

Typo3插件改名之tt_news

作者: Zhao Zhang

邮件:zo.zhang@gmail.com

网址:<http://www.fainz.info>

▼ WEB

Page

View

List

Info

Access

Functions

Versioning

Template

▼ FILE

Filelist

▼ USER TOOLS

Task center

User settings

Workspace

▼ ADMIN TOOLS

User Admin

Ext Manager

DB check

Configuration

Install

Log

Reports

▼ HELP

About Modules

TypoScript Help

Import extensions

2

Extension Manager

Group by: Category

Show: Details

Show obsolete: ☐

Extensions in TYPO3 Extension Repository

Look up extensions:

Look up

Connect to the current mirror and retrieve the current extensions

Retrieve/Update

Did you find a security problem?

Please be aware that extensions are third-party extensions.

If you think you have found a security issue in an extension, please report it to the TYPO3 Security Team.

PRIVACY NOTICE:

When you interact with the online repository, server cookies are used to improve your experience.

Upload extension file directly (.t3x):

Upload extension file (.t3x):

tt_news_3.0.1.t3x

...to location:

Local (../typo3conf/ext/)

☐ Overwrite any existing extension!

Upload extension file

为了避免插件改名后不能在后台插件管理器中找到插件，首先使用外部导入方式，将插件导入到typ3conf/ext目录下，当出现安装数据库时，不要进行选择；

1

2

3

▼ WEB

- Page
- View
- List
- Info
- Access
- Functions
- Versioning
- Template

▼ FILE

- Filelist

▼ USER TOOLS

- Task center
- User settings
- Workspace

▼ ADMIN TOOLS

- User Admin
- Ext Manager

- DB check
- Configuration
- Install
- Log
- Reports

▼ HELP

- About Modules
- TypoScript Help



Information

Extension Manager

Extension: **News** (tt_news)

 Installing News: DATABASE NEEDS TO BE UPDATED

Before the extension can be installed the database needs to be updated with new tables or fields.
Please select which operations to perform:

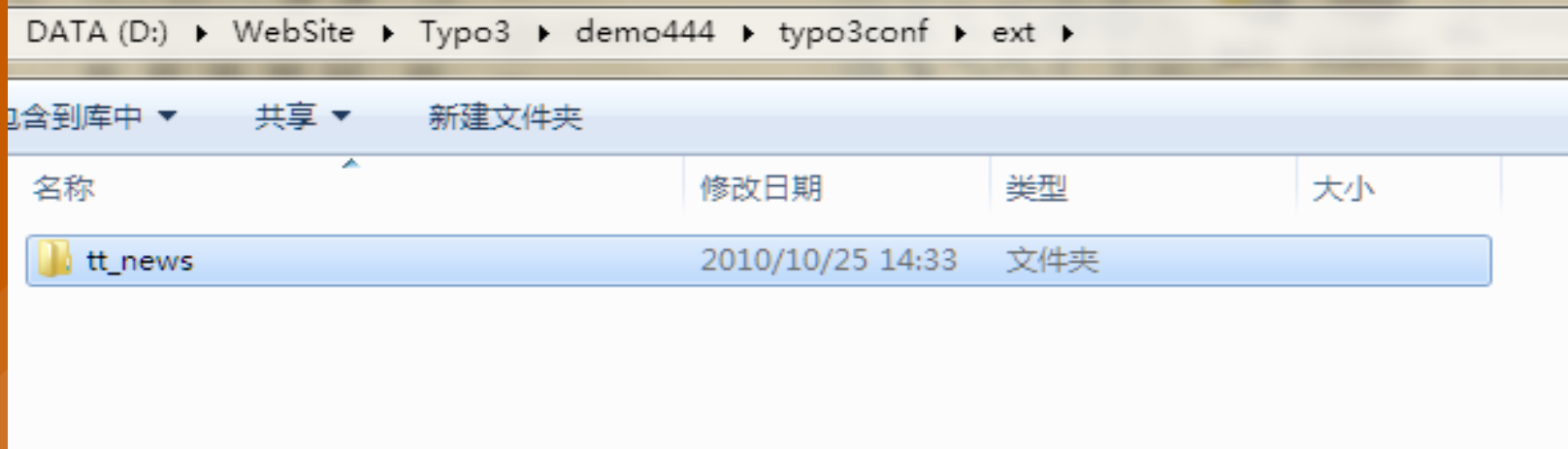
Add fields

- ☒ ALTER TABLE be_users ADD tt_news_categorymounts varchar(255) NOT NULL default '';

