

EXT: Mobile devices capabilities detector

Extension Key: tk_mobiledetector

Language: en

Keywords: mobile, device, uaprof, capabilities, typoscript, conditions

Copyright 2011-2013, Tomasz Krawczyk, <tomasz@typo3.pl>

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

EXT: Mobile devices capabilities detector.....	1	Backend module.....	4
Introduction.....	3	TypoScript conditions.....	5
What does it do?.....	3	Examples.....	7
How it works?.....	3	Mobile site debugging.....	11
Why not WURFL?.....	3	Firefox plugin - Modify headers.....	11
Administration.....	4	FE Plugin "Mobile detector – tester"	11
Installation.....	4	Known problems.....	12
Extension configuration.....	4	To-Do list.....	13

Introduction

What does it do?

- This extension tries to detect mobile device capabilities and stores them into database.
- Provides new TypoScript conditions that may help you create better mobile sites.

How it works?

Device capabilities detection is based on [UAProf \(User Agent Profile\)](#). Extension looks for HTTP headers: HTTP_PROFILE, HTTP_X_WAP_PROFILE, WAP-PROFILE, X-WAP-PROFILE or XX-PROFILE. Those headers may contain link to UAProf file. The UAProf file is an XML document that describes mobile device capabilities such as: vendor, model, screen size, supported image types and many other. Extension checks if found UAProf link exists in data table. If yes then gets device capabilities from the table. If the link doesn't exist in data table then UAProf link is red, parsed and mobile device capabilities are stored in database.

For detecting device type from User Agent method from <http://detectmobilebrowsers.com> was taken.

Why not WURFL?

Since August 2011 the great WURFL database is no longer Open Source. From its last free version devices was extracted and all unique device UAProfs was stored to static extension data.

Administration

You may administer the extension in two places. In The Extension Manager and in the Backend module.

Installation

- Install the extension tk_mobiledetector in the Extension manager.
- Import static data.
- Set the extension configuration options.
- Include example static templates or write them on you own. Two templates are provided "Mobile Main" for a whole mobile site and "Mobile tt_news" for news images sizing.

Extension configuration

Option	Description	Default
DetectTablets	If you turn this feature on then this ext will try to detect tablets. Device type "tablet" will be available in TypoScript. In most cases tablets should be considered full sized browsers.	0
DetectedDeviceType	This is a short string used in Typoscript in order to tell difference between browser, grabber or robot. Curently TYPO3 supports "pda" and "wap" for all mobile devices. This extension adds new type mobi .	mobi
MobileSiteUrl	Enter here your mobile site address. This will be used in the future release for exchange with capabilities server.	http://mysite.mobi

Backend module

Extension adds to the backend "Mobile devices" module to the Admin Tools section. Currently you can use two options:

Device statistics – provides information about imported devices from the last open source WURFL database and collected by your site from visiting devices.

Check and add device – lets you check if UAProf link exists in database. If the link is new then it is parsed and added to data table.

