

Core - Feature # 19693

Status:	Resolved	Priority:	Should have
Author:	Oliver Hader	Category:	Communication
Created:	2008-12-11	Assigned To:	Oliver Hader
Updated:	2009-11-27	Due date:	
PHP Version:	5.2		
Complexity:			
Sprint Focus:			
Subject:	Integrate possibility to define additional header data via TypoScript		
Description			
<p>Imagine you've got an extension that shall add a JavaScript and a CSS file to the HEAD part of the HTML document of the current page. This is possible in modifying \$TSFE->additionalHeaderData. However, when your extension does not come with a PHP part and consists only of TypoScript, it's not possible.</p> <p>Thus, this feature request is about to integrate an accordant possibility.</p> <p>Example:</p> <pre>10 = HEADERDATA 10 { data = path:EXT:myextension/res/stylesheet.css wrap = <link rel="stylesheet" type="text/css" href=" " /> prepend = TEXT prepend.char = 9 }</pre> <p>This would do the following:</p> <ul style="list-style-type: none">- add the CSS file "stylesheet.css" to the HEAD part of the current document <p>(issue imported from #M9923)</p>			
Related issues:			
related to Core - Bug # 20661: [Feature] Add class for render HTML header/foo...		Closed	2009-06-24

History

#1 - 2008-12-11 20:59 - Oliver Hader

Documentation changes:

doc_core_tsref, functions, stdWrap:

Property: additionalHeaderData

Data Type: cObject

Description: Adds the cObject to the <HEAD> part of the current document

#2 - 2008-12-26 14:55 - Steffen Kamper

Hi olly,

This is a nice idea, it allows adding header data depending on defined objects.

+1!

#3 - 2009-06-15 00:50 - Johannes von Bargaen

Yeah! That would be cool! I just had a discussion with several people at typo3camp Hamburg about such a possibility.

+1!!!

I am currently working on a extension which integrates jQuery Galleriffic and Galleria into cType Image. That brought the ugly headerData array back to my mind... Additionally it would be cool if headerData was associative and extensions would automatically have a private namespace in the array to add their stuff (the extension key). This way the case where one ext overwrites js /css/meta stuff added by another one would never happen again by accident but otherwise fields of the array could be addressed explicitly if ext B absolutely wants to change something ext A has added.

I think I will open up an own bug for that...

#4 - 2009-06-15 10:21 - Vladimir Podkovanov

+1 for stdWrap.additionalHeaderData

to Johannes von Bargaen,

+ 1 for converting page.headerData to assoc array so extension does not bother which numeric keys are used yet. I was running into it too.

#5 - 2009-06-25 19:31 - Oliver Hader

Documentation changes:

add a new entry to "Content Objects (cObject)" (tsref.cObject):

HEADERDATA

HEADER is very similar to the cObject "TEXT". However the result will not be added to the content but to the HTML HEAD section of the page. This can be used to define JavaScript or Stylesheets on an extension level for example.

Property	Data type	Description
value	string	text, wrap with stdWrap properties (stdWrap properties)

Example:

```
10 = HEADERDATA
10 {
  value = script.js
  wrap = <script src="|" type="text/javascript"></script>
}
```

#6 - 2009-11-27 16:38 - Oliver Hader

This is solved with the PageRenderer in issue #20661

Files

0009923.patch	1 kB	2008-12-11	Administrator Admin
0009923_v2.patch	1.4 kB	2009-06-25	Administrator Admin