

TYP03 Version 10.1 – What's New

Zusammenfassung der neuen Funktionen, Änderungen
und Verbesserungen

erstellt von:

Michael Schams

übersetzt von:

Henrietta Kucsovan

18/October/2019

Creative Commons BY-NC-SA 3.0



TYPO3 Version 10.1 - What's New

Kapitelübersicht

Einführung

Backend User Interface

Änderungen für Integriatoren

Änderungen für Entwickler

Veraltete/Entfernte Funktionen

Sonstiges

Quellen und Autoren

Einführung

Fakten

Einführung

TYPO3 Version 10.1 - Fakten

- Veröffentlichungsdatum: 01 Oktober 2019
- Releasetyp: Sprint Release



Einführung

Zusammenfassung

TYPO3 Version 10.1 ist der zweite Sprint Release auf dem Weg zur LTS (langfristige Unterstützung) im Jahr 2020.

Die neue Version hat seit der vor zehn Wochen veröffentlichten Vorgängerversion 10.0 mehr als 240 Git-Commits erhalten (überprüfte, getestete und genehmigte Quellcodeänderungen).

Obwohl Backend-Benutzer nicht viele offensichtliche Änderungen oder wichtige neue Funktionen als solche sehen, erhielt TYPO3 Version 10.1 eine Reihe von Verbesserungen unter der Haube.

Einführung

Systemvoraussetzungen

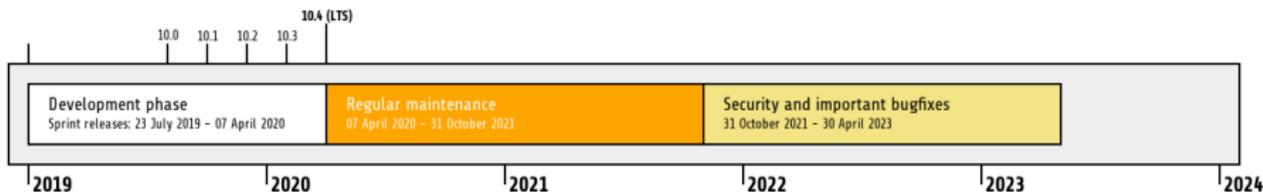
- PHP Version 7.2 or 7.3
- PHP Einstellungen:
 - `memory_limit` \geq 256M
 - `max_execution_time` \geq 240s
 - `max_input_vars` \geq 1500
 - Die Kompilierungsoption `--disable-ipv6` darf nicht benutzt werden
- Die meisten von **Doctrine DBAL** unterstützten Datenbankserver arbeiten auch mit TYPO3. Getestete DB-Engines sind zum Beispiel:



Einführung

Zeitplan für Entwicklung, Veröffentlichung und Instandhaltung

TYPO3 v10



Erweiterter Support

Die [TYPO3 GmbH](#) bietet weitere Supportmöglichkeiten für TYPO3 v10 LTS auch nach dem 30. April 2023 für bis zu zwei weitere Jahre.

Einführung

TYP03 v10 Roadmap

Voraussichtliche Veröffentlichung und deren Hauptfokus:

- 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/>

Einführung

Installation

- Empfohlene *klassische* Installationsschritte unter Linux/Mac OS X (DocumentRoot ist beispielsweise /var/www/site/htdocs):

```
$ cd /var/www/site
$ wget --content-disposition get.typo3.org/10.1
$ tar xzf typo3_src-10.1.0.tar.gz
$ cd htdocs
$ ln -s ../typo3_src-10.1.0 typo3_src
$ ln -s typo3_src/index.php
$ ln -s typo3_src/typo3
$ touch FIRST_INSTALL
```

- Symbolische Links unter Microsoft Windows:
 - Unter Windows XP/2000 kann `junction` benutzt werden
 - Unter Windows Vista und Windows 7 oder höher kann `mklink` benutzt werden

Einführung

Installation mit composer

- Installation mit *composer* unter Linux, Mac OS X und Windows 10:

```
$ cd /var/www/site/  
$ composer create-project typo3/cms-base-distribution typo3v10 ^10.1
```

- Alternativ eine benutzerdefinierte *composer.json* Datei erstellen und ausführen:

```
$ composer install
```

Weitere *composer.json* Beispielsdateien können unter:

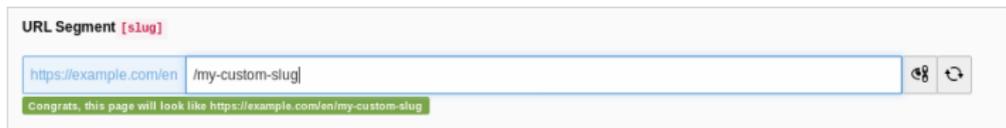
<https://composer.typo3.org> heruntergeladen werden.