TypoScript conditions

Condition:	Data type:	Description:
vendor	string	<p>Device vendor like: Nokia, SonyEricson, HTC, LG, ... You can use wildcard * or PCRE Regular Expressions (see doc_tut_tsref and "UserAgent" example).</p> <pre>[vendor= = Nokia] lib.xtest.value = It is Nokia [vendor= Acer*] // may be "ACER" or "ACER Corporation" or "Acer Incorporated" [end]</pre>
model	string	<p>Mobile device model like Blackstone, N95, 6630... You can use wildcard * or PCRE Regular Expressions (see doc_tut_tsref and "UserAgent" example).</p>
screen_width	integer	<p>Screen width in pixels. You may use integer operators to compare screen sizes.</p> <pre>[screen_width= >= 240]</pre>
screen_height	integer	<p>Screen height in pixels. You may use integer operators to compare screen sizes.</p>
browser_name	string	<p>Mobile browser name You can use wildcard * or PCRE Regular Expressions (see doc_tut_tsref and "UserAgent" example).</p>
browser_version	string	<p>Mobile browser version. Because some browser versions are not only numeric simple string operator may be used here.</p> <pre>[browser_version= = 3.8]</pre>
frames	boolean	<p>True when device supports HTML frames.</p> <pre>[frames= = 1] // The browser supports frames [end]</pre>
html_tables	boolean	True when device supports HTML tables
java	boolean	True when device supports Java
javascript	boolean	True when device supports JavaScript
html_version	string	<p>Supported HTML version. Numeric compare operators are allowed.</p> <pre>[html_version= >= 4.0]</pre>
xhtml_version	string	Supported XHTML version
os	string	<p>Device OS like Android, Symbian, Windows Mobile, ... You can use wildcard * or PCRE Regular Expressions (see doc_tut_tsref and "UserAgent" example).</p>
os_version	string	<p>OS version. Numeric compare operators are allowed.</p> <pre>[os_version= >= 2.0]</pre>
os_vendor	string	<p>OS vendor. You can use wildcard * or PCRE Regular Expressions (see doc_tut_tsref and "UserAgent" example).</p>
gif	boolean	True when device supports GIF images.
jpg	boolean	True when device supports JPG images.
png	boolean	True when device supports PNG images.
call_str	string	<p>String that may be used to make phone calls directly from the site.</p> <p>Note: This is not implemented yet.</p>

Examples

You can set global image size on the mobile site.

```
387: # Maksymalne wymiary obrazków -
388: styles.content.imgtext.maxW = 400
389: styles.content.imgtext.maxWInText = 400
390: styles.content.imgtext.linkwrap.width = 400
391: [screen_width= < 220]
392: styles.content.imgtext.maxW = 160
393: styles.content.imgtext.maxWInText = 160
394: styles.content.imgtext.linkwrap.width = 160
395: [screen_width= >= 220]
396: styles.content.imgtext.maxW = 220
397: styles.content.imgtext.maxWInText = 220
398: styles.content.imgtext.linkwrap.width = 220
399: [screen_width= >= 300]
400: styles.content.imgtext.maxW = 300
401: styles.content.imgtext.maxWInText = 300
402: styles.content.imgtext.linkwrap.width = 300
403: [screen_width= >= 400]
404: styles.content.imgtext.maxW = 400
405: styles.content.imgtext.maxWInText = 400
406: styles.content.imgtext.linkwrap.width = 400
407: [end]
```

HTML headers:

```
3148: # Markup wynikowy
3149: [xhtml_version= = 1.0]
3150: config.doctype = xhtml_trans
3151: [xhtml_version= >= 1.1]
3152: config.doctype = xhtml_11
3153: [end]
3154:
3155:
3156: [device= mobi]
3157: page.headerData.200 = TEXT
3158: page.headerData.200.insertData=1
3159: page.headerData.200.wrap = <meta name="HandheldFriendly" content="|" />
3160: page.headerData.200.value = True
3161: [end]
3162:
```

```

3163:
3164: page.headerData.201 = TEXT
3165: page.headerData.201.insertData=1
3166: page.headerData.201.wrap = <meta name="viewport" content="width=|, user-scalable=yes, ini
3167: page.headerData.201.value = 400
3168: [screen_width= < 220]
3169: page.headerData.201.value = 160
3170: [screen_width= >= 220]
3171: page.headerData.201.value = 220
3172: [screen_width= >= 300]
3173: page.headerData.201.value = 300
3174: [screen_width= >= 400]
3175: page.headerData.201.value = 400
3176: [screen_width= >= 220] && [browser_name= Microsoft Mobile Explorer]
3177: page.headerData.201.wrap = <meta name="mobileoptimized" content="width=|" />
3178: page.headerData.201.value = 220
3179: [screen_width= >= 300] && [browser_name= Microsoft Mobile Explorer]
3180: page.headerData.201.wrap = <meta name="mobileoptimized" content="width=|" />
3181: page.headerData.201.value = 300
3182: [screen_width= >= 400] && [browser_name= Microsoft Mobile Explorer]
3183: page.headerData.201.wrap = <meta name="mobileoptimized" content="width=|" />
3184: page.headerData.201.value = 400
3185: [end]
3186:

