

TYPO3 CMS 7.3 - обновления

Нововведения, изменения и исправления

Авторы:

Patrick Lobacher и Michael Schams

29/June/2015

Creative Commons BY-NC-SA 3.0



Обзор глав

Введение

Backend / Внутренний интерфейс
TSconfig и TypoScript

Глубинные изменения

Extbase и Fluid

Устаревшие/удаленные функции

Источники и авторы

Введение

The Facts

TYPO3 CMS 7.3 - факты

- Дата выхода: 16 июня 2015
- Тип: "Sprint Release"
- Направленность: охват, инновации, доступность
- Фокус: экосистема пакетов, Composer и обработка расширений

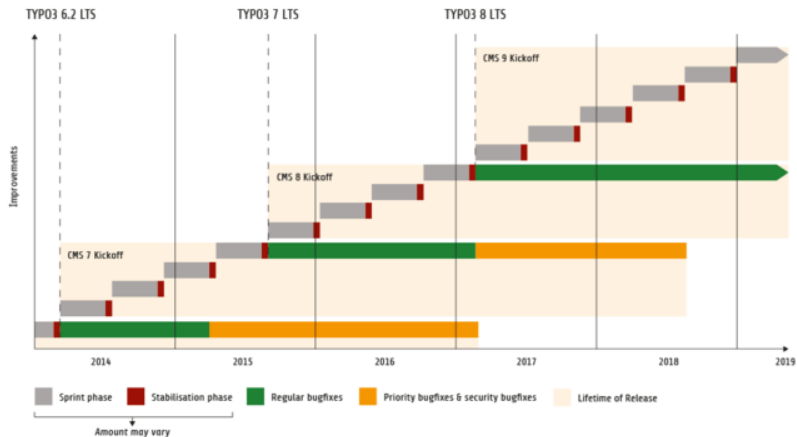
Системные требования

- PHP*: v5.5.0 - v5.6.x
- MySQL: v5.5.x - v5.6.x (no strict mode)
- Дисковое пространство: мин. 200 МБ
- PHP настройки:
 - `memory_limit` \geq 128M
 - `max_execution_time` \geq 240s
 - параметр компиляции `-disable-ipv6` не должен использоваться
- Внутренний интерфейс требует IE \geq 9 или любой другой современный браузер

*) Подробности: [PHP Minimum Requirements for TYPO3 CMS 7](#)

Введение

График разработки и выхода



Примерные даты выхода и их направленность:

- v7.0 02/Dec/2014 Переработка внутреннего интерфейса часть 1
- v7.1 24/Feb/2015 Чистка ядра и оптимизация
- v7.2 28/Apr/2015 Внешний интерфейс
- **v7.3 16/Jun/2015 Экосистема пакетов, Composer и работа с расширениями**
- v7.4 04/Aug/2015 Переработка внутреннего интерфейса часть 2
- v7.5 29/Sep/2015 (будет определено...)
- v7.6 xx/xxx/2015 TYPO3 CMS 7 LTS (Long Term Release)

<https://typo3.org/typo3-cms/roadmap/>

<http://typo3.org/news/article/embrace-and-innovate-typo3-cms-7/>

Установка

- Официальная процедура установки под Linux/Mac OS X (DocumentRoot, например /var/www/site/htdocs):

```
$ cd /var/www/site
$ wget --content-disposition get.typo3.org/7.3
$ tar xzf typo3_src-7.3.0.tar.gz
$ cd htdocs
$ ln -s ../typo3_src-7.3.0 typo3_src
$ ln -s typo3_src/index.php
$ ln -s typo3_src/typo3
$ touch FIRST_INSTALL
```

- Symbolic links под Microsoft Windows:
 - Используйте junction под Windows XP/2000
 - Используйте mlink под Windows Vista и Windows 7

Обновление до TYPO3 CMS 7.x

- Обновление возможно лишь с TYPO3 CMS 6.2 LTS
- Сначала TYPO3 CMS < 6.2 необходимо обновить до TYPO3 CMS 6.2 LTS

