TYPO3 CMS 8.5 - What’s New

Chapter Overview

Introduction

Backend User Interface

TSconfig & TypoScript

In-Depth Changes

Extbase & Fluid

Deprecated/Removed Functions

Sources and Authors
Introduction

The Facts
Introduction

TYPO3 CMS 8.5 - The Facts

- Release date: 20 December 2016
- Release type: Sprint Release
- Slogan: On the clock
Introduction

System Requirements

- PHP: version 7
- MySQL: version 5.5 to 5.7
- Disk space: min 200 MB
- PHP settings:
  - memory_limit >= 128M
  - max_execution_time >= 240s
  - max_input_vars >= 1500
  - compilation option --disable-ipv6 must not be used
- The backend requires Microsoft Internet Explorer 11 or later, Microsoft Edge, Google Chrome, Firefox, Safari or any other modern, compatible browser
Introduction

Development and Release Timeline

Typo3 CMS 8.5 - What's New
Introduction

TYPO3 CMS Roadmap

Release dates and their primary focus:

- v8.0  22/Mar/2016  Adding last minute things
- v8.1  03/May/2016  Cloud Integration
- v8.2  05/Jul/2016  Doctrine Prerequisites
- v8.3  30/Aug/2016  Rich Text Editor
- v8.4  18/Oct/2016  Doctrine Migration + Upgrades
- v8.5  20/Dec/2016  New RTE + Integrator Support
- v8.6  14/Feb/2017  to be determined
- v8.7  04/Apr/2017  LTS Preparation

https://typo3.org/typo3-cms/roadmap/
https://typo3.org/news/article/kicking-off-typo3-v8-development/
Introduction

Installation

- **Official classic installation procedure under Linux/Mac OS X** (DocumentRoot for example `/var/www/site/htdocs`):

  ```
  $ cd /var/www/site
  $ wget --content-disposition get.typo3.org/8.5
  $ tar xzf typo3_src-8.5.1.tar.gz
  $ cd htdocs
  $ ln -s ../typo3_src-8.5.1 typo3_src
  $ ln -s typo3_src/index.php
  $ ln -s typo3_src/typo3
  $ touch FIRST_INSTALL
  ```

- **Symbolic links under Microsoft Windows:**
  - Use junction under Windows XP/2000
  - Use mklink under Windows Vista and Windows 7
Introduction

Upgrade to TYPO3 CMS 8.x

- Upgrades only possible from TYPO3 CMS 7.6 LTS or 8.x
- TYPO3 CMS < 7.6 LTS should be updated to TYPO3 CMS 7.6 LTS first

- Upgrade instructions:
  
  http://wiki.typo3.org/Upgrade#Upgrading_to_8.5

- Official TYPO3 guide "TYPO3 Installation and Upgrading":
  
  http://docs.typo3.org/typo3cms/InstallationGuide

- General approach:
  
  - Check minimum system requirements (PHP, MySQL, etc.)
  - Review `deprecation_*.log` in old TYPO3 instance
  - Update all extensions to the latest version
  - Deploy new sources and run Install Tool -> Upgrade Wizard
  - Review startup module for backend users (optionally)
Introduction

PHP Version 7

- PHP 7.0 is the minimum requirement for TYPO3 CMS 8.x
- TYPO3 will support subsequent PHP 7 releases as they come out
- This version raise gives a significant performance boost to the overall system
- Not only backend editors will notice a more fluent interface, but the new all-time record for a full cached page call in the frontend is below 7 milliseconds now, which is approximately 40% faster compared to running the very same website with PHP version 5.5
- We also started using new features from this PHP version, for instance the cryptographically secure pseudo-random generators are in active use already
Chapter 1:

Backend User Interface
A flexible new framework for building forms has been integrated in TYPO3 CMS 8.5.

It replaces the legacy Form Wizard based on ExtJS and the depending frontend rendering system.

The new Form Editor uses jQuery and uses a modern architecture, ensuring high flexibility and extensibility.

Highly customizable and configuration settings are stored in YAML files.

The feature list is impressive
(stay tuned for the complete documentation)

Demonstration video of a preview is available at YouTube: https://www.youtube.com/watch?v=F9sTAOEcTIO
Backend User Interface

New Form Framework (2)
Backend User Interface

New Form Framework (3)
The next generation of rich text editing has been implemented in the TYPO3 backend: **CKEditor**.

The current state is explicitly marked as *experimental* and the extension is not installed by default.

