

TYPO3-Domains

by *Bernd Warken* at *netcos AG*

Chapter 0: Introduction	3
<i>0.1 Preface</i>	3
<i>0.2 License</i>	3
Chapter 1: Webserver Configuration	4
<i>1.1 Configure an URL in the Directory sites-available</i>	4
<i>1.2 Link in sites-enabled and Reload of Apache</i>	4
<i>1.3 The File hosts</i>	5
<i>1.4 The File .htaccess</i>	5
Chapter 2: Configuration of TYPO3 for different domains	7
<i>2.1 Configuration of the Domain</i>	7
<i>2.2 Configuration of Templates</i>	7
<i>2.3 The File localconf.php</i>	7
Chapter 3: Domains as Redirection to some Website	9
Chapter 4: Several Domains on the same place	10
<i>4.1 Configuration of Apache</i>	10
<i>4.2 Configuring Domains</i>	10
<i>4.3 Deactivate RealURL</i>	11
<i>4.4 Using RealURL</i>	11
Chapter 5: Overview	13

Chapter 0: Introduction

0.1 Preface

Within a single TYPO3 installation, you can write different code for several websites at the same time. Each website has a page in TYPO3 that serves as a folder for the source code of this website. Each website gets thereby a *domain* name as data record in the list module. After some configurations, you then can announce each website via its *domain* name and exclusively access this website by its name.

It is also possible to assign different website names for the same piece of code. In this process, several *domain* names are attributed to the same TYPO3 page.

0.2 License

This document is saved as a file called **domains_en.sxw**. The actual version is **1.1** of 10 September 2007. Author is **Bernd Warken** with support of **Sebastian Roth**, both working at [netcos AG](http://netcos.de)..

The document and the source codes are under the free licence **GPL (GNU General Public License)** version **3**. The original text in English is available under <http://www.gnu.org/licenses/gpl.html>.

Chapter 1: Webserver Configuration

In this chapter, we assume that the TYPO3 installation is placed on a *Linux Debian* computer with the web server *Apache2*. The installation on other operating systems is similar. For each operating system, the development with the TYPO3 backend can be done via the network or the internet on each operating system using an arbitrary web browser.

1.1 Configure an URL in the Directory sites-available

The URL for a *domain* can be some official online URL or some local URL that is freely chosen. In both cases, the *Apache* installation is the same. In *Apache*, a URL is always described as a server. For each of these servers, one creates a file in `/etc/apache2/sites-available` with its data information. As an example, we construct the *domain* www.freely.invented on a computer with the IP **192.168.1.1**. For that, we create a file `/etc/apache2/sites-available/freely_invented`:

```
<VirtualHost 192.168.1.1:80>
  ServerName www.freely.invented
  ServerAlias freely.invented
  DocumentRoot /typo3/mysite
</VirtualHost>
```

That means that on the computer with the IP-Adresse **192.168.1.1** under the Port **80** a new *Apache* server with the name www.freely.invented was created. This server can also be addressed by the alias **freely.invented**. The root site of this server is located at the directory `/typo3/mysite/`; this directory contains the file **index.php** of the TYPO3 installation. This directory can be completely different from the default directory for <http://localhost/>.

As an additional example, we create another website www.other.website. It serves as additional *doamin* in *TYPO3* and accesses a different source code part. Again we create a file `/etc/apache2/sites-available/other.website`.

```
<VirtualHost 192.168.1.1:80>
  ServerName www.other.website
  ServerAlias other.website
  DocumentRoot /typo3/mysite
</VirtualHost>
```

1.2 Link in sites-enabled and Reload of Apache

The configuration of this file is not yet enough. For *Apache* does not read the informations in `/etc/apache2/sites-available`, but only the files in `/etc/apache2/sites-enabled`. Hence just a link on the generated files will be created here:

```
cd /etc/apache2/sites-enabled
ln -s /etc/apache2/sites-available/freely_invented
ln -s /etc/apache2/sites-available/other_website
```

After this, *Apache* must be restarted:

```
/etc/init.d/apache2 restart
```

Now you can check whether the new servers were adopted by calling the following command:

```
apache2 -t -D DUMP_VHOSTS
```

Some other operating systems have the command **httpd2** instead of **apache2**.