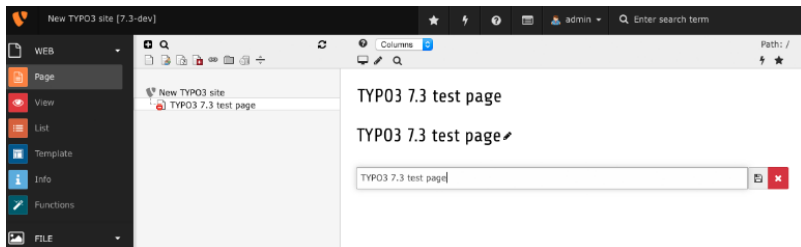
- Инструкции по обновлению:
http://wiki.typo3.org/Upgrade#Upgrading_to_7.3
- Официальное руководство TYPO3 "TYPO3 Installation and Upgrading": <http://docs.typo3.org/typo3cms/InstallationGuide>
- Общий подход:
 - Проверка минимальных системных требований (PHP, MySQL и т. д.)
 - Просмотр deprecation_*.log из старой установки TYPO3
 - Обновление всех расширений до последних версий
 - Загрузка новых исходных файлов и запуск Install Tool → Upgrade Wizard
 - Обзор модулей, запускаемых для внутренних пользователей (не обязательно)

Глава 1:

Backend / Внутренний интерфейс

Название страницы в модулях Страница и Список

Изменить названия страниц в модулях "Страница" и "Список" теперь можно дважды щёлкнув по заголовку страницы или по значку редактирования.



Install Tool: удаление обработанных файлов

В разделе "Clean up" из Install Tool добавлен новый функционал для удаления временных файлов (например, эскизов изображений) из FAL. Это полезно при изменении настроек для работы с графикой, либо после обновления GraphicsMagick/ImageMagick для принудительного пересоздания всех изображений.

Clear processed files

The File Abstraction Layer stores a database record for every file it needs to process. (e.g. image thumbnails) In case you modified some graphics settings (All Configuration [GFX]) and you need all processed files to get regenerated, you can use this tool to remove the existing ones. The new processed files are created once they are needed.

Clear processed files

Новое поле в мета данных FAL Meta Data

К мета данным записей FAL было добавлено поле "Copyright" (system extension: filemetadata).

Edit File Metadata "test.txt" on root level

General	Access	Metadata	Categories
Creator <input type="text"/>			
Creator Tool <input type="text"/>		Publisher <input type="text"/>	
Source <input type="text"/>		Copyright <input type="text"/>	
Geo Location			
Country <input type="text"/>	Region <input type="text"/>	City <input type="text"/>	

File Metadata [6]

Глава 2:

TScnfig и TypoScript

Новая функция stdWrap strtotime

- Новое свойство TypoScript stdWrap strtotime позволяет преобразовывать форматированные даты в Unix timestamps, например, для вычисления дат
- Воспринимаются значения 1 или любая строка времени, используемая в виде первого аргумента функции PHP strtotime()

```
date_as_timestamp = TEXT
date_as_timestamp {
    value = 2015-04-15
    strtotime = 1
}
```

```
next_weekday = TEXT
next_weekday {
    data = GP:selected_date
    strtotime = + 2 weekdays
    strftime = %Y-%m-%d
}
```

GPmerged в Условиях

- Если в условиях TypoScript использовать переменные GP, то возвращены будут лишь POST переменная, в случае, если запрос содержит и POST, и GET переменные.
- Новый параметр GPmerged объединяет оба метода и возвращает результат

```
[globalVar = GPmerged:tx_demo|foo = 1]
  page.90 = TEXT
  page.90.value = DEMO
[global]
```


Новые параметры для stdWrap.case

- Параметры uppercamelcase и lowercamelcase добавлены к stdWrap.case
- Пример:

```
tt_content = CASE
tt_content {
    key.field = CType
    my_custom_ctype =< lib.userContent
    my_custom_ctype {
        file = EXT:site_base/Resources/Private/Templates/SomeOtherTemplate.html
        settings.extraParam = 1
    }
    default =< lib.userContent
    default {
        file = TEXT
        file.field = CType
        file.stdWrap.case = uppercamelcase
        file.wrap = EXT:site_base/Resources/Private/Templates/|.html
    }
}
```

