TYPO3 Version 10.2 – What’s New

Summary of the new features, changes and improvements

Created by:
Michael Schams

02/December/2019
Creative Commons BY-NC-SA 3.0
TYPO3 Version 10.2 – What’s New

Chapter Overview

Introduction

Backend User Interface

Changes for Integrators

Changes for Developers

System Extension "Form"

Deprecated/Removed Functions

Sources and Authors
Introduction

The Facts
Introduction

TYPO3 Version 10.2 – The Facts

- Release date: 03 December 2019
- Release type: Sprint Release
Introduction

Executive Summary

TYPO3 version 10.2 is the third sprint release on the way to the LTS-version (long-term support) in 2020. It is also the last sprint release of the year.

A lot of functionality was developed during the TYPO3 Initiative Week (T3INIT19) and TYPO3 v10.2 already contains some of these components.

This release paves the way for a cutting-edge environment. TYPO3 v10.2 not only supports Symfony version 5.0, but is also the first TYPO3 release that supports PHP version 7.4. It also marks the last release before the feature freeze release in February 2020.
Introduction

System Requirements

- PHP version 7.2, 7.3 or 7.4
- PHP settings:
  - `memory_limit` $\geq 256\text{M}$
  - `max_execution_time` $\geq 240\text{s}$
  - `max_input_vars` $\geq 1500$
  - compilation option `--disable-ipv6` must not be used
- Most database servers supported by **Doctrine DBAL** also work with TYPO3. Tested DB engines are for example:

![MySQL](image1.png)  ![MariaDB](image2.png)  ![PostgreSQL](image3.png)  ![SQL Server](image4.png)  ![SQLite](image5.png)
**Introduction**

Development, Release and Maintenance Timeline

**TYPO3 v10**

**Extended Support**

The **TYPO3 GmbH** offers further support options for TYPO3 v10 LTS even after 30 April 2023 for up to two additional years.
Introduction

TYPO3 v10 Roadmap

Release dates and their primary focus:

- v10.0 23/July/2019  Pave the way for exciting new concepts and APIs
- v10.1 01/Oct/2019  Routing Improvements and Site Handling v2
- v10.2 03/Dec/2019  Fluid/Rendering Engine Improvements
- v10.3 04/Feb/2020  Feature Freeze
- v10.4 07/Apr/2020  LTS Release (Long-term Support)

https://typo3.org/article/typo3-v10-roadmap/
https://typo3.org/article/typo3-v10-safe-and-sound/
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/10.2
  $ tar xzf typo3_src-10.2.0.tar.gz
  $ cd htdocs
  $ ln -s ../typo3_src-10.2.0 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 and higher
Introduction

Installation Using composer

- Installation using composer under Linux, Mac OS X and Windows 10:
  
  $ cd /var/www/site/
  $ composer create-project typo3/cms-base-distribution typo3v10 ^10.2

- Alternatively, create your custom composer.json file and run:
  
  $ composer install

Further details and examples for composer.json files are available at:

https://composer.typo3.org
Chapter 1: Backend User Interface
## Backend User Interface

### Extension Manager

The Extension Manager now shows links to extension documentation.

### Get Extensions

<table>
<thead>
<tr>
<th>Actions</th>
<th>Extension</th>
<th>Key</th>
<th>Version</th>
<th>Last Updated</th>
<th>Description</th>
<th>Manual</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Zabbix Client</td>
<td>zabbix_client</td>
<td>0.2.5</td>
<td>16-11-19</td>
<td>Client for zabbix monitoring system. Secure up...</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dynamic Content Elements (DCE)</td>
<td>dce</td>
<td>2.3.1</td>
<td>16-11-19</td>
<td>Best flexform based content elements since 2012...</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Minify for vhs assets</td>
<td>vhsminify</td>
<td>2.0.3</td>
<td>16-11-19</td>
<td>minify for vhs assets</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mailman Extension</td>
<td>mailmanext</td>
<td>1.2.0</td>
<td>16-11-19</td>
<td>An extension to manage mailman mailinglists</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AOE Database Sequenzer</td>
<td>aoe_dbsequenzer</td>
<td>3.0.0</td>
<td>15-11-19</td>
<td>With this extension you can ensure different...</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Translatable Labels for Editors</td>
<td>translatelabels</td>
<td>2.0.6</td>
<td>15-11-19</td>
<td>This extension extends the TYPO3 translation...</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Modules</td>
<td>modules</td>
<td>3.2.0</td>
<td>15-11-19</td>
<td>Modules - Little helper for creating ...</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[NITSAN] News Slider</td>
<td>ns_news_slider</td>
<td>2.0.1</td>
<td>15-11-19</td>
<td>Do you want cool sliders for most popular TYPO3...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Backend User Interface

