EXT: URL shortener

Extension Key: tinyurls
Language: en
Version: 0.0.1
Keywords:
Copyright 2012, Alexander Stehlik, <alexander.stehlik.deleteme@gmail.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

Table of Contents

Introduction

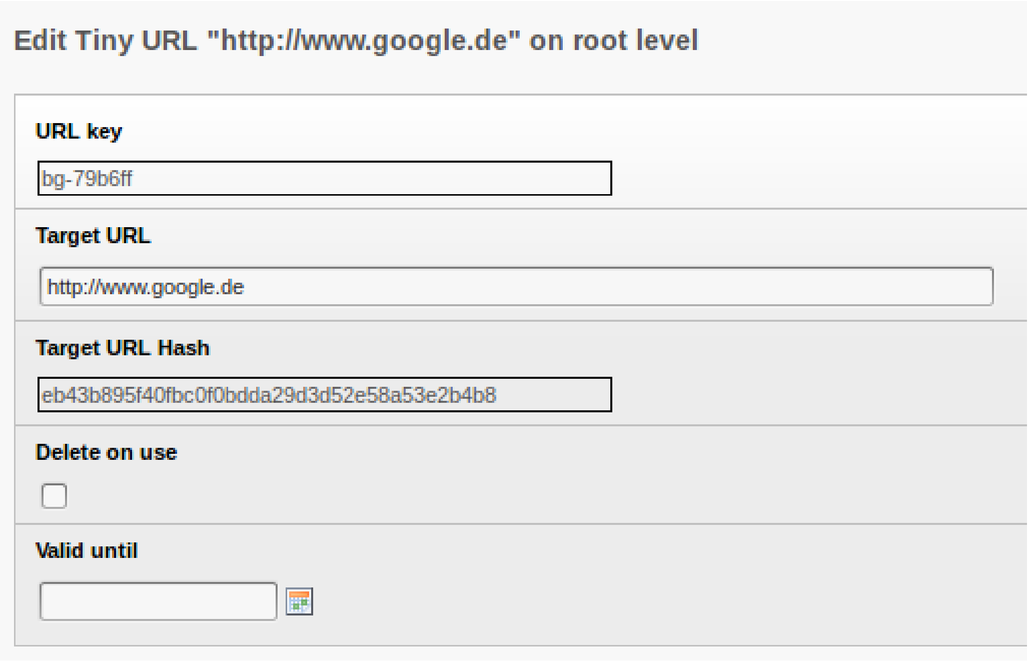
What does it do?

It converts a normal URL to a tiny URL. It works similar as services provided by https://bitly.com/ or http://tinyurl.com/.

It has multiple integration options. You can create tiny URLs in the backend by adding a tiny URL database record. You can convert any typolink to a tinyurl by simply adding some configuration options. And you can use the API of this extension to create tiny URLs in your own extensions.

Screenshots

Settings for a tiny URL in the Backend;



Edit Tiny URL "http://www.google.de" on root level

URL key

bg-79b6ff

Target URL

http://www.google.de

Target URL Hash

eb43b895f40fbc0f0bdda29d3d52e58a53e2b4b8

Delete on use

□

Valid until

Administration

The first step, to get this extension up and running is importing it in the extension manager and creating the required database tables.

Then you can edit the extension configuration according to your needs.

Now you are able to create and use tiny URLs in your TYPO3 instance.

Tiny URL configuration

A tiny URL key consists of two parts:

1. a base62 encoded integer based on a custom dictionary
2. and an optional random part, seperated by a dash

A URL key might look like this:

```
Aefc-3E
```

The first part is generated from the UID of the tiny URL database record which is converted to a base62 integer.

The second part (after the dash) is simply a random hexadecimal string.

In the extension configuration you can edit two options that influence the tiny URL key generation. If **minimalTinyurlKeyLength** is set the URL key must have at least this amount of characters. If the base62 part is shorter the missing characters will be appended. An example:

- minimalTinyurlKeyLength is 6
- the generated base62 part is Aefc
- the URL generator will add two more characters after a dash
- if the minimalTinyurlKeyLength was only 4 no more characters would have been added

Another option is the **minimalRandomKeyLength**. If this is set to a value greater than zero the defined number of random characters will *always* be appended to the base62 part. Another example:

- minimalTinyurlKeyLength is 4
- minimalRandomKeyLength is 3
- the generated base62 part is Aefc
- the URL generator will add *three more characters* after a dash

Speaking URL configuration

By default, a tiny URL created by this extension will look like this:

http://mytypo3page.tld/index.php?eID=tx_tinyurls&tx_tinyurls[key]=Aefc-3E

This is not very short and readable. This is why using speaking URLs is recommended. For this to work you need to set **createSpeakingURLs** to 1 and maybe edit the **speakingUrlTemplate**. By default, speaking URLs will look like this:

http://mytypo3page.tld/goto/Aefc-3E

Obviously you will need to tell you webserver how to handle these URLs.