Now the new *domain* URLs are available for the web server, but you still cannot access them.

1.3 The file hosts

In *Linux*, the file **hosts** is located in the directory **/etc/**; in Microsoft Windows, it is found under **C:\WINDOWS\system32\drivers\etc**. In this file, the new URLs must be entered, except when these are official internet URLs. So it is not necessary to enter a URL like www.google.com, but www.freely.invented must be done. The entry for our *domain* URLs looks like

```
192.168.1.1 www.freely.invented freely.invented  
192.168.1.1 www.other.website other.website
```

These entries must be entered on each computer with access to this *domain*.

This *IP* address is only valid for the local network. It is only possible to access it from the outside if this computer is officially included within the internet. Then it must have an additional *IP* different from **192.168.*.*** for the internet access. For computers external to the local network, this *IP* must be included into the file **hosts** with the invented name. As an example, assume that the computer with the local *IP* **192.168.1.1** has an additional official *IP* **82.137.54.38**. Then it is possible to include the invented websites as

```
82.137.54.38 www.freely.invented freely.invented  
82.137.54.38 www.other.website other.website
```

This allows access from any computer external to the network.

1.4 The File .htaccess

You should activate the *Rewrite* module of *Apache*. This is the base for the creation of domains. This is usually done by installing a suitable package in your operating system. In *Linux Debian*, you can verify the installation of the modul if **/etc/apache2/mods-enabled/rewrite.load** is a valid link.

In the *Apache* installation, we fixed the root directory for the new *domain* URLs to **/typo3/mysite/**. If the file **.htaccess** does not exist in this root directory you should rename the file **_.htaccess** into **.htaccess** and configure it. *Rewrite* should there be activated by the line

```
RewriteEngine On
```

Now all domains must be accessible by two lines in **.htaccess**. For our *domain* URLs, this looks like

```
RewriteCond %{HTTP_HOST} freely.invented$ [NC]
RewriteRule ^$ /index.php?id=1
RewriteCond %{HTTP_HOST} other.website$ [NC]
RewriteRule ^$ /index.php?id=100
```

Instead of the numbers **1** and **100**, the right *ids* of the TYPO3 pages with the root page for the *domains* must be inserted. This *id* can be read in the backend within the module *Page*.

Chapter 2: Configuration of TYPO3 for different domains

We assume here that TYPO3 is already installed and basically configured.

2.1 Configuration of the Domain

For each *domain* that should be used internally by the TYPO3 installation there should be created a single page as *Shortcut*. This *Shortcut* serves as folder for all pages that are part of the *domain*. This page is also called the root page of the *domain*.

We use the module *List*, choose as page one of the *domain* root pages, and create a new record *Domain*. As *Domain* name, we insert the short form, in our example **freely.invented**. To apply its long name as well, we choose *Show secondary options (palettes)* in the checkboxes below. Then a new part *Redirect to:* is generated. There we insert the full URL <http://www.freely.invented>. We can also choose the checkbox *Transfer parameters to Redirect URL:*. With that, the domain is configured.

We apply these steps for all *domains*, in our example also for **other.website**.

2.2 Configuration of Templates

We propose that you use *RealURL*. Then it is necessary to newly set the template parameter **config.baseURL** for each domain.

To achieve that, go to the root page of a *domain* in the TYPO3 module *Template*. There choose *Setup*.