Kapitel 1: Backend User Interface

Backend User Interface

Slug-Updates und Weiterleitungen (1)

- Wenn Backend-Benutzer den URL-Pfad einer Seite ändern (den sogenannten "Slug"), wird die alte URL nicht mehr verfügbar.
- Dies führt möglicherweise zu einem "Seite nicht gefunden" Fehler, auch für die URLs aller Unterseiten.

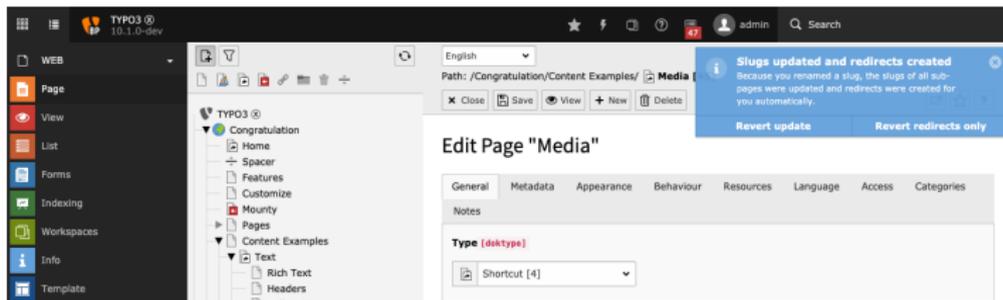


- Im Rahmen der TYPO3 Version 10.1, wird dies durch zwei Aktionen verhindert:
 - Slugs werden für alle Unterseiten automatisch aktualisiert
 - Es werden Weiterleitungen von den alten zu den neuen URLs erstellt

Backend User Interface

Slug-Updates und Weiterleitungen (2)

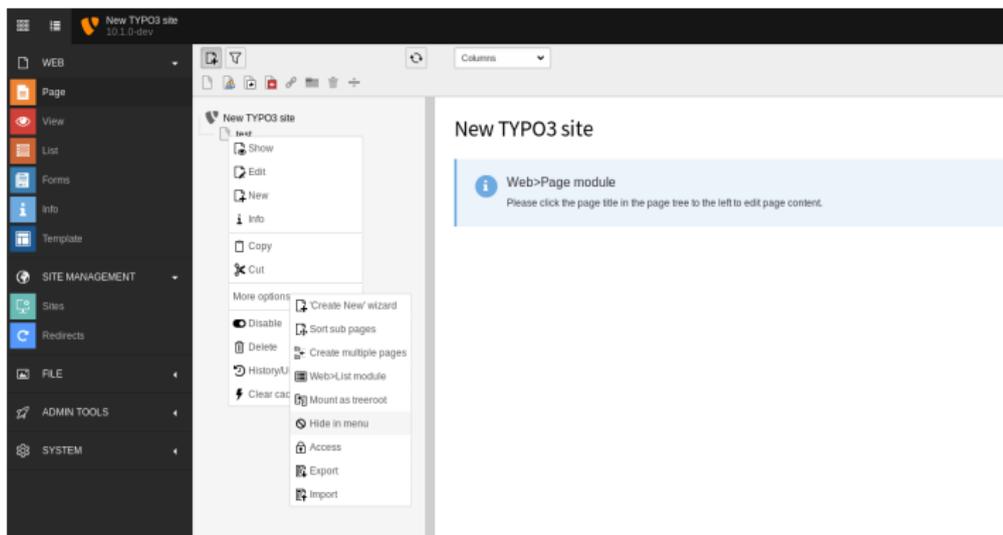
- Backend-Benutzer werden über diese Aktionen informiert und können die Änderungen bei Bedarf einfach per Mausklick rückgängig machen:



Backend User Interface

Im Menü ein-/ausblenden

Ein neuer Eintrag wurde zum Kontextmenü hinzugefügt, um Seiten im Menü anzuzeigen/auszublenden.