Add tables

- ☒ **select/deselect all**
- ☒ CREATE TABLE tt_news (
 - uid int(11) NOT NULL auto_increment,
 - pid int(11) NOT NULL default '0',
 - tstamp int(11) unsigned NOT NULL default '0',
 - crdate int(11) unsigned NOT NULL default '0',
 - cruser_id int(11) unsigned NOT NULL default '0',
 - editlock tinyint(4) unsigned NOT NULL default '0',
 - deleted tinyint(3) unsigned NOT NULL default '0',
 - hidden tinyint(4) unsigned NOT NULL default '0',
 - starttime int(11) unsigned NOT NULL default '0',
 - endtime int(11) unsigned NOT NULL default '0',
 - fe_group varchar(100) NOT NULL default '0',
 - title text,
 - datetime int(11) unsigned NOT NULL default '0',
 - image text,
 - imagecaption text,
 - imagealttext text,
 - imagetitletext text,
 - related int(11) NOT NULL default '0',
 - short text,
 - bodytext mediumtext,
 - author varchar(255) NOT NULL default ''
 - author_email varchar(255) NOT NULL default ''
 - category int(11) NOT NULL default '0',
 - news_files text,
 - links text,

出现这个介面的时候不要进行下一步，这时返回插件目录；



这时候在**typo3conf/ext**目录下可以看到刚才导入的**tt_news**插件目录,首先把**tt_news**这个目录名称改为你想要的名称,然后使用工具替换该目录下文件名称和文件内容;

由于大多数编辑工具,都只能替换文件内容,不能替换文件名称,所以使用**php**写了一个替换文件名称和文件内容的小工具,使用方法是在目录里填上要替换内容的目录,然后在查找中输入要替换的名称,在替换那里填入替换后的名称..单击替换,完成后替换后的文件会在下面的列表中以青色显示,在这个例子里面改了如下名称:

tt_news	-->>>>>>>	ed_news	ttnews	-->>>>>>>	ednews
NEWS	-->>>>>>>	EDNEWS	News	-->>>>>>>	edNews

插件作者、公司等信息在插件目录下**ext_emconf.php**文件中修改,如果想修改后台左侧菜单栏上名称可修改插件**mod1/locallang_mod.xml**文件;

<label index="mlang_tabs_tab">edNews Admin</label> 这个对应的是显示的名称

<label index="mlang_labels_tablabel">edNews Admin</label> 这个对应的是鼠标移上去显示的名称

由于TT_News本身不符合Typo3插件命名规范,这样替换完之后,直接使用你会发现前台什么也没有,这时候还需做以下几步:

1、修改插件目录下ext_locaconf.php文件,在defined下面添加下面的代码, 文件名称要按实际的名称填写;

```
// Add PlugIn to Static Template #43 and create USER cObject
```

```
t3lib_extMgm::addPltoST43($_EXTKEY, 'pi1/class.tx_ednews.php', '', 'list_type', 1 );
```

2、把ext_tabs.php文件下例代码中的9改为插件名称, 本例为'ed_news'

```
// remove some fields from the tt_content content element
```

```
$TCA['tt_content']['types']['list']['subtypes_excludelist'][9] =
```

```
'layout,select_key,pages,recursive';
```

```
// add FlexForm field to tt_content
```

```
$TCA['tt_content']['types']['list']['subtypes_addlist'][9] = 'pi_flexform';
```

```
// add ed_news to the "insert plugin" content element (list_type = 9)
```

```
t3lib_extMgm::addPlugin(array('LLL:EXT:ed_news/locallang_tca.xml:ed_news', 9));
```

```
// switch the XML files for the FlexForm depending on if "use StoragePid" (general record  
Storage Page) is set or not.
```

```
if ($confArr['useStoragePid']) {
```

```
t3lib_extMgm::addPiFlexFormValue(9, 'FILE:EXT:ed_news/flexform_ds.xml');
```

```
} else {
```

```
t3lib_extMgm::addPiFlexFormValue(9, 'FILE:EXT:ed_news/flexform_ds_no_sPID.xml');
```

```
}
```

3、修改插件目录下pi/class.tx_xxxx_wizicon.php文件中的9也换成插件名称
'params'=>'&defVals[tt_content][CType]=list&defVals[tt_content][list_type]=ed_news'