Further details about this open source editor: [http://ckeditor.com](http://ckeditor.com)
Backend User Interface

Position and Order of Elements

- The order and position of certain fields in the backend of TYPO3 has been streamlined
- The aim is to meet users’ expectation where to find commonly used options in the user interface
- This is especially important for recurring field definitions and generic categories shared by a lot of records
- Extension authors are encouraged to follow the specified positions and orders of elements in the official documentation

- Backend consistency is king! :-)
Chapter 2:

TSconfig & TypoScript
In previous versions of TYPO3 CMS, it was possible to override the "New Page Creation Wizard" via custom scripts:

```php
mod.web_list.newPageWiz.overrideWithExtension = myextension
```

The new way of handling entry-points and custom scripts is now built via modules/routes and the option listed above has been removed.

The following new TSconfig option can be used instead:

```php
mod.newPageWizard.override = my_custom_module
```

Instead of setting the option to a certain extension key, a custom module or route needs to be specified.
Number of Search Results

- Maximum number of search results can be configured in TypoScript now: `plugin.tx_indexedsearch.settings.blind.numberOfResults`
- This setting stores a list of values
- If number of search results is passed in the request and matches one of the values configured, this number is used
- If number of search results is not passed in the request or does not match any of the values configured, the first value of the list is used
- To keep backward compatibility, the default values are: 10, 25, 50 and 100
Menu processor utilizes HMENU to generate a JSON-encoded menu string that is be decoded again and assigned to FLUIDTEMPLATE.

Additional DataProcessing is supported and applied to each record.

Supported options: as, levels, expandAll, includeSpacer, titleField (see TyposcriptReference for more options).
Example TypoScript configuration:

```typescript
10 = TYPO3\CMS\Frontend\DataProcessing\MenuProcessor
10 {
    special = list
    special.value.field = pages
    levels = 7
    as = menu
    expandAll = 1
    includeSpacer = 1
    titleField = nav_title // title
    dataProcessing {
        10 = TYPO3\CMS\Frontend\DataProcessing\FilesProcessor
        10 {
            references.fieldName = media
        }
    }
}
```
**TSconfig & TypoScript**

**TypoScript Function _encapsLines**

- TypoScript function _encapsLines rendered two paragraphs for one empty trailing line-break in the content. This has been fixed now.
- The change possibly affects the appearance in the frontend, if multiple empty trailing paragraphs exist in the RTE content. The last paragraph is no longer rendered twice in the frontend since TYPO3 CMS version 8.5.
Chapter 3: In-Depth Changes
In-Depth Changes

FormEngine Data Providers

- FormEngine data provider TcaFlexFetch has been merged into TcaFlexPrepare
- This only affects instances in the unlikely case that a custom data provider declared a dependency to TcaFlexFetch
In-Depth Changes

TCA in `ext_tables.php`

- Frontend requests no longer load `ext_tables.php` in requests
- This change impacts extensions which configure the TCA in `ext_tables.php` (which is not allowed anyway)
- Install Tool provides a test "TCA ext_tables check" to identify such extensions

TCA in `ext_tables.php` check
Check if an extension changes `$_GLOBALS['TCA']` in `ext_tables.php`.

No TCA changes in `ext_tables.php` files. Good job!

Check loaded extensions
In-Depth Changes

TCA in `ext_tables.php`

- Database tables which hold localized and translated records were configurable in the TCA
  - `$TCA[<table_name>]['ctrl']['transForeignTable']`
    (usually pointed to table: `pages_language_overlay`)
  - `$TCA[<table_name>]['ctrl']['transOrigPointerTable']`
    (usually pointed to table: `pages`)

- This configuration has been replaced with hard-coded table names in order to prevent special handling and prepare for a combination of both tables in the future
In-Depth Changes

Tables removed from defaultCategorizedTables

- The following tables have been removed from defaultCategorizedTables:
  - pages
  - tt_content
  - sys_file_metadata

- For these tables the core API ExtensionManagementUtility::makeCategorizable() is executed to define a common position of the categories field
In-Depth Changes

Low-level Parameter Changes (1)

- Low-level commands listed below use the Symfony Console now.
- New commands behave like the old ones, but allow using certain parameters.
  - DeletedRecordsCommand
  - CleanFlexFormsRecordsCommand
  - OrphanRecordsCommand
  - LostFilesCommand
  - MissingFilesCommand
  - MissingRelationsCommand
  - DoubleFilesCommand
  - RteImagesCommand
In-Depth Changes