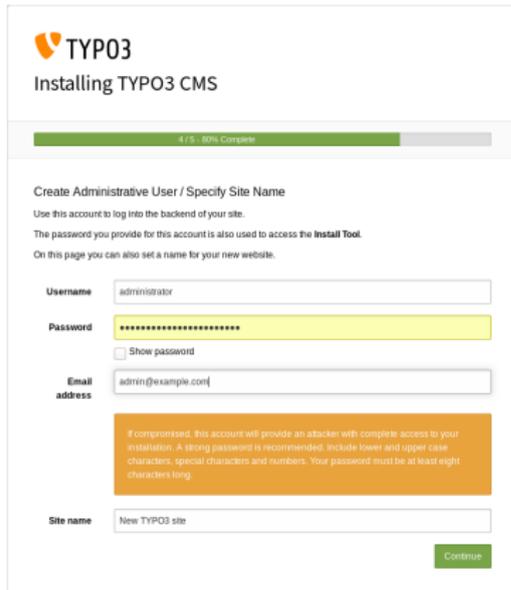
Kapitel 2: Änderungen für Integratoren

Änderungen für Integriatoren

E-Mail-Adresse des Administrators

Während des Installationsvorgangs kann jetzt eine E-Mail-Adresse eingegeben werden. Diese Adresse wird für den initialen Admin Backend-Benutzer verwendet.

Die selbe Option gibt es im Wartungsmodul **Create Administrative User** des Install Tools.



The screenshot shows the TYPO3 CMS installation interface. At the top, the TYPO3 logo and the text 'Installing TYPO3 CMS' are visible. A progress bar indicates '4 / 5 - 80% Complete'. The main heading is 'Create Administrative User / Specify Site Name'. Below this, there are instructions: 'Use this account to log into the backend of your site.' and 'The password you provide for this account is also used to access the Install Tool.' A note states: 'On this page you can also set a name for your new website.' The form contains three input fields: 'Username' with the value 'administrator', 'Password' with masked characters and a 'Show password' checkbox, and 'Email address' with the value 'admin@example.com'. A warning box below the password field reads: 'If compromised, this account will provide an attacker with complete access to your installation. A strong password is recommended. Include lower and upper case characters, special characters and numbers. Your password must be at least eight characters long.' At the bottom, there is a 'Site name' field with the value 'New TYPO3 site' and a green 'Continue' button.

Änderungen für Integratoren

Cache Speichertyp (1)

- TYPO3 verfügt über ein flexibles Caching-System mit einer Standardkonfiguration, die für die meisten Anwendungsfälle ideal ist.
- Der Speichertyp kann nun auch konfiguriert werden, um die Caches zu optimieren und die Leistung je nach individueller Umgebung zu erhöhen.
 - Wählen Sie den Speicher **database** für eine Standardumgebung aus oder wenn beispielsweise ein Netzwerkdateisystem (NFS) verwendet wird.
 - Wählen Sie **file system** wenn zum Beispiel eine verteilte Datenbankeinrichtung verwendet wird.
 - Wählen Sie **custom cache settings** um den Speichertyp für jeden Cache individuell zu konfigurieren.
- Bei komplexeren Installationen und speicherbasierten Caches sollten zum Beispiel [Redis](#) oder [Memcached](#) berücksichtigt werden.

Änderungen für Integriatoren

Cache Speichertyp (2)

- Backend: **ADMIN TOOLS** → **Einstellungen** → **Konfigurationsvoreinstellungen:**

Cache settings

...

Prefer database storage for caching [Active]

Prefer File storage for caching

Custom configuration

Custom cache settings:

<code>SYSicaching\cacheConfigurations\hash\backend</code>	<input type="text" value="TYPO3\CMS\Core\Cache\Backend\Typo3DatabaseBackend"/>
<code>SYSicaching\cacheConfigurations\pages\backend</code>	<input type="text" value="TYPO3\CMS\Core\Cache\Backend\Typo3DatabaseBackend"/>
<code>SYSicaching\cacheConfigurations\pagesection\backend</code>	<input type="text" value="TYPO3\CMS\Core\Cache\Backend\Typo3DatabaseBackend"/>
<code>SYSicaching\cacheConfigurations\imagesizes\backend</code>	<input type="text" value="TYPO3\CMS\Core\Cache\Backend\Typo3DatabaseBackend"/>
<code>SYSicaching\cacheConfigurations\rootline\backend</code>	<input type="text" value="TYPO3\CMS\Core\Cache\Backend\Typo3DatabaseBackend"/>