Для файлов JavaScript добавлено свойство integrity (1)

- Свойство integrity добавлено при включении файлов JavaScript для указания SRI и возможности верификации ресурсов (SRI: Sub-Resource Integrity, смотрите следующий слайд)
- Это применимо к свойствам TypeScript PAGE page.includeJSLibs, page.includeJSFooterlibs, includeJS и includeJSFooter
- Пример:

```
page {  
  includeJS {  
    jQuery = https://code.jquery.com/jquery-1.11.3.min.js  
    jQuery.external = 1  
    jQuery.disableCompression = 1  
    jQuery.excludeFromConcatenation = 1  
    jQuery.integrity = sha256-7LkWEzqTdpEfELxcZZlS6wAx5Ffl3zZ83lYO2/ujj7g=  
  }  
}
```

Для файлов JavaScript добавлено свойство integrity (2)

- SRI — это спецификация W3C, позволяющая разработчикам убедиться, что расположенные на сторонних серверах ресурсы не были подделаны, для чего используется
- Создание хешей целостности:
 - Параметр 1: <https://srihash.org>
 - Параметр 2: использование следующей команды shell

```
cat FILENAME.js | openssl dgst -sha256 -binary | openssl enc -base64 -A
```

- Подробнее:
 - <http://www.w3.org/TR/SRI/>

Глава 3:

Глубинные изменения

Теперь `CommandController` позволяет использовать `Symfony/Console` изнутри и предоставляет разные методы:

- `TableHelper`
 - `outputTable($rows, $headers = NULL)`
- `DialogHelper`
 - `select($question, $choices, $default = NULL, $multiSelect = false, $attempts = FALSE)`
 - `ask($question, $default = NULL, array $autocomplete = array())`
 - `askConfirmation($question, $default = TRUE)`
 - `askHiddenResponse($question, $fallback = TRUE)`
 - `askAndValidate($question, $validator, $attempts = FALSE, $default = NULL, array $autocomplete = NULL)`
 - `askHiddenResponseAndValidate($question, $validator, $attempts = FALSE, $fallback = TRUE)`

Интеграция Symfony/Console в CommandController (2)

- ProgressHelper

- `progressStart($max = NULL)`
- `progressSet($current)`
- `progressAdvance($step = 1)`
- `progressFinish()`

(примеры кода на следующих слайдах)

Интеграция Symfony/Console в CommandController (3)

```
<?php
namespace Acme\Demo\Command;
use TYPO3\CMS\Extbase\Mvc\Controller\CommandController;

class MyCommandController extends CommandController {
    public function myCommand() {

        // render a table
        $this->output->outputTable(array(
            array('Bob', 34, 'm'),
            array('Sally', 21, 'f'),
            array('Blake', 56, 'm')
        ),
        array('Name', 'Age', 'Gender'));

        // select
        $colors = array('red', 'blue', 'yellow');
        $selectedColorIndex = $this->output->select('Please select one color', $colors, 'red');
        $this->outputLine('You choose the color %s.', array($colors[$selectedColorIndex]));

        [...]
    }
}
```

Интеграция Symfony/Console в CommandController (4)

```
[...]
// ask
$name = $this->output->ask('What is your name?' . PHP_EOL, 'Bob', array('Bob', 'Sally',
    'Blake'));
$this->outputLine('Hello %s.', array($name));

// prompt
$likesDogs = $this->output->askConfirmation('Do you like dogs?');
if ($likesDogs) {
    $this->outputLine('You do like dogs!');
}

// progress
$this->output->progressStart(600);
for ($i = 0; $i < 300; $i++) {
    $this->output->progressAdvance();
    usleep(5000);
}
$this->output->progressFinish();
}
}
?>
```


Backend Login API (1)