4、修改插件目录下的pi/static/ts_xxx/setup.txt，在最后一个[global]下添加一个引用,plugin
的右侧是插件名称除去下划线，前面再加上tx_,引用的名称可以在这个文件的上面查找，
这里是plugin.ed_news
plugin.tx_ednews >
plugin.tx_ednews = < plugin.ed_news

接着返回插件管理器，单击刚修改的插件左边的小图标，进行正常安装，然后添加到页面上看看功能是否正常，如果正常则修改成功;

插件读取TS配置

Typoscript for this content element (plugin.ed_news.[your TS])

```
customer.image {  
  file.maxW = 1000  
  file.maH = 500  
}
```

比如我在Ts中添加了一些自定义的配置，想在插件前台程序中调用,首先要搞清楚插件的入口方法是哪个，前台相关的到pi目录下找，后台在mod、回调在hook目录下,tt_news的Ts中有这样一段代码：

userFunc = tx_ednews->main_news

简单的把它理解为指定入口方法，你会奇怪它是怎么知道tx_ednews这个类，是在哪个文件中？如果你是从头开始看它的Ts就会发现，最上面有这样一段代码：

Includes the newsLib:

includeLibs.ts_news = EXT:ed_news/pi/class.tx_ednews.php

tx_ednews->后面跟的是类文件中的方法，这里使用的是main_news，打开class.tx_ednews.php查找main_news(\$content,\$conf)你会看到有两个参数，第一个参数代表的是内容，第二个是这个插件的配置信息,如果想得到刚才上面写的参数，tt_news需要在\$this->preInit();之后调用\$this->conf['customer.']才能得到刚才添加的值，这时你会发现名称后面多了一个'.',之所以要加这个点，是因为刚才添加的customer后面还有几个子选项，如果是这种形式**customer = xxxx**就不需要在后面加'.';

到这里需要使用Typo3内置的debug来查看数据，开启debug需要在localconf.php中配置两个参数

```
$TYPO3_CONF_VARS['SYS']['displayErrors'] = '2';
```

```
$TYPO3_CONF_VARS['SYS']['devIPmask'] = '*';
```

DisplayErrors 为2时根据**devIPmask**的值匹配客户端IP地址，匹配上的时候才显示错误或调试信息，**devIPmask**为*时不匹配IP地址，直接显示，配置好之后在插件程序中使用**debug()**就可以查看数据是否调用成功或查看调试信息；

模板标签的使用

有时需要改模板中的标签或添加新标签，这个其实挺简单的，拿tt_news为例，我要在模板list视图下添加一个自定义的标签，显示自定义的内容，首先打开模板文件，根据后台插件中设置的显示模式选择对应的标签，这里是LIST,在模板文件中找到第一个####TEMPLATE_LIST#### 在它和第二个相同标签之间写上你想添加的标签比如：####CUSTOMER####,新添加的标签要在左三#和右三#中间；然后在userFunc入口方法中顺着找处理对应显示模式的代码，这里处理LIST使用的是\$this->displayList();到displayList()方法中找\$markerArray=array();的位置，在它之后把刚才在模板中添加的标签加到这个数组中并赋上你想要的值,\$markerArray['####CUSTOMER####'] = 'CUSTOMER SET VALUE!';清空缓存刷新在前台，就可以看到标签起作用了，把某个标签内的结果替换或添加东西也同理，只准位置，简单修改即可