Low-level Parameter Changes (2)

- Related PHP classes have been removed
  (e.g. TYPO3\CMS\Lowlevel\DeletedRecordsCommand)
- Executing the command via cli_dispatch does not work anymore
  (e.g. typo3/cli_dispatch lowlevel cleaner deleted)
- Calling the PHP class results in a fatal PHP error now
- Commands can now be executed via CLI as follows:
  /typo3/sysext/core/bin/typo3 cleanup:<command>
  for example:
  /typo3/sysext/core/bin/typo3 cleanup:deletedrecords
In-Depth Changes

Re-factor FlexForm Data Structure Handling

- With the deprecation of BackendUtility::getFlexFormDS() the hook getFlexFormDSClass is no longer called
In-Depth Changes

Admin Panel

- Admin Panel features a new setting to debug Fluid output: Preview -> Show fluid debug output
- If enabled, the following details are shown in the frontend:
  - path to the template file of a partial
  - name of a section
- This feature enables integrators to easily locate the correct template and section
In-Depth Changes

System Status Updates (Reports)

- Results of test in the "System Status Updates (reports)" can be sent via email
- A checkbox has been added to the job configuration to:
  - send an email if the system has warnings or errors
  - always generate an email
- The default is to include warnings and errors only
In-Depth Changes

Language Packs

- Deactivating languages in the module "Languages" left language data remaining in directory typo3conf/l10n/<locale>/
- A "remove"-button has been added, which disables the language and purges the data in the directory
In-Depth Changes

Hook in DataHandler \texttt{localize()} 

- A new hook has been added to the \texttt{localize()} function
- This allows for example to use external translation services or custom transliteration functions that handle various content transformations

- Hook:
  \[ \text{Hook:} \]
  \[ \$\text{GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']} \]
  \[ \text{['t3lib/class.t3lib_tcemain.php']} \]
  \[ \text{['processTranslateToClass']} \]

- Example usage:
  
  \begin{verbatim}
  class YourHookClass
  {
      public function processTranslateTo_copyAction(&$content, $lang, $dataHandler)
      {
          // Do something with content (translate, transliterate etc.)
      }
  }
  \end{verbatim}
In-Depth Changes

Update Wizard

The Update Wizard in the Install Tool lists all tasks marked as completed.

Checkboxes and a button "Recheck chosen wizards" allow to re-initiate the updates. The wizard will test if the task needs to be executed again.
In-Depth Changes

Suggest Wizard

- The FormEngine ("TCEforms") allows to configure the order of results by the suggest wizard
- The new option is a standard SQL order-by definition:
  'orderBy' => 'field ASC/DESC'
- Example TCA configuration:

```php
'config' => [
  ...
  'wizards' => [
    'suggest' => [
      'type' => 'suggest',
      'default' => [
        'searchWholePhrase' => true,
        'addWhere' => ' AND tx_news_domain_model_news.uid != ###THIS_UID###',
        'orderBy' => 'datetime DESC',
      ],
      'orderBy' => 'datetime DESC',
    ],
  ],
]
```
In-Depth Changes

Miscellaneous

- All system information added by `addSystemInformation()` have `InformationStatus::STATUS_NOTICE` as the default value now.
- Enumeration constants can be retrieved easily now:
  - `EnumerationClass::getName($value);`
  - `EnumerationClass::getHumanReadableName($value);`
- Priorities of core TypeConverters have changed from 1, 2, 3,... to 10, 20, 30,... When registering custom TypeConverter(s), make sure they are using the correct priorities.
- **ISO-8601** is used to pass date and datetime values between server and client now. Check if your custom FormEngine render types need to be updated (`eval=date/datetime`).

TYPO3 CMS 8.5 - What's New
Chapter 4: Extbase & Fluid
Direct SQL query functionality also supports QueryBuilder objects and instances of `\Doctrine\DBAL\Statement` as prepared statements.