Pagetree Accessibility

Backend users can now use their keyboard to navigate through the pagetree. For example the arrow keys, "home", "end", "enter", "space", etc. This is in accordance to the best practices as described in WAI-ARIA Authoring Practices 1.1 by the W3C.
Chapter 2:
Changes for Integrators
Changes for Integrators

Site Configuration (1)

- The site title can now be configured in **SITE CONFIGURATION → Sites**.
- This lets integrators specify different site titles per language.
- The field in the template record is obsolete and has been marked as **deprecated**.
- The field `sys_template.sitetitle` (database and TCA) will be removed in TYPO3 v11.
- The site title is used for the page title as well as for future `schema.org` integrations.
Changes for Integrators

Site Configuration (2)

- It is now possible to use environment variables in imports of site configuration YAML files:

```yaml
imports:
  -
    resource: 'Env_%env("foo")%.yaml'
```
Changes for Integrators

Frontend Login (1)

- TYPO3 v10.2 now includes an Extbase-version of the frontend login functionality.

- This solution has a few advantages:
  - Modify the templates more easily.
  - Send out HTML-based password recovery emails.
  - Adjust and modify validators to enforce password restrictions.

- The new Extbase plugin is available out-of-the-box for new installations.

- Existing TYPO3 instances will continue to use the old templates.

- Integrators can switch between the "old" and the "new" plugin by using a feature toggle.
Changes for Integrators

Frontend Login (2)

- A password recovery form has been added as part of the Extbase plugin.
- Users can request a password change and will receive an email with a link which redirects them to the form.
- Default password validation rules:
  - NotEmptyValidator - passwords cannot be empty.
  - StringLengthValidator - passwords must have a minimum length.
Changes for Integrators

Frontend Login (3)

- These validation rules can be customized.
- For example:

```php
plugin.tx_felogin_login {
    settings {
        passwordValidators {
            10 = TYPO3\CMS\Extbase\Validation\Validator\AlphanumericValidator
            20 {
                className = TYPO3\CMS\Extbase\Validation\Validator\StringLengthValidator
                options {
                    minimum = 12
                    maximum = 32
                }
            }
            30 = Vendor\MyExtension\Validation\Validator\MyCustomPasswordPolicyValidator
        }
    }
}
```
Changes for Integrators

Localization Management Platform

- **crowdin** aims to replace the existing **Pootle** solution as a localization/translation management platform.
- A feature toggle has been added in TYPO3 v10.2 that uses crowdin.com as the source for translations if enabled.
- Please note: this is in **beta status**.
- Read more about the **initiative**.

[crowdin logo]
Changes for Integrators

Multiple Sitemaps

- It is now possible to configure multiple sitemaps.

- Syntax:

```plaintext
plugin.tx_seo {
    config {
        <sitemapType> {
            sitemaps {
                <unique key> {
                    provider = TYPO3\CMS\Seo\XmlSitemap\RecordsXmlSitemapDataProvider
                    config {
                        ...
                    }
                }
            }
        }
    }
}
```
Changes for Integrators

**HTML5 attribute `nomodule`**

- The HTML5 attribute `nomodule` is now supported when including JavaScript files in TypoScript.

```plaintext
page.includeJSFooter.file = path/to/classic-file.js
page.includeJSFooter.file.nomodule = 1
```

- This attribute prevents a script from being executed in browsers that support module scripts.