Änderungen für Integriatoren

Site-Konfiguration

- Wenn eine neue Seite auf Root Level erstellt wird, wird damit automatisch eine Standard-Site-Konfiguration generiert.
- Demzufolge kann eine grundlegende TYPO3-Site schnell aufgesetzt werden.
- Die Site-Konfigurationsfunktionen:
 - eine vordefinierte Identifier (e.g. `site-42-a1d0c6e83f`)
 - ein Einstiegspunkt (e.g. `https://example.com/site-42`)
 - eine default Sprache (e.g. Englisch)

Änderungen für Integriatoren

Konflikte bei Redirects (1)

- Ein neuer Symfony-Befehl wurde eingeführt, um Weiterleitungen zu erkennen, die mit Seiten-URLs in Konflikt stehen.
- Führen Sie den Befehl in der CLI aus:
(Der optionale Parameter `--site` begrenzt die Prüfung auf eine bestimmte Seite)

```
typo3v10:~/www$ ./bin/typo3 redirects:checkintegrity
Redirect (Host: *, Path: /page-1) conflicts with /page-1
typo3v10:~/www$ █
```

- Der Befehl ist auch als Scheduler-Task verfügbar:

	ID	Task	Type	Frequency	Parallel Execution	Last Execution	Next Execution	
<input type="checkbox"/>	1	Execute console commands (scheduler) redirects:checkintegrity	Recurring	86400	No	-	15-09-19 06:12	    

Änderungen für Integratoren

Konflikte bei Redirects (2)

- Im Modul Berichte kann auch eine Liste widersprüchlichen Weiterleitungen zugegriffen werden:

Conflicting Redirects

1 conflicting redirects
These redirects cause a conflict as there are pages that are still accessible with the same URL.

- /page-1
Source Domain: *
Source Path: /page-1

- **Anmerkung:** Der Befehl muss erneut ausgeführt werden, um die Liste 'zurückzusetzen'. Durch Beheben des Problems (z.B. durch Entfernen der Weiterleitung) wird die Liste nicht geleert.

Änderungen für Integratoren

Distributionspakete

- Distributionen können jetzt Site-Konfigurationsdateien bereitstellen.
- Erstellen Sie ein Verzeichnis/eine Datei im Distributionspaket wie folgt:
`Initialisation/Site/<siteIdentifier>/config.yaml`
- Ähnlich wie bei Assets, die nach `fileadmin/` verschoben werden, werden Standardkonfigurationen in den Ordner `config/` verschoben.
- Wenn das Zielverzeichnis bereits vorhanden ist, wird keine Änderung an der vorhandenen Konfiguration vorgenommen.

Änderungen für Integratoren

Application Context in CLI

- Der aktuelle Application Context wird nun in CLI-Anforderungen neben der TYPO3-Versionsnummer angezeigt:

```
typo3v10:~/www$ ./bin/typo3 --version
TYPO3 CMS 10.1.0 (Application Context: Production)
typo3v10:~/www$ █
```

Änderungen für Integratoren

Vimeo Video-Rendering

- Der Parameter `api=1` in Vimeo-Video-URLs ermöglicht API-Interaktionen mit dem Videoplayer (z.B. Hinzufügen von Schaltflächen zur Steuerung des Videos).
- Integratoren können diesen Parameter jetzt auf zwei verschiedene Weisen einstellen:

- Mit TypoScript:

```
lib.contentElement.settings.media.additionalConfig.api = 1
```

- In Fluid, mit Media-ViewHelper:

```
<f:media
  file="{file}"
  alt="{file.properties.alternative}"
  title="{file.properties.title}"
  additionalConfig="{api: 1}"
/>
```

Änderungen für Integratoren

Hochladen von Dateien

- Es ist jetzt möglich, die Standardaktion beim Hochladen von Dateien im Dateilistenmodul durch Drag and Drop zu konfigurieren.
- Benutzer TSConfig:

```
# Set default to replace:
```

```
options.file_list.uploader.defaultAction = replace
```

```
# Set default to rename:
```

```
options.file_list.uploader.defaultAction = rename
```

