work around.

#### Solution:

Add this line in your localconf.php in typo3conf directory.

```
$TYPO3_CONF_VARS['BE']['flexformForceCDATA'] = '1';
```

This fix is non breaking and works only for new added elements. Older flexform values should / could be fixed manually. This is a recommended default setting for every TYPO3 installation (see mailing list why).

#### Bug:

<http://bugs.typo3.org/view.php?id=9359>

#### Mailinglist Post:

[TYPO3-dev] A ticking ticking timebomb with libxml update to 2.7.1?

### How to change language strings

Please see config.strings in Configuration → Typoscript

### I don't like my users to overwrite my default TS configuration

Please see below.

### How to hide gutter, toolbar or collapse options for users

If you don't like your user to overwrite your TS configuration via the *Options* tab within the flexform (plugin FCE) you should take a look at the extension *sp\_betterflex*: "Exclude static flexform fields made by extensions like normal table fields in backend group configuration or via TSConfig."

[http://typo3.org/extensions/repository/view/sp\\_betterflex/current/](http://typo3.org/extensions/repository/view/sp_betterflex/current/)

Install the extension and add these lines to your TSconfig:

```
TCEFORM.tt_content.beautyofcode_cGutter.disabled = 1
TCEFORM.tt_content.beautyofcode_cToolbar.disabled = 1
TCEFORM.tt_content.beautyofcode_cCollapse.disabled = 1
```

## Where to post improvements or bugs

Please feel free add questions, bugs and improvements at <http://forge.typo3.org/projects/extension-beautyofcode/issues>

## “Missing Ext. Manager configuration” error in FE

You forgot to save the extension manager configuration. Please go to *Extension Manager* → *beautyOfCode* → click *Update*

## “t3lib\_div::array\_merge\_recursive\_overrule()” error in BE

This error could look like this:

PHP Catchable Fatal Error: Argument 1 passed to t3lib\_div::array\_merge\_recursive\_overrule() must be an array, null given, called in /html/typo3/typo3\_src-4.4.2/typo3/sysex/rtehtmlarea/class.tx\_rtehtmlareaapi.php on line 80 and defined in /html/typo3/typo3\_src-4.4.2/t3lib/class.t3lib\_div.php line 2059

This issue was already patched but seems to be introduced again in specific TYPO3 version. Some kind of regression I guess. Sadly Im not able to reproduce the issue and it even seems to be an Core / RTE problem, so please take a look at these bugtracker issues which should help you to solve the problem:

- <http://bugs.typo3.org/view.php?id=15864>
- <http://bugs.typo3.org/view.php?id=15893>

## Enable t3editor in BE

You need to enable the *enable\_t3editor* option in Extension Manager and check your user settings.

## Known problems

Please see FAQ and check <http://forge.typo3.org/projects/extension-beautyofcode/issues>

- PHP5 and libxml module cause problems in some Installations. This is a server side problem with an existing work around. See the FAQ of this manual.

## To-Do list

Please see <http://forge.typo3.org/projects/extension-beautyofcode/issues>



## ChangeLog

Please see ChangeLog file in extension directory: `typo3conf/ext/beautyofcode/`