Apache example configuration

When you are using mod_rewrite you can use this line in your .htaccess file or your webserver configuration:

```
RewriteRule ^tinyurl/([a-zA-Z0-9]+(\-[a-zA-Z0-9]+)?)$ /index.php?
eID=tx_tinyurls&tx_tinyurls[key]=$1
```

Lighttpd example configuration

When you use lighttpd you can use this configuration for rewriting tiny URLs:

```
url.rewrite-if-not-file = (

    # rewrite goto urls to tinyurls extension
  "^/tinyurl/(.*)$" => typo3path + "index.php?eID=tx_tinyurls&tx_tinyurls[key]=$1",
)
```

Editing tiny URL records

Once the extension is installed you can add tiny URLs in the TYPO3 backend. If you didn't change the default value of **urlRecordStoragePID** in the extension configuration you must add the records in the TYPO3 root. Otherwise you can add them in the page (folder) with the PID you specified in the config.

Using tinyurls in TypoScript

You can convert any typolink to a tinyurl by simply setting the property tinyurl to 1. For more information about the TypoScript configuration options please have a look in the TypoScript configuration section.

This is a quick example:

```
page.30 = TEXT
page.30 {
      value = Permalink
      typolink = 1
      typolink.parameter.data = getIndpEnv:REQUEST_URI
      typolink.tinyurl = 1
}
```

Using tinyurls in your extension

When you want to generate tiny URLs in your own extension you can use the `Tx_Tinyurls_TinyUrl_Api` class. Please have a look at the PHPDoc annotations for further information. This is a quick example how to use it:

```
$tinyUrlApi = t3lib_div::makeInstance('Tx_Tinyurls_TinyUrl_Api');
$tinyUrlApi->setDeleteOnUse(1);
$tinyUrlApi->setUrlKey($myKey);
$tinyUrlApi->setValidUntil($validUntil);
$myTinyUrl = $tinyUrlApi->getTinyUrl($url);
```

Configuration

Most of the configuration is done in the extension manager. At the moment there are also a few TypoScript options available.

Extension configuration

Property:	Data type:	Description:	Default:
createSpeakingURLs	int	If true (1) then the tiny URL will be generated depending on the speakingUrlTemplate. Hint! If you enable this you might need to add a rewrite rule to you webserver!	0
speakingUrlTemplate	string	The template that is used for creating a speaking URL (only relevant if createSpeakingURLs is set to 1). You can use all available keys for t3lib_div::getIndpEnv() as template markers (e.g. ###TYPO3_SITE_URL### and the ###TINY_URL_KEY### template marker will be replaced with the shortened URL key	See [1]
base62Dictionary	string	Dictionary for creating bas 62 based integers (see http://jeremygibbs.com/2012/01/16/how-to-make-a-url-shortener)), use random string to increase security (e.g. http://textmechanic.com/String-Randomizer.html)	See [2]
minimalRandomKeyLength	int	The minimum length that the random part of the tiny URL must have	2
MinimalTinyurlKeyLength	int	The minimum length that the whole tiny URL key must have	8
urlRecordStoragePID	int	The PID where the tiny URL records are stored, use 0 to store them in the TYPO3 root	0

TypoScript configuration

typolink configuration

This additional property is available for typolink objects when the extension is installed:

Property:	Data type:	Description:	Default:
tinyurl	int / tinyurl	If you set this to 1 the final typolink URL is converted to a tiny URL. You an set some configuration options for the tiny URL generator int the tinyurl. namespace.	0

tinyurl configuration

You can modify the URL generation process with these options:

[1] ###TYPO3_SITE_URL###tinyurl/###TINY_URL_KEY###
[2] abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789

Property:	Data type:	Description:	Default:
deleteOnUse	int / stdWrap	If this is is true, the tiny URL is deleted from the database on the first hit. This makes the most sense when used together with the urlKey property.	0
validUntil	int / stdWrap	The timestamp until the URL is valid. If this is set to 0 the URL will never be invalid.	0
urlKey	string / stdWrap	Normally the URL key is generated automatically. Here you can set you own unique urlKey. This may be used together with the deleteOnUse property for creating one-time valid URLs, e.g. for activating a subscription.	0

Known problems

Please be aware that this is a very young and **not** heavily tested extension. The there might be problems not yet detected. Testing and reporting problems is highly appreciated.

To-Do list

You and find the roadmap in the forge project of this extension.

ChangeLog

You and find the change log in the forge project of this extension.