Therein provide the full *domain* name as value for **config.baseURL**, i.e. in our example

```
config.baseURL = http://www.freely.invented/
```

At the root page of the second *domain*, you configure

```
config.baseURL = http://www.other.website/
```

2.3 The File *localconf.php*

The last thing to do is the configuration of the file **typo3conf/localconf.php** for *RealURL*. Two lines must here be appended for each *Domain*, but only when an extension for *RealURL* was implemented in this file. This can be traced by the existing line:

```
$TYPO3_CONF_VARS['EXTCONF']['realurl'] = array(
```

The lines are appended near the end of the file, but before the final line

```
?>
```

The two lines to be appended are:

```
$TYPO3_CONF_VARS['EXTCONF']['realurl']['domain-name'] =
$TYPO3_CONF_VARS['EXTCONF']['realurl']['_DEFAULT'];
$TYPO3_CONF_VARS['EXTCONF']['realurl']['domain-name']['pagePath']['rootpage_id'] = 1;
```

Please remove the line breaks in the first line. Instead of the number **1**, the real *id* for the root side of the *domain* must be inserted. This number can be read in the backend using the module *Page*.

For our two example websites, the lines look like:

```
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.freely.invented'] =
$TYPO3_CONF_VARS['EXTCONF']['realurl']['_DEFAULT'];
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.freely.invented']['pagePath']['rootpage_id']
= 1;
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.other.website'] =
$TYPO3_CONF_VARS['EXTCONF']['realurl']['_DEFAULT'];
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.other.website']['pagePath']['rootpage_id'] =
100;
```

Please remove the line breaks around the = signs and reset the two numbers **1** and **100** by the root page *id* of the domains.

After having done this, the system is completely configured for the *domain*. Now you can access the frontend with <http://freely.invented> or <http://www.freely.invented> in each web browser. The backend can be reached by <http://freely.invented/typo3> or <http://www.freely.invented/typo3>. Correspondingly, the second address can be reached by <http://other.website> or <http://www.other.website>. For the backend, the additional sites <http://other.website/typo3> or <http://www.other.website/typo3> are available, but it is not necessary to use them.

All *TYPO3* functions of the *domain* should now work without problems.

Chapter 3: Domains as Redirection to some Website

The simplest possibility for different *domains* is to insert some website as main site, e.g. <http://www.my.website>, and redirect another *domain* to this site, such as <http://www.redirected.website>. A call of <http://www.redirected.website> would then immediately be replaced by <http://www.my.website>.

The configuration of this concept is very easy. Only the main site needs to be configured as in the chapters 1 and 2. The *Apache* configuration of the redirected website reduces to the description of **ServerAlias**.

```
<VirtualHost 192.168.1.1:80>
  ServerName www.my.website
  ServerAlias my.website redirected.website www.redirected.website
  DocumentRoot /typo3/mysite
</VirtualHost>
```

Also the **hosts** file must eventually be configured, but not the *Rewrite* engine in **.htaccess**.

Then the redirected website must be specified as a *domain* that is really redirected to the main site. On the root page of the main *domain*, we go to the module *List* and *Create a new record domain*. In the field *Domain*, we specify **redirected.website**. In order to have displayed the additional field with the redirection, the field *Show secondary options (palettes)* must be activated. In the now arising field *Redirect to:* we specify the main *domain* <http://www.my.website>.

It is recommended that the creation of the *domain* is repeated for **www.redirected.website**.

That's already all, the adaptation of the templates and of **localconf.php** is not necessary.

Chapter 4: Several Domains on the same place

In this chapter, we access the same code page with several website names. This is different from the previous chapter by not redirecting any site. All websites stay independently.

Let's assume that we want to access the TYPO3 page **websites** both with <http://www.first.website> and <http://www.second.website>. Then the access to each websites is resident without indirection.

4.1 Configuration of Apache

The configuration of the web server is identical to chapter 1. For both websites, we generate a file in **/etc/apache2/sites-available**.

```
<VirtualHost 192.168.1.1:80>
  ServerName www.first.website
  ServerAlias first.website
  DocumentRoot /typo3/meineseite
</VirtualHost>
```

and

```
<VirtualHost 192.168.1.1:80>
  ServerName www.second.website
  ServerAlias second.website
  DocumentRoot /typo3/meineseite
</VirtualHost>
```

For both of these files, we create a link in **/etc/apache2/sites/enabled** and restart the web server *Apache*. We append the new website names in **/etc/hosts**

```
192.168.1.1 www.first.website first.website
192.168.1.1 www.second.website second.website
```

We append the following lines to the file **.htaccess** in the root directory of the TYPO3 installation

```
RewriteEngine On
RewriteCond %{HTTP_HOST} first.website$ [NC]
RewriteCond %{HTTP_HOST} second.website$ [NC]
RewriteRule ^$ /index.php?id=1
```

Again the *id* number must be changed to the right value. As both websites, however, show to the same TYPO3 page the **RewriteRule** with the *id* must only given once.