```

Company logo:

Setup:

mobile_company_logo

```

294:
295: [GLOBAL]
296: temp.dnbBanner = IMAGE
297: temp.dnbBanner.file = {$filepaths.logos}bmlogo400.gif
298: [jpg= 1] && [screen_width= < 220]
299: temp.dnbBanner.file = {$filepaths.logos}bmlogo160.jpg
300: [png= 1] && [screen_width= < 220]
301: temp.dnbBanner.file = {$filepaths.logos}bmlogo160.png
302: [gif= 1] && [screen_width= < 220]
303: temp.dnbBanner.file = {$filepaths.logos}bmlogo160.gif
304: [jpg= 1] && [screen_width= >= 220]
305: temp.dnbBanner.file = {$filepaths.logos}bmlogo220.jpg
306: [png= 1] && [screen_width= >= 220]
307: temp.dnbBanner.file = {$filepaths.logos}bmlogo220.png
308: [gif= 1] && [screen_width= >= 220]
309: temp.dnbBanner.file = {$filepaths.logos}bmlogo220.gif
310: [jpg= 1] && [screen_width= >= 300]
311: temp.dnbBanner.file = {$filepaths.logos}bmlogo300.jpg
312: [png= 1] && [screen_width= >= 300]
313: temp.dnbBanner.file = {$filepaths.logos}bmlogo300.png
314: [gif= 1] && [screen_width= >= 300]
315: temp.dnbBanner.file = {$filepaths.logos}bmlogo300.gif
316: [jpg= 1] && [screen_width= >= 400]
317: temp.dnbBanner.file = {$filepaths.logos}bmlogo400.jpg
318: [png= 1] && [screen_width= >= 400]
319: temp.dnbBanner.file = {$filepaths.logos}bmlogo400.png
320: [gif= 1] && [screen_width= >= 400]
321: temp.dnbBanner.file = {$filepaths.logos}bmlogo400.gif
322: [end]

```

tt_news image sizes:

```

2034: # Maksymalne wymiary obrazków -
2035: plugin.tt_news.singleMaxW = 400
2036: plugin.tt_news.singleMaxH = 300
2037: plugin.tt_news.displayList.image.file.maxW = 200
2038: plugin.tt_news.displayList.image.file.maxH = 150
2039: [screen_width= < 220]
2040: plugin.tt_news.singleMaxW = 160
2041: plugin.tt_news.singleMaxH = 120
2042: plugin.tt_news.displayList.image.file.maxW = 80
2043: plugin.tt_news.displayList.image.file.maxH = 60
2044: [screen_width= >= 220]
2045: plugin.tt_news.singleMaxW = 220
2046: plugin.tt_news.singleMaxH = 150
2047: plugin.tt_news.displayList.image.file.maxW = 110
2048: plugin.tt_news.displayList.image.file.maxH = 75
2049: [screen_width= >= 300]
2050: plugin.tt_news.singleMaxW = 300
2051: plugin.tt_news.singleMaxH = 235
2052: plugin.tt_news.displayList.image.file.maxW = 150
2053: plugin.tt_news.displayList.image.file.maxH = 117
2054: [screen_width= >= 400]
2055: plugin.tt_news.singleMaxW = 400
2056: plugin.tt_news.singleMaxH = 300
2057: plugin.tt_news.displayList.image.file.maxW = 200
2058: plugin.tt_news.displayList.image.file.maxH = 150
2059: [end]