- Read more about the standard in the specification and about the concept of modules.
Changes for Integrators

Sorting of Forms

- Forms can now be sorted in either ascending or descending order.
- Two new settings were introduced: sortByKeys and sortAscending.
- Forms are initially sorted by their name and their file UID (ascending).
- To change the sorting, the following configuration needs to be added in the YAML configuration file:

```yaml
TYPO3:
  CMS:
    Form:
      persistenceManager:
        sortByKeys: ['name', 'fileUid']
        sortAscending: true
```
Changes for Integrators

Link Validator (1)

- The Link Validator now supports additional configuration for external links.
- Values for `httpAgentUrl` and `httpAgentEmail` should be provided.
- Settings `headers`, `method` and `range` are advanced settings.

```plaintext
mod.linkvalidator {
  linktypesConfig {
    external {
      httpAgentName = ...
      httpAgentUrl = ...
      httpAgentEmail = ...
      headers {
      }
      method = HEAD
      range = 0-4048
    }
  }
}
```
Changes for Integrators

Link Validator (2)

- Link Validator now marks broken **external** links in the RTE too.
- This feature was only available for internal links.
- It is recommended to run the Link Validator as a Scheduler task to regularly crawl for broken links.
Chapter 3:
Changes for Developers
Changes for Developers

Widget ViewHelpers

- Widget ViewHelpers set a session cookie in the frontend under certain circumstances.
- As this is not always desired (for example due to GDPR), this can be controlled now.
- A boolean `storeSession` has been introduced that lets developers enable/disable this feature.

```html
<f:widget.autocomplete
  for="name"
  objects="{posts}"
  searchProperty="author"
  storeSession="false" />
```
Changes for Developers

PSR-14 Events in FAL

- Approximately 40 new PSR-14 based Events have been introduced in the File Abstraction Layer (FAL).
- They replace existing Extbase Signal/Slots.
- Using the Signals continues to work (without producing any deprecation message!). However, the Signals in the FAL will likely be removed in TYPO3 v11.
- Extension authors are advised to migrate their code and use Events.
- Review the new PHP classes to learn more about PSR-14.
Changes for Developers

PSR-14 Events in the TYPO3 Core

A number of new PSR-14 Events replace Signal/Slots in the TYPO3 core:

- TYPO3\CMS\Core\Imaging\Event\ModifyIconForResourcePropertiesEvent
- TYPO3\CMS\Core\DataHandling\Event\IsTableExcludedFromReferenceIndexEvent
- TYPO3\CMS\Core\DataHandling\Event\AppendLinkHandlerElementsEvent
- TYPO3\CMS\Core\Configuration\Event\AfterTcaCompilationEvent
- TYPO3\CMS\Core\Database\Event\AlterTableDefinitionStatementsEvent
- TYPO3\CMS\Core\Tree\Event\ModifyTreeDataEvent
- TYPO3\CMS\Backend\Backend\Event\SystemInformationToolbarCollectorEvent
Changes for Developers

Prepared Statements

- Two new PHP classes have been introduced to load and parse PageTSconfig:
  - TYPO3\CMS\Core\Configuration\Loader\PageTsConfigLoader
  - TYPO3\CMS\Core\Configuration\Parser\PageTsConfigParser

- For example:

```php
// Fetch all available PageTS of a page/rootline:
$loader = GeneralUtility::makeInstance(PageTsConfigLoader::class);
$tsConfigString = $loader->load($rootLine);

// Parse the string and apply conditions:
$parser = GeneralUtility::makeInstance(
    PageTsConfigParser::class, $typoScriptParser, $hashCache
);

$pagesTSconfig = $parser->parse($tsConfigString, $conditionMatcher);
```
Changes for Developers

Prepared Statements

- The `pdo_mysql` driver uses prepared statements by default now.
- In TYPO3 < v10.2, *emulated prepared statements* are used. This means, all returned values of a query were strings.
- This behavior has changed and prepared statements are used which return native data types.
- For example: values of a column defined as integer are returned in PHP as `int`.
- This feature can be deactivated by setting the option `PDO::ATTR_EMULATE_PREPARES` in your database connection.
Changes for Developers

Lazy Loading Proxy

- A method `getUid()` has been added to the class `TYPO3\CMS\Extbase\Persistence\Generic\LazyLoadingProxy`.
- This allows developers to fetch the UID of the proxied object without fetching the object from the database.
Changes for Developers

Denote Site Language Awareness

- A SiteLanguageAwareInterface has been introduced.
- The interface can be used to denote a class as aware of the site language.
- Routing aspects, that take the site language into account, are now using the SiteLanguageAwareInterface in addition to the SiteLanguageAwareTrait.
Changes for Developers

System Log API

- The following options have been removed from TYPO3’s default configuration:
  - $GLOBALS[‘TYPO3_CONF_VARS’][‘SYS’][‘systemLog’]
  - $GLOBALS[‘TYPO3_CONF_VARS’][‘SYS’][‘systemLogLevel’]