- Полностью переработана авторизация во внутреннем интерфейсе и представлен новый API
- Извлечена форма OpenID, которая теперь использует новый API (давая независимость от главных классов ядра)
- Новая концепция авторизации базируется на "провайдерах авторизации", которые можно зарегистрировать в файле `ext_localconf.php` следующим образом:

```
$GLOBALS['TYPO3_CONF_VARS']['EXTCONF']['backend']['loginProviders'][1433416020] = [
    'provider' => \TYPO3\CMS\Backend>LoginProvider\UsernamePasswordLoginProvider::class,
    'sorting' => 50,
    'icon-class' => 'fa-key',
    'label' => 'LLL:EXT:backend/Resources/Private/Language/locallang.xlf:login.link'
];
```

- Параметры задаются следующим образом:
 - provider:
название класса провайдера, который должен реализовывать `TYPO3\CMS\Backend>LoginProvider>LoginProviderInterface`
 - sorting:
порядок ссылок на возможные провайдеры авторизации на форме авторизации
 - icon-class:
название значка-надписи для ссылки на форме авторизации
 - label:
метка для ссылки на провайдера авторизации для формы авторизации

- The `LoginProviderInterface` содержит лишь метод `public function render(StandaloneView $view, PageRenderer $pageRenderer, LoginController $loginController);`
- Параметры определяются таким образом:
 - `$view`:
Fluid `StandaloneView` выводящий форму авторизации. Необходимо указать файл шаблона, а также, по необходимости, добавить переменные для собственных нужд.
 - `$pageRenderer`:
Экземпляр `PageRenderer` имеет возможность добавления необходимых ресурсов JavaScript.
 - `$loginController`:
Экземпляр `LoginController`.

- В шаблоне должен быть раздел `<f:layout name="Login">` и `<f:section name="loginFormFields">` (для полей формы):

```
<f:layout name="Login" />
<f:section name="loginFormFields">
  <div class="form-group t3js-login-openid-section" id="t3-login-openid_url-section">
    <div class="input-group">
      <input type="text" id="openid_url"
        name="openid_url"
        value="{presetOpenId}"
        autofocus="autofocus"
        placeholder="{f:translate(key: 'openid', extensionName: 'openid')}}"
        class="form-control input-login t3js-clearable t3js-login-openid-field" />
      <div class="input-group-addon">
        <span class="fa fa-openid"></span>
      </div>
    </div>
  </div>
</f:section>
```

CategoryRegistry с новыми параметрами

- Метод CategoryRegistry->addTcaColumn получает параметры для задания l10n_mode и l10n_display:

```
\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::makeCategorizable(  
    $extensionKey,  
    $tableName,  
    'categories',  
    array(  
        'l10n_mode' => 'string (keyword)',  
        'l10n_display' => 'list of keywords'  
    )  
);
```

Спрайты в модулях внутреннего интерфейса

- Модули внутреннего интерфейса (основные, вроде "Веб" и подмодули, вроде "Список файлов") теперь могут использовать спрайты для значков (можно использовать лишь спрайты, известные для TYPO3!)
- Пример:

```
\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::addModule(  
    'web',  
    'layout',  
    'top',  
    \TYPO3\CMS\Core\Utility\ExtensionManagementUtility::extPath($_EXTKEY) . 'Modules/  
        Layout/',  
    array(  
        'script' => '_DISPATCH',  
        'access' => 'user,group',  
        'name' => 'web_layout',  
        'configuration' => array('icon' => 'module-web'),  
        'labels' => array(  
            'll_ref' => 'LLL:EXT:cms/layout/locallang_mod.xlf',  
        ),  
    ),  
);
```

FormEngine NodeFactory API (1)

- Теперь возможно зарегистрировать новые узлы и переназначать существующие

```
$GLOBALS['TYPO3_CONF_VARS']['SYS']['formEngine']['nodeRegistry'][1433196792] = array(  
    'nodeName' => 'input',  
    'priority' => 40,  
    'class' => \MyVendor\MyExtension\Form\Element\T3editorElement::class  
);
```