```

Other conditions for particular devices:

```

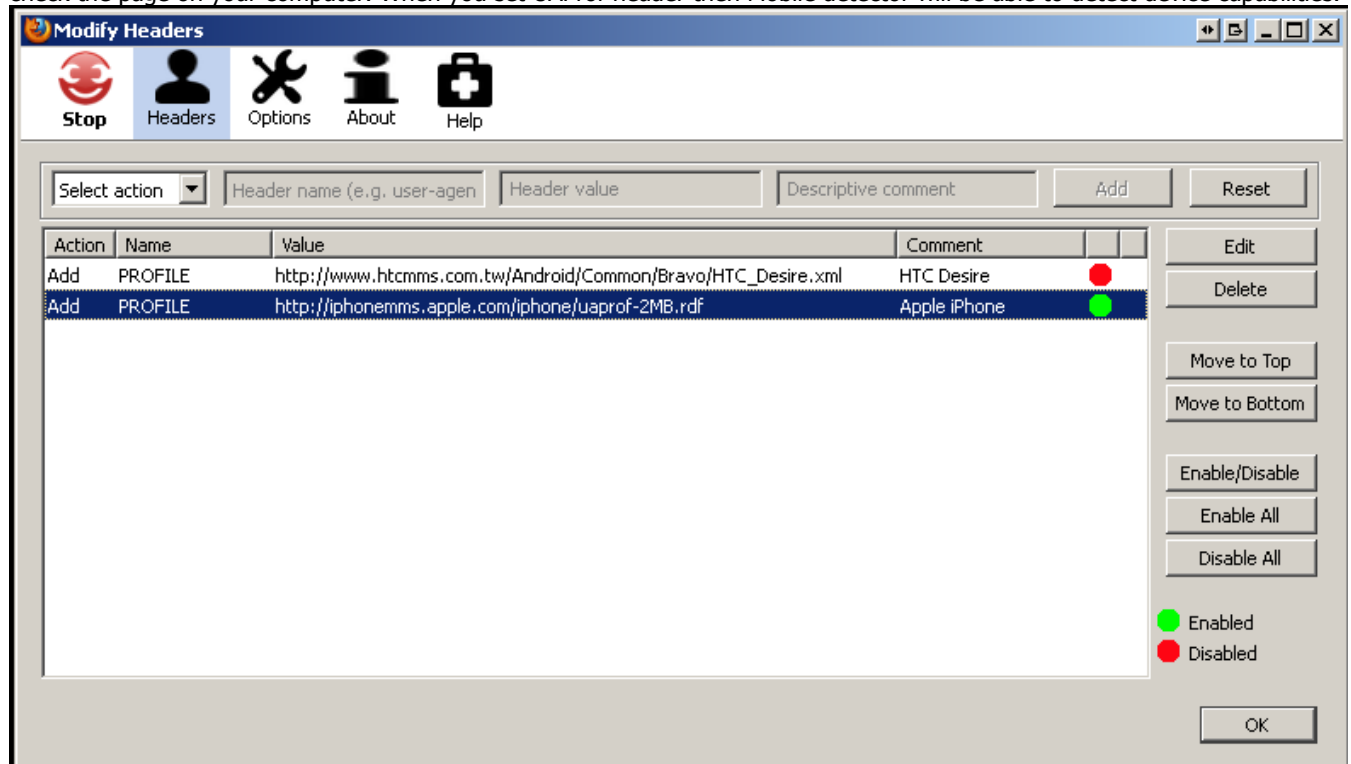
6: # topLeft
7: lib.topLeft = TEXT
8: lib.topLeft.value = It is not the HTC Apache
9: [vendor= HTC] && [model= *Apache*]
0: lib.topLeft.value = It is the HTC Apache !!!

```

Mobile site debugging

Firefox plugin - Modify headers

During mobile site creation it is really helpful to use Firefox plugin called "Modify headers". Using this plugin you will be able to check the page on your computer. When you set UAProf header then Mobile detector will be able to detect device capabilities.



FE Plugin "Mobile detector – tester"

This plugin shows a few tables with device detection results. The first table "Browser info" presents data from UAProf header.

Known problems

- none so far

To-Do list

- Webserver for exchange device capabilities among all sites using this extension.
- Scheduler class for exchange devices information with capabilities server.
- call_str – generation of links type `Call me!` or `Call me!`