The following example works in any Extbase repository using native Doctrine DBAL statements:

```php
$connection = $this->objectManager->get(ConnectionPool::class)->getConnectionForTable('mytable');
$statement = $this->objectManager->get(
    \Doctrine\DBAL\Statement::class,
    'SELECT * FROM mytable WHERE uid=? OR title=?',
    $connection
);

$query = $this->createQuery();
$query->statement($statement, [$uid, $title]);
```
cHash Argument

- URIs to Extbase actions now require a valid cHash by default (cached and uncached actions)
- The behavior can be disabled for all actions using the feature switch requireCHashArgumentForActionArguments
Content for ViewHelper \texttt{f:form.select}

- Introduced two new ViewHelpers allowing the manual definition of all options and optgroups for the \texttt{f:form.select} as tag content of the select field
  - OptionViewHelper
  - OptgroupViewHelper

- Example:

```xml
<f:form.select name="myproperty">
  <f:form.select.option value="1">Option one</f:form.select.option>
  <f:form.select.option value="2">Option two</f:form.select.option>
  <f:form.select.optgroup>
    <f:form.select.option value="3">Grouped option one</f:form.select.option>
    <f:form.select.option value="4">Grouped option twi</f:form.select.option>
  </f:form.select.optgroup>
</f:form.select>
```
Global Fluid ViewHelper Namespaces

- Global Fluid ViewHelper namespaces are configurable now:
  \$GLOBALS['TYPO3_CONF_VARS']['SYS']['fluid']['namespaces']

- This allows the namespaces to be manipulated as part of the site configuration

- Benefits:
  - Third party ViewHelper packages can manipulate the global Fluid namespace:
  - Third party ViewHelper packages are able to register new global namespaces as required
  - Template developers can use such global namespaces without importing them first and can use them in all Fluid templates regardless of context
The FLUIDTEMPLATE content element can mimic an actual Extbase web request now.

This makes it possible to access submitted data for example:

```php
$view->getRenderingContext()->getControllerContext()->getRequest()->getArguments();
```
Chapter 5: Deprecated/Removed Functions
Deprecation/Removed Functions

ExtJS Removal (1)

- As part of the ExtJS removal work package, the following JavaScript methods have been removed from the Backend main frame (defined in file backend.js):
  - TYPO3._instances
  - TYPO3.addInstance
  - TYPO3.getInstance
  - TYPO3.helpers.split
Deprecation/Removed Functions

ExtJS Removal (2)

- New class `TYPO3\CMS\Workspaces\Controller\AjaxDispatcher` replaces the ExtDirect router functionality in `EXT:workspaces`.

- The following classes have been moved:
  - `Classes/ExtDirect/AbstractHandler.php` now as `Classes/Controller/Remote/AbstractHandler.php`
  - `Classes/ExtDirect/ActionHandler.php` now as `Classes/Controller/Remote/ActionHandler.php`
  - `Classes/ExtDirect/MassActionHandler.php` now as `Classes/Controller/Remote/MassActionHandler.php`
Deprecated/Removed Functions

**Classes** *DatabaseConnection* and *PreparedStatement*

- The following classes have been marked as *deprecated*:
  - `TYPO3\CMS\Core\Database\DatabaseConnection`
  - `TYPO3\CMS\Core\Database\PreparedStatement`

- Use Doctrine DBAL in TYPO3 CMS 8 instead *(ConnectionPool and QueryBuilder classes)*

- These two classes will be removed in TYPO3 CMS 9
Deprecated/Removed Functions

JavaScript settings under TYPO3.configuration

- The following JavaScript settings have been removed:
  - TYPO3.configuration.debugInWindow
  - TYPO3.configuration.moduleMenuWidth
  - TYPO3.configuration.topBarHeight

- These options were not used by the TYPO3 core anyway
Deprecated/Removed Functions

Public Properties of FlexFormTools

- Two public properties have been dropped from class TYPO3\CMS\Core\Configuration\FlexForm\FlexFormTools:
  - public $traverseFlexFormXMLData_DS = array();
  - public $traverseFlexFormXMLData_Data = array();

- Accessing those properties will throw a warning now
Deprecated/Removed Functions

sys_action Translations

The following translations have been removed from
EXT:sys_action/Resources/Private/Language/locallang.xlf:

- action_BEu_hidden
- action_BEu_username
- action_BEu_password
- action_BEu_realName
- action_BEu_email
- action_BEu_usergroups

The following translation has been removed from
EXT:sys_action/Resources/Private/Language/locallang_tca.xlf:

- tx_sys_action
**Deprecated/Removed Functions**

**Extbase’s ArrayUtility Class**

- Class `\TYPO3\CMS\Extbase\Utility\ArrayUtility` has been marked as deprecated
- Alternatives are available for the following methods:
  - `integerExplode()`: see `GeneralUtility::intExplode()`
  - `trimExplode()`: see `GeneralUtility::trimExplode()`
- To replace the following methods, check the TYPO3 core class `\TYPO3\CMS\Core\Utility\ArrayUtility`:
  - `arrayMergeRecursiveOverrule()`
  - `getValueByPath()`
  - `setValueByPath()`
  - `unsetValueByPath()`
  - `sortArrayWithIntegerKeys()`
Deprecated/Removed Functions