- Приведённый пример регистрирует новый класс `MyVendor\MyExtension\Form\Element\T3editorElement` в качестве класса, формирующего тип `TCA input`, который должен реализовывать интерфейс `TYPO3\CMS\Backend\Form\NodeInterface`
- Ключом массива служит Unix timestamp даты добавления регистрируемого элемента

FormEngine NodeFactory API (2)

- Если для обработки того же типа было зарегистрировано несколько элементов, используется имеющий высший приоритет (от 0 до 100)
- Новый тип TCA можно регистрировать следующим образом:
TCA

```
'columns' => array(  
    'bodytext' => array(  
        'config' => array(  
            'type' => 'text',  
            'renderType' => '3dCloud',  
        ),  
    ),  
)
```

ext_localconf.php

```
$GLOBALS['TYPO3_CONF_VARS']['SYS']['formEngine']['nodeRegistry'][1433197759] = array(  
    'nodeName' => '3dCloud',  
    'priority' => 40,  
    'class' => \MyVendor\MyExtension\Form\Element\ShowTextAs3dCloudElement::class  
);
```


Сигнал postProcessMirrorUrl

- Сигнал postProcessMirrorUrl перемещён в новый класс

BREAKING CHANGE!

- В следующем примере кода принимается во внимание версия TYPO3:

```
$signalSlotDispatcher->connect(  
    version_compare(TYPO3_version, '7.0', '<')  
    ? 'TYPO3\CMS\Lang\Service\UpdateTranslationService'  
    : 'TYPO3\CMS\Lang\Service\TranslationService',  
    'postProcessMirrorUrl',  
    'Vendor\Extension\Slots\CustomMirror',  
    'postProcessMirrorUrl'  
);
```

Глава 4:

Extbase и Fluid

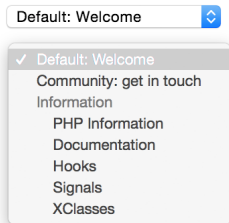
ActionMenuItemGroupViewHelper (1)

- Этот проектор/`ViewHelper` позволяет использовать группы параметров в полях выбора внутреннего интерфейса, контролирующие выбираемые действия
- Пример:

```
<f:be.menus.actionMenu>
  <f:be.menus.actionMenuItem label="Default: Welcome" controller="Default" action="
    index" />
  <f:be.menus.actionMenuItem label="Community: get in touch" controller="Community"
    action="index" />
  <f:be.menus.actionMenuItemGroup label="Information">
    <f:be.menus.actionMenuItem label="PHP Information" controller="Information"
      action="listPhpInfo" />
    <f:be.menus.actionMenuItem label="Documentation" controller="Information"
      action="documentation" />
    <f:be.menus.actionMenuItem label="Hooks" controller="Information" action="hooks"
      />
    <f:be.menus.actionMenuItem label="Signals" controller="Information" action="
      signals" />
    <f:be.menus.actionMenuItem label="XClasses" controller="Information" action="
      xclass" />
  </f:be.menus.actionMenuItemGroup>
</f:be.menus.actionMenu>
```

ActionMenuItemGroupViewHelper (2)

- Пример предыдущего слайда выводится следующим образом:



Поддержка шаблонов для FlashMessagesViewHelper

- Теперь FlashMessagesViewHelper поддерживает шаблоны/templates
- Новый атрибут as позволяет указать название переменной, которую можно использовать внутри дочерних элементов проектора/ViewHelper для доступа к всплывающим сообщениям
- Пример:

```
<f:flashMessages as="flashMessages">
  <ul class="myFlashMessages">
    <f:for each="{flashMessages}" as="flashMessage">
      <li class="alert {flashMessage.class}">
        <h4>{flashMessage.title}</h4>
        <span class="fancy-icon">{flashMessage.message}</span>
      </li>
    </f:for>
  </ul>
</f:flashMessages>
```

- Замечание: теперь параметр renderMode устарел

Новые свойства cObject FLUIDTEMPLATE (1)

- cObject FLUIDTEMPLATE был дополнен `templateRootPaths` и `templateName`
- Что позволяет указать название шаблона и для вывода будет использован шаблон с этим названием наряду с указанным форматом по пути для шаблонов из настройки `templateRootPaths`
- `templateRootPaths` использует ту же логику резервирования, что и `layoutRootPath` и `partialRootPath`
 - `templateName`: `string/stdWrap`
 - `templateRootPaths`: массив путей к файлам с поддержкой префикса "EXT:"