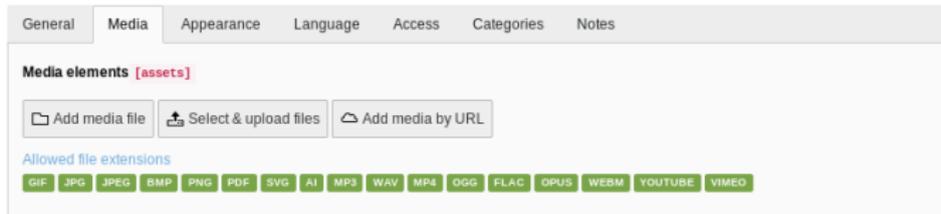
```
# Set default to cancel:
```

```
options.file_list.uploader.defaultAction = cancel
```

Änderungen für Integriatoren

Tasten für Medienelemente

- Die Tasten **"Add media by URL"** und **"Select & upload files"** können nun unabhängig voneinander aktiviert/deaktiviert werden.



- Im folgenden Beispiel werden beide Tasten ausgeblendet:

```
$GLOBALS['TCA']['pages']['columns']['media']['config']['appearance'] = [  
    'fileUploadAllowed' => false,  
    'fileByUrlAllowed' => false,  
];
```

Änderungen für Integriatoren

Admin-Panel

- Der Admin-Panel enthält ein neues Panel **USER_INT** unter dem Modul 'Info'.



The screenshot shows the Admin Panel interface with the 'USER_INT' panel selected. The panel title is 'USER_INT' and the identifier is 'INT_SCRIPT.4c62089c320a4e49719267400a6131c7'. Below the title is a table with two columns: 'Key' and 'Value'.

Key	Value
TYPE	"FUNC"
userFunc	"Vendor\MyExtension\ExampleTime->printTime"

The bottom status bar shows 'TypoScript (7 Messages)', 'Info (Load Time: 94 ms)', 'Debug (0 Errors/Warnings)', and 'Settings'.

Kapitel 3: Änderungen für Entwickler

Änderungen für Entwickler

Cache Dependency Injection (1)

- Extension-Entwickler wird empfohlen, Caches direkt einzuspeisen, statt den CacheManager zu verwenden.
- Dies erfordert einige einfache Änderungen, siehe folgendes Beispiel:
- **Vorher:**

```
class MyClass
{
    /**
     * @var TYPO3\CMS\Core\Cache\Frontend\FrontendInterface
     */
    private $cache;

    public function __construct()
    {
        $cacheManager = GeneralUtility::makeInstance(CacheManager::class);
        $this->cache = $cacheManager->getCache('my_cache');
    }
}
```

Änderungen für Entwickler

Cache Dependency Injection (2)

- Seit **TYPO3 v10.1**, sollte diese Klasse folgenderweise aussehen:

```
class MyClass
{
    /**
     * @var TYPO3\CMS\Core\Cache\Frontend\FrontendInterface
     */
    private $cache;

    public function __construct(FrontendInterface $cache)
    {
        $this->cache = $cache;
    }
}
```

Änderungen für Entwickler

Cache Dependency Injection (3)

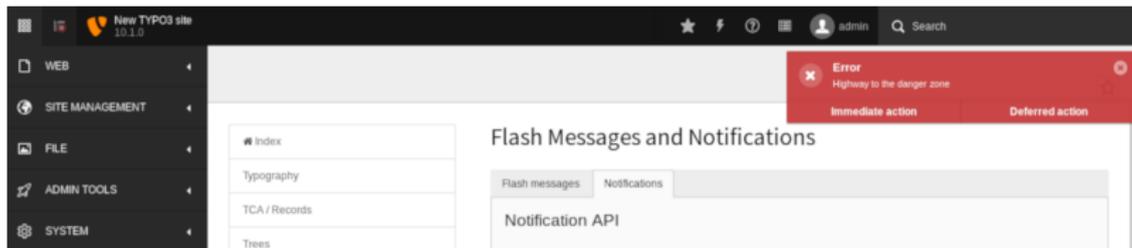
- ... und folgende Container-Service-Konfiguration ist erforderlich:

```
services:  
  cache.my_cache:  
    class: TYPO3\CMS\Core\Cache\Frontend\FrontendInterface  
    factory: ['@TYPO3\CMS\Core\Cache\CacheManager', 'getCache']  
    arguments: ['my_cache']  
  
MyClass:  
  arguments:  
    $cache: '@cache.my_cache'
```

Änderungen für Entwickler

JavaScript Benachrichtigungen

- JavaScript Benachrichtigungen im BE unterstützen nun Buttons.