Frameset and frame

- frameset and frame are not supported in HTML5 anymore
- The following TypoScript objects have been marked as deprecated:
  - frameset
  - frame

- The following TypoScript options have been marked as deprecated:
  - config.frameReloadIfNotInFrameset
  - config.doctype = xhtml_frames
  - config.xhtmlDoctype = xhtml_frames
  - frameSet (and its options)
  - FRAME (and its options)
  - FRAMESET (and its options)
Deprecated/Removed Functions

FlexForm Related Parsing

- Three FlexForm data structure related parsing methods have been marked as deprecated:
  - BackendUtility::getFlexFormDS()
  - GeneralUtility::resolveSheetDefInDS()
  - GeneralUtility::resolveAllSheetsInDS()

- Use the following methods instead:
  - FlexFormTools->getDataStructureIdentifier()
  - FlexFormTools->parseDataStructureByIdentifier()

*(read the method comments for a detailed description)*
Deprecated/Removed Functions

Miscellaneous (1)

- Argument $caseSensitive of the method Query::like has been marked as deprecated
  (change the collation of the appropriate fields to be stored in a case sensitive fashion or use a binary column type)

- Previously undefined member parentMenuArr in AbstractMenuContentObject has been added as public member and marked as deprecated
  (use the API function getParentMenuArr() instead)

- Method ContentObjectRenderer::fileResource() has been marked as deprecated
  (migrate your code to use file_get_contents())
The following JavaScript property has been marked as deprecated:
`top.TYPO3.Backend.ContentContainer.iframe`
(use accessor method `top.TYPO3.Backend.ContentContainer.get()` instead)

TypoScript setting `config.noScaleUp` has been marked as deprecated
(use the following global TYPO3 configuration instead:
$GLOBALS['TYPO3_CONF_VARS']['GFX']['processor_allowUpscaling'])

The following (unused) method has been marked as deprecated:
`AbstractFunctionModule->getBackPath()`
The following TCA option has been removed:
$TCA[$table][ctrl][versioning_followPages]

Adding items to TCA tree with pageTsConfig addItem requires an icon
identifiers from the icon registry now (paths are not supported
anymore):
TCEFORM.pages.category.addItem.12345.icon = my-registered-icon

All XLIF Language files of EXT:lang have been moved to
Resources/Private/Language/
This affects all extensions which use labels from EXT:lang!
OLD: EXT:lang/locallang_alt_doc.xlf
NEW: EXT:lang/Resources/Private/Language/locallang_alt_doc.xlf
The following TypoScript option has been marked as deprecated: `config.mainScript`

Various CharsetConverter methods have been marked as deprecated (use the `mb_string` functionality from the `symfony/polyfill-mbstring` package instead).

The following method call is not required anymore and triggers a deprecation log entry now:
`TYPO3\CMS\IndexedSearch\Indexer->includeCrawlerClass()`

The following method call with a "&" symbol triggers a deprecation log entry now: `GeneralUtility::callUserFunction()`

Method `getRecordUidsToCopy()` has been marked as deprecated.
Chapter 6:
Sources and Authors
Sources and Authors

Sources

**TYPO3 News:**
- [http://typo3.org/news](http://typo3.org/news)

**Release Infos:**
- [http://wiki.typo3.org/TYP03_CMS_8.5.0](http://wiki.typo3.org/TYP03_CMS_8.5.0)
- INSTALL.md and ChangeLog
- typo3/sysext/core/Documentation/Changelog/8.5/*

**TYPO3 Bug-/Issuetracker:**

**TYPO3 and Fluid Git Repositories:**
- [https://git.typo3.org/Packages/TYP03.CMS.git](https://git.typo3.org/Packages/TYP03.CMS.git)
- [https://github.com/TYPO3/Fluid](https://github.com/TYPO3/Fluid)
Sources and Authors

TYPO3 CMS What’s New Team:

Pierrick Caillon, Sergio Catala, Richard Haeser, Jigal van Hemert,
Patrick Lobacher, Michel Mix, Sinisa Mitrovic,
Nena Jelena Radovic, Michael Schams and Roberto Torresani

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

Licensed under Creative Commons BY-NC-SA 3.0