Новые свойства cObject FLUIDTEMPLATE (2)

■ Пример TypoScript:

```
lib.stdContent = FLUIDTEMPLATE
lib.stdContent {
    templateName = TEXT
    templateName.stdWrap {
        cObject = TEXT
        cObject {
            data = levelfield:-2,backend_layout_next_level,slide
            override.field = backend_layout
            split {
                token = frontend__
                1.current = 1
                1.wrap = |
            }
        }
    }
    ifEmpty = Default
}
templateRootPaths {
    10 = EXT:frontend/Resources/Private/Templates
    20 = EXT:sitemodification/Resources/Private/Templates
}
}
```

Удаление xmlns-атрибутов и тегов HTML (1)

- С введением в обиход атрибутов `xmlns:*` для подключения проекторов/`ViewHelpers`, стала возможна поддержка со стороны IDE для шаблонов Fluid. Проблема была в том, что атрибуты `xmlns:*` с соответствующим тегом также выводились, что обычно нежелательно.
- Обойти проблему удавалось с использованием `sections`, но такое решение неинтуитивно понятно, и недопустимо в макетах/`layouts`. Это также негативно отражается на задействованных ресурсах.
- `xmlns:*` атрибуты для верных областей именования `ViewHelper` теперь удаляются перед выводом, если используется следующий синтаксис: `http://typo3.org/ns/<phpNamespace>` (`xmlns` атрибуты для не-`ViewHelper` областей — сохраняются)

Удаление xmlns-атрибутов и тегов HTML (2)

- Включение областей именованя ViewHelper внутри тега HTML с атрибутом data-namespace-typo3-fluid="true" для предотвращения вывода всего тега HTML

```
<html data-namespace-typo3-fluid="true"
  xmlns:f="http://typo3.org/ns/TYPO3/CMS/Fluid/ViewHelpers"
  xmlns:n="http://typo3.org/ns/GeorgRinger/News/ViewHelpers">

  <f:if condition="{newsItem.title}">
    <f:then>
      <n:titleTag>{newsItem.title}</n:titleTag>
    </f:then>
    <f:else>
      <n:titleTag>News-Detail</n:titleTag>
    </f:else>
  </f:if>

</html>
```

Новые методы во Fluid-StandaloneView

- StandaloneView дополнен setTemplateRootPaths(\$templatePaths) и setTemplate(\$templateName, \$throwException = TRUE)
- с тем же функционалом, что и с Object FLUIDTEMPLATE
- Пример (вывод шаблона email):

```
$view = GeneralUtility::makeInstance(StandaloneView::class);  
$view->setLayoutRootPaths(array(GeneralUtility::getFileAbsFileName(  
    'EXT:my_extension/Resources/Private/Layouts')));  
$view->setPartialRootPaths(array(GeneralUtility::getFileAbsFileName(  
    'EXT:my_extension/Resources/Private/Partials')));  
$view->setTemplateRootPaths(array(GeneralUtility::getFileAbsFileName(  
    'EXT:my_extension/Resources/Private/Templates')));  
$view->setTemplate('Email/Notification');  
$emailBody = $view->render();
```

Обработка данных для FLUIDTEMPLATE сObject (1)

- сObject FLUIDTEMPLATE дополнен dataProcessing
- Эта настройка служит для добавления одного или нескольких обработчиков для работы с переменной \$data выводимого в текущий момент сObject (например, tt_content или page)
- Процессор должен реализовывать интерфейс FluidTemplateDataProcessorInterface и содержать следующий метод:

```
function process(array &$data, array $processorConfiguration,  
    array $configuration, StandaloneView $view) {  
    [...]  
}
```