Änderungen für Entwickler

Broadcast-Kanäle und Messaging (1)

- Es ist jetzt möglich, 'Broadcast-Nachrichten' mit JavaScript zu senden und zu empfangen.

Die API wird vorerst als **intern** betrachtet und kann sich jederzeit ändern, bis sie als 'stabil' betrachtet wird.

- Beispiel für das **Senden** einer Nachricht:

```
require(['TYPO3\CMS\Backend\BroadcastService'], function (BroadcastService) {
  const payload = {
    componentName: 'my_extension',
    eventName: 'my_event',
    foo: 'bar'
  };
  BroadcastService.post(payload);
});
```

Änderungen für Entwickler

Broadcast-Kanäle und Messaging (2)

- Beispiel für den **Empfang** der Nachricht:

```
define([], function() {
  document.addEventListener('typo3:my_component:my_event', (e) => eventHandler(e.detail));
  function eventHandler(detail) {
    // output contains key 'foo' as the payload
    console.log(detail);
  }
});
```

- Siehe developer.mozilla.org für weitere Details.

Änderungen für Entwickler

PSR-17 HTTP Message Factories

- Die [PSR-17](#) HTTP Message Factories Implementierung wurde hinzugefügt.
- HTTP Message Factory-Schnittstellen sollten als Abhängigkeiten für Anforderungshandler oder Dienste verwendet werden, die PSR-7-Nachrichtenobjekte erstellen.
- PSR-17 besteht aus sechs Factory Schnittstellen:
 - `\Psr\Http\Message\RequestFactoryInterface`
 - `\Psr\Http\Message\ResponseFactoryInterface`
 - `\Psr\Http\Message\ServerRequestFactoryInterface`
 - `\Psr\Http\Message\StreamFactoryInterface`
 - `\Psr\Http\Message\UploadedFileFactoryInterface`
 - `\Psr\Http\Message\UriFactoryInterface`
- Siehe die [Dokumentation](#) für ein Beispiel im Code.

Änderungen für Entwickler

PSR-18 HTTP Client

- Der [PSR-18](#) HTTP Client Implementierung wurde hinzugefügt.
- Entwickler können HTTP-Anforderungen basierend auf PSR-7 Nachrichtenobjekten generieren ohne auf eine bestimmte HTTP-Client-Implementierung angewiesen zu sein.
- Dies ersetzt den vorhandenen Wrapper [Guzzle](#) nicht, sondern bietet eine gewöhnlichere Alternative.
- PSR-18 besteht aus einer Client-Schnittstelle und drei Ausnahmeschnittstellen:
 - `\Psr\Http\Client\ClientInterface`
 - `\Psr\Http\Client\ClientExceptionInterface`
 - `\Psr\Http\Client\NetworkExceptionInterface`
 - `\Psr\Http\Client\RequestExceptionInterface`
- Siehe die [Documentation](#) für ein Codebeispiel.

Änderungen für Entwickler

RequestFactory Middleware Handler

- Es ist jetzt möglich, benutzerdefinierte Middleware-Handler als Array zu definieren.
- Die RequestFactory erstellt einen Handler-Stack basierend auf dem Array
`$GLOBALS['TYPO3_CONF_VARS']['HTTP']['handler']` und injiziert ihn in den Client.
- Zum Beispiel:

```
use \TYPO3\CMS\Core\Utility\GeneralUtility;
use \Vendor\MyExtension\Middleware\Guzzle\CustomMiddleware;
use \Vendor\MyExtension\Middleware\Guzzle\SecondCustomMiddleware;

# Add custom middleware to default Guzzle handler stack
$GLOBALS['TYPO3_CONF_VARS']['HTTP']['handler'][] =
    (GeneralUtility::makeInstance(CustomMiddleware::class))->handler();
$GLOBALS['TYPO3_CONF_VARS']['HTTP']['handler'][] =
    (GeneralUtility::makeInstance(SecondCustomMiddleware::class))->handler();
```