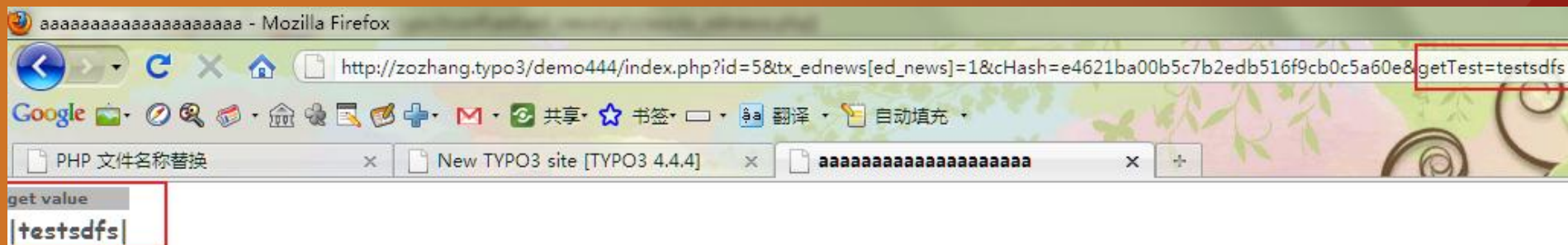
Typo3 内置基本函数介绍

首先介绍一下 `t3lib_div::getIndpEnv('_ARRAY')`; 这是个静态方法，在使用前包含了 `class.t3lib_div.php`, 后就可以调用 `t3lib_div` 这个类中的静态方法，上面那个静态方法主要的功能是获取客户端浏览器的一些信息，`_ARRAY` 参数是代表把所有的信息都取出来，如果不确定要获取的信息用什么参数，可以使用上面那个参数，把所有都取出来，然后在分析自己需要的数据，前面的索引值就是对应的参数，信息如下图：

HTTP_HOST	zozhang.typo3
TYPO3_HOST_ONLY	zozhang.typo3
TYPO3_PORT	
PATH_INFO	
QUERY_STRING	id=5&tx_ednews[ed_news]=1&cHash=e4621ba00b5c7b2edb516f9cb0c5a60e
REQUEST_URI	/demo444/index.php?id=5&tx_ednews[ed_news]=1&cHash=e4621ba00b5c7b2edb516f9cb0c5a60e
HTTP_REFERER	http://zozhang.typo3/demo444/index.php?id=2
TYPO3_REQUEST_HOST	http://zozhang.typo3
TYPO3_REQUEST_URL	http://zozhang.typo3/demo444/index.php?id=5&tx_ednews[ed_news]=1&cHash=e4621ba00b5c7b2edb516f9cb0c5a60e
TYPO3_REQUEST_SCRIPT	http://zozhang.typo3/demo444/index.php
TYPO3_REQUEST_DIR	http://zozhang.typo3/demo444/
TYPO3_SITE_URL	http://zozhang.typo3/demo444/
TYPO3_SITE_SCRIPT	index.php?id=5&tx_ednews[ed_news]=1&cHash=e4621ba00b5c7b2edb516f9cb0c5a60e
TYPO3_SSL	
TYPO3_REV_PROXY	
SCRIPT_NAME	/demo444/index.php
TYPO3_DOCUMENT_ROOT	D:/WebSite/Typo3
SCRIPT_FILENAME	D:/WebSite/Typo3/demo444/index.php
REMOTE_ADDR	127.0.0.1
REMOTE_HOST	
HTTP_USER_AGENT	Mozilla/5.0 (Windows; U; Windows NT 6.1; zh-CN; rv:1.9.2.12) Gecko/20101026 Firefox/3.6.12 GTB7.1
HTTP_ACCEPT_LANGUAGE	zh-cn,zh;q=0.5

使用Typo3内置的方法获取浏览器GET或POST方法传过来的参数,下面代码是在某个插件的入口方法中, 使用t3lib_div静态方法_GP来获取浏览器传递过来的参数, 从该方法的名称上可以看出是GET和POST的简写, 意思就是支持这两种方式的传递

```
$getTest = t3lib_div::_GP('getTest'); //获取浏览器上传传递过来的getTest 的值  
debug($getTest,'get value'); //使用debug查看 是否能获取到值,下面为测试图;
```



t3lib_div::validEmail(); 验证Email是否合法
t3lib_div::isValidUrl(); 验证是否为URL
t3lib_div::validIP(); 验证IP地址是否合法;
t3lib_div::validIPv4(); 验证IP地址是否合法, 并且为IPv4
t3lib_div::validIPv6(); 验证IP地址是否合法, 并且为IPv6
t3lib_div::makeInstance('className'); 实例化一个对象, 要注意要实例化对象类文件一定要在它之前引入;
t3lib_div::clientInfo(); 得到客户端系统, 浏览器等信息,
t3lib_div::getThisUrl(); 得到当前主机+目录, 不包含http://和脚本名称
t3lib_div::debug(); 输出调试信息, t3lib/config_default.php有一个debug的函数(其内部也是调用这个方法) 所以不用写类名就可以用;
t3lib_div::print_array(); 以清晰的格式打印一个数组, 类似 debug();
t3lib_div::getMaximumPathLength(); 获取当前环境支持的最大路径长度
t3lib_div::getMaxUploadFileSize(); 获取最大允许上传的千字节数