■ Пример:

```
my_custom_ctype = FLUIDTEMPLATE
my_custom_ctype {
    templateRootPaths {
        10 = EXT:your_extension_key/Resources/Private/Templates
    }
    templateName = CustomName
    settings {
        extraParam = 1
    }
    dataProcessing {
        1 = Vendor\YourExtensionKey\DataProcessing\MyFirstCustomProcessor
        2 = AnotherVendor\AnotherExtensionKey\DataProcessing\MySecondCustomProcessor
        2 {
            options {
                myOption = SomeValue
            }
        }
    }
}
```

Глава 5:

Устаревшие/удаленные функции

TCA:

- Параметры `_PADDING`, `_VALIGN` и `DISTANCE` удалены из `TCA['aTable']['columns']['aField']['config']['wizards']`
- Ключ `TCA['aTable']['ctrl']['mainPalette']` удалён

TScfig:

- Ключи `mod.web_layout.tt_content.fieldOrder` и `TCEFORM.aTable.aField.linkTitleToSelf` удалены

Уловки/hooks:

- Уловки/hooks используют ключ `type` вместо `form_type`
- Уловка/hook `getSingleFieldClass` удалена

Удаление IdentityMap из Extbase Persistence

- Класс IdentityMap удалён из Extbase persistence (генерируется ReflectionException при его использовании)
- Доступ к ранее существующему свойству IdentityMap внутри DataMapper и Repository теперь не удастся, а создание экземпляра IdentityMap более невозможно
- Вместо этого используйте экземпляр "Sessions":

```
$session = GeneralUtility::makeInstance(ObjectManager::class)->get(
    \TYPO3\CMS\Extbase\Persistence\Generic\Session::class
);

$session->registerObject($object, $identifier);

if($session->hasIdentifier($identifier)) {
    $object = $session->getObjectByIdentifier($identifier, $className);
}
```

- Использование файла `typo3conf/ext/Tables.php` не рекомендовано. Вместо этого используйте файл:
`<your_extension>/Configuration/TC/Overrides/pages.php`
- Настройка `$TYPO3_CONF_VARS[GFX][png_to_gif]` удалена
- В установках TYPO3 CMS, не имеющих установленное расширение `rsaauth`, пароли внутреннего интерфейса теперь преобразуются в обычный текст (решение: установить расширение `rsaauth` или использовать HTTPS для внутреннего интерфейса)
- Метод `exec_SELECTgetRows()` теперь проверяет параметр `$uidIndexField`. Если указанное поле не присутствует в результатах из базы данных, вызывается `InvalidArgumentException`.

Разное (2)

- DBAL параметр `config.classFile` удалён
- Параметры `iconOnly` и `styleAttributes` из `CshViewHelper` теперь помечены как не рекомендуемые
- Параметр `TYPOScript page.bgImg` теперь не рекомендован
- Метод `isEnabled()` класса `T3editor` теперь не рекомендован
- Старый TYPO3 `ClassLoader` удалён для пользы или используется `Composer ClassLoader`

Глава 6:

Источники и авторы

Источники

Новости TYPO3:

- <http://typo3.org/news>

Информация по выходу версий:

- http://wiki.typo3.org/TYPO3_CMS_7.3.0
- [INSTALL.md](#) и [ChangeLog](#)
- [typo3/sysexst/core/Documentation/Changelog/7.3/](http://typo3.org/sysexst/core/Documentation/Changelog/7.3/)*

TYPO3 Bug-/Issuetracker:

- <https://forge.typo3.org/projects/typo3cms-core>

Репозитории TYPO3 Git:

- <https://git.typo3.org/Packages/TYPO3.CMS.git>
- <https://git.typo3.org/Packages/TYPO3.Fluid.git>

Слайды нововведений TYPO3 CMS:

Patrick Lobacher

(Исследования, сбор информации и немецкая версия)

Michael Schams

(Лидер проекта и английская версия)

Переводчики:

Андрей Аксёнов, Sergio Catala, Jigal van Hemert, Michel Mix, Sinisa Mitrovic,
Angeliki Plati, Nena Jelena Radovic, Roberto Torresani

<http://typo3.org/download/release-notes/whats-new>

Licensed under Creative Commons BY-NC-SA 3.0