4.2 Configuring Domains

To configure both websites on one page, we choose the module *List* with the page **websites** in the backend. On this page, we create two *domains* for our two websites. As in chapter 2.1, we generate a new record *Domain* for each. Again we provide the short form **first.website** and **second.website** and add the long form <http://www.first.website> and <http://www.second.website> in the field *Redirect to*.

Now we can access the same source code with all of these names.

4.3 Deactivate RealURL

The simplest possibility is now to deactivate *RealURL*, but that is a pity. This deactivating occurs by commenting out two lines in the *TypoScript template*. This is done by prefixing each line with a # character.

```
# config.baseURL = http://...  
# config.tx_realurl_enable = 1
```

Alternatively **config.tx_realurl_enable** can also be set to **0**.

```
config.tx_realurl_enable = 0
```

But the parameter **config.baseURL** cannot be applied because both websites would disturb each other.

By this process, *RealURL* is completely deactivated when no other effects are used. But you can now access your source code from each browser using <http://www.first.website> and <http://www.second.website>.

4.4 Using RealURL

If *RealURL* shall stay activated then the variable **config.baseURL** must be set to several values at one place. That leads to problems. This variable must show to the corresponding URL the website is accessed. Fortunately *TypoScript* has conditions to get this.

As first measure, a *TypoScript* constant is created for each website URL. In our example, this would look like

```
domain_wwwfirst = www.first.website  
domain_wwwsecond = www.second.website
```

Please note that this must be done in the constant field.

Then, in the *Setup* of the template, the variable **config.baseURL** is set to a default value using one of the constants, e.g.

```
config.baseURL = http://{domain_wwwfirst}
```

Next the *TypoScript* condition is used to set the value of the second constant for the other URL:

```
[globalString = ENV:HTTP_HOST=*{domain_wwwsecond}]  
config.baseURL = http://{domain_wwwsecond}/  
[global]
```

This says that, if the starting URL in the browser relies on the value of the second constant then set the variable to the second language.

Now **config.tx_realurl_enable** must just be set to **1**.

```
config.tx_realurl_enable = 1
```

Finally the file **typo3conf/localconf.php** must be configured for the URLs. This happens analogously to chapter 2.3. But now all URLs rely on the same *id* because all names show to the same TYPO3 page. In our example this looks like

```
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.first.website'] =
$TYPO3_CONF_VARS['EXTCONF']['realurl']['_DEFAULT'];
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.first.website']['pagePath']['rootpage_id'] =
1;
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.second.website'] =
$TYPO3_CONF_VARS['EXTCONF']['realurl']['_DEFAULT'];
$TYPO3_CONF_VARS['EXTCONF']['realurl']['www.second.website']['pagePath']['rootpage_id']
= 1;
```

Again please remove the line breaks and replace the value for **rootpage_id** by the right *id* of the root side of the *domains*.

Now the configuration of several *domains* to the same page is complete. On each browser, you can now access the same source code using <http://www.first.website> and <http://www.second.website>.

Chapter 5: Overview

To configure *domains*, it is necessary to follow the following points.

- Configure *Apache*
 - For each *domain*, create a file in **/etc/apache2/sites-available**.
 - On these, generate links in **/etc/apache2/sites-enabled**.
 - Append the new URLs with *Rewrite* in **.htaccess**.
 - If necessary extend the file **hosts** on each client by the URLs.
- Create *domains* in TYPO3.
- Deactivate *RealURL* or apply the following points.
 - Activate *RealURL*.
 - Configure the file **typo3conf/localconf.php** for *RealURL*.
 - Generate constants for **config.baseURL**.