- Extension authors are advised to use the Logging API and remove the systemLog options.
Changes for Developers

Native List Pagination

- Native support for the pagination of lists such as arrays or QueryResults of Extbase has been introduced.
- The PaginatorInterface defines a basic set of methods.
- The AbstractPaginator class holds the main pagination logic.
- This enables developers to implement all kinds of paginators.

```php
use TYPO3\CMS\Core\Pagination\ArrayPaginator;

$items = ['apple', 'banana', 'strawberry', 'raspberry', 'ananas'];
$currentPageNumber = 3;
$itemsPerPage = 2;

$paginator = new ArrayPaginator($itemsToBePaginated, $currentPageNumber, $itemsPerPage);

$paginator->getNumberOfPages(); // returns 3
$paginator->getCurrentPageNumber(); // returns 3
$paginator->getKeyOfFirstPaginatedItem(); // returns 5
$paginator->getKeyOfLastPaginatedItem(); // returns 5
```
Changes for Developers

Service API

- Argument $excludeServiceKeys is used for skipping certain services when using a chain.
- The argument has been changed from a comma-separated list to an array in TYPO3 v10.2.
- This change affects the Service API within the following components:
  - GeneralUtility::makeInstanceService()
  - ExtensionManagementUtility::findService()
- Passing a comma-separated list still works but has been marked as deprecated.
System Extension "Form"

Summary

Several changes have been made to the system extension "Form". These changes affect editors, integrators as well as developers.

Some of the changes are based on concepts developed during the TYPO3 Initiative Week (T3INIT19).
System Extension "Form"

Form Setup

- Three files were used previously: BaseSetup.yaml, FormEditorSetup.yaml, and FormEngineSetup.yaml.
- This has been streamlined and consolidated into one file now: FormSetup.yaml.
- This file contains the basic setup including imports of the configuration for validators, form elements and finishers.
- All previously used inheritances and mixins have been resolved which makes it very easy to understand the entire configuration.
System Extension "Form"

YAML Files

- YAML files now use the TYPO3 core YAML file loader.
- This enabled features such as:
  - Import of other YAML files via imports directive.
  - Replacement of %placeholders%. 
System Extension "Form"

Multi-step Wizard

- A new JavaScript module MultiStepWizard has been introduced, that adds the following features:
  - Navigation to previous steps.
  - Steps support descriptive labels such as "Start" or "Finish", rather than the numerical indicator "Step x of y".
  - Optimized configuration structure.
- See ChangeLog for JavaScript code examples.
- This new feature improves the user experience significantly: backend users will notice an enhanced form creation wizard.
The record browser can now be configured to use custom tables:

```markdown
TYPO3:
  CMS:
    Form:
      prototypes:
        standard:
          formElementsDefinition:
            MyCustomElement:
              formEditor:
                editors:
                  # ...
    300:
      identifier: myRecord
      # ...
    browsableType: tx_myext_mytable
    propertyPath: properties.myRecordUid
    # ...
```
System Extension "Form"

Record Browser

The button icon of the record browser are now configurable:

```yaml
TYPO3:
  CMS:
    Form:
      prototypes:
        standard:
          formElementsDefinition:
            MyCustomElement:
              formEditor:
                editors:
                  # ...
                  300:
                    identifier: contentElement
                    # ...
                    browsableType: tt_content
                    iconIdentifier: mimetypes-x-content-text
                    propertyPath: properties.contentElementUid
                    # ...
```
System Extension "Form"

Record Browser

- A new RenderFormValue-ViewHelper lets integrators/developers access single form values in templates:

```
<p>
The following message was just sent by
<formvh:renderFormValue renderable="{page.rootForm.elements.name}" as="formValue">
  {formValue.processedValue}
</formvh:renderFormValue>:
</p>

<blockquote>
<formvh:renderFormValue renderable="{page.rootForm.elements.message}" as="formValue">
  {formValue.processedValue}
</formvh:renderFormValue>
</blockquote>
```
System Extension "Form"

Fieldset Labels

- The section element `<fieldset>` is now accessible in templates.
- By default this affects the `SummaryPage` form element as well as the `EmailToReceiver` and `EmailToSender` finishers.
- Typical use-case:
  A form with a shipping and a billing address. Both sections could have a field with the same name, e.g. `street`. It is now possible to distinguish between both fields by using fieldset labels.
Predefined allowedMimeTypes of the following form elements have been marked deprecated:
  - FileUpload
  - ImageUpload

All valid MIME types must be explicitly listed in the form definition now (predefined MIME types will be removed in TYPO3 v11)

Integrators can already activate the new behaviour in TYPO3 v10 by using a feature toggle.
Form Mixins

- Mixins have been marked as **deprecated** and should not be used anymore.
- This affects all inheritances from `TYPO3.CMS.Form.mixins.*`.
- Migration options:
  - Embed the essential parts from `TYPO3.CMS.Form.mixins.*`, or
  - migrate them to custom mixins.
Chapter 5:
Deprecated/Removed Functions
The following functions of the FormEngine have been marked as deprecated:

- `setFormValueOpenBrowser()` (use `FormEngine.openPopupWindow()` instead)
- `setFormValueFromBrowseWin()` (use `FormEngine.setSelectOptionFromExternalSource()` instead)
- `setHiddenFromList()` (use `FormEngine.updateHiddenFieldValueFromSelect()` instead)
- `setFormValueManipulate()` (no replacement, as this is internal logic)
- `setFormValue_getFObj()` (use `use FormEngine.getFormElement()` instead)
Deprecated/Removed Functions

Signal/Slot

The following Signal/Slots were replaced by PSR-14 events and therefore have been marked as deprecated:

- TYPO3\CMS\Core\Imaging\IconFactory::buildIconForResourceSignal
- TYPO3\CMS\Core\Database\SoftReferenceIndex::setTypoLinkPartsElement
- TYPO3\CMS\Core\Database\ReferenceIndex::shouldExcludeTableFromReferenceIndex
- TYPO3\CMS\Core\Utility\ExtensionManagementUtility::tcaIsBeingBuilt
- TYPO3\CMS\Install\Service\SqlExpectedSchemaService::tablesDefinitionIsBeingBuilt
- TYPO3\CMS\Core\Tree\TableConfiguration\DatabaseTreeDataProvider::PostProcessTreeData
- TYPO3\CMS\Backend\Backend\ToolbarItems\SystemInformationToolbarItem::getSystemInformation
- TYPO3\CMS\Backend\Backend\ToolbarItems\SystemInformationToolbarItem::loadMessages
The method `GeneralUtility::getApplicationContext()` has been marked **deprecated**.

The following method should be used instead:

`TYPO3\CMS\Core\Core\Environment::getContext()`.
Deprecated/Removed Functions

Miscellaneous

- The property $environmentService of the following class has been marked as **deprecated**: `TYPO3\CMS\Extbase\Mvc\Web\Response`. (inject the environment service manually in the subclass if required)
- The following internal class has been marked as **deprecated**: `TYPO3\CMS\Extbase\Mvc\Controller\AbstractController`.
- The following method has been marked as **deprecated**: `TYPO3 CMS Backend Utility BackendUtility::TYPO3_copyRightNotice`. (use the new PHP class Typo3Copyright and its method `getCopyrightNotice()` instead)
Deprecated/Removed Functions

Miscellaneous

- The PHP property `TypoScriptFrontendController->divSection` has been marked as deprecated.
  (use `$GLOBALS['TSFE']ㄧ->additionalHeaderData[]` to add JavaScript within uncached content instead)

- As part of the Symfony 5.0 support in TYPO3 v10.2, incompatible environment related dependency injection services have been removed:
  - `env.is_unix`
  - `env.is_windows`
  - `env.is_cli`
  - `env.is_composer_mode`
Sources and Authors

Sources

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

**Release Infos:**
- [https://get.typo3.org/release-notes/10.x/TYPO3_CMS_10.2.0](https://get.typo3.org/release-notes/10.x/TYPO3_CMS_10.2.0)
- **TYPO3 v10 Changelog**
- `typo3/sysext/core/Documentation/Changelog/10.2/*`

**TYPO3 Bug-/Issuetracker:**

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

TYPO3 CMS What’s New Team:
Paul Blondiaux, Pierrick Caillon, Andreas Fießer
Richard Haeser, Jigal van Hemert, Henrietta Kucsovan, Sinisa Mitrovic,
Michael Schams and Roberto Torresani

https://typo3.org/help/documentation/whats-new/

Licensed under Creative Commons BY-NC-SA 3.0