Änderungen für Entwickler

Benutzerdefinierte Dateiprozessoren

- Entwickler können nun ihre eigenen Dateiprozessoren registrieren.
- Fügen Sie diesen Code in die Datei `ext_localconf.php` ein:

```
$GLOBALS['TYPO3_CONF_VARS']['SYS']['fal']['processors']['ExampleImageProcessor'] = [  
    'className' => \Vendor\MyExtension\Resource\Processing\ExampleImageProcessor::class,  
    'before' => 'LocalImageProcessor',  
];
```

- Typische Anwendungsfälle:
 - Wasserzeichen auf Bilder setzen
 - hochgeladene Dateien in ein ZIP-Archiv komprimieren
 - bearbeitete Kopien von Bildern speichern
 - usw.

Änderungen für Entwickler

Extbase und Fluid

- Lassen Sie den Anbieternamen weg, wenn Sie Plugins mit `\TYPO3\CMS\Extbase\Utility\ExtensionUtility::registerPlugin()` registrieren.
- Verwenden Sie beispielsweise "Form" stat "TYPO3\CMS\Form" (erstes Argument)

```
\TYPO3\CMS\Extbase\Utility\ExtensionUtility::registerPlugin(  
    'Form',  
    'Formframework',  
    'Form',  
    'content-form',  
);
```

Änderungen für Entwickler

Sonstiges (1)

- Die `hashParameters` zur Berechnung der `HashBase` in der folgenden Klasse wurden geändert:

`TYPO3\CMS\Frontend\Controller\TypoScriptFrontendController`

- `gr_list` wurde durch `groupIds` ersetzt.
 - `cHash` wurde durch `dynamicArguments` ersetzt.
 - `domainStartPage` wurde durch `site (site identifier)` ersetzt.
- Zwei neue Ereignisse werden ausgelöst, wenn Datensätze zurückgesetzt werden:
 - `TYPO3\CMS\Backend\History\Event\BeforeHistoryRollbackStartEvent`
 - `TYPO3\CMS\Backend\History\Event\AfterHistoryRollbackFinishedEvent`

Änderungen für Entwickler

Sonstiges (2)

- Die Methode `set()` des Abfrage-Generators akzeptiert jetzt ein viertes Argument, um den Typ des benannten Parameters anzugeben:
`TYPO3\CMS\Core\Database\Query\QueryBuilder::set()`
(der Standardwert ist `\PDO::PARAM_STR`)

Kapitel 4: Veraltete/Entfernte Funktionen

Veraltete/Entfernte Funktionen

Veraltetes JavaScript (1)

- Zwei globale JavaScript-Variablen wurden als **veraltet** markiert:
 - T3_THIS_LOCATION
 - T3_RETURN_URL
- Die bekannte JavaScript-Funktion `jumpToUrl()` wurde als **veraltet** markiert. Migrationsoptionen:
 - verwenden Sie `window.location.href = '...'`;
 - oder benutzen Sie einen Link in HTML wie zum Beispiel `link`
- Die JavaScript-Funktion `jumpExt()` wurde als **veraltet** markiert.

Veraltete/Entfernte Funktionen

Veraltetes JavaScript (2)

- Das jQuery Plugin `jquery.clearable`, das eine Schaltfläche zum Löschen eines Eingabefeldes bereitgestellt wurde, wurde als **veraltet** markiert.
- Migration: verwenden Sie den Modul `TYPO3\CMS\Backend\Input\Clearable` und die Methode `clearable()` auf einem nativen `HTMLInputElement`.

```
require(['TYPO3\CMS\Backend\Input\Clearable'], function() {
    const inputField = document.querySelector('#myinput');
    if (inputField !== null) {
        inputField.clearable();
    }

    const clear = Array.from(document.querySelectorAll('.t3js-clearable')).filter(inputElement => {
        return !inputElement.classList.contains('t3js-datetimepicker');
    });
    clear.forEach(clearableField => clearableField.clearable());
});
```

Veraltete/Entfernte Funktionen

CLI Command Handler

- CLI-Befehle werden mit der Klasse `CommandApplication` verarbeitet.
- Diese Klasse ist ein Wrapper um die [Symfony Konsole](#).
- Die zuvor verwendete Schnittstelle und die Klasse `CommandRequestHandler` wurden als **veraltet** markiert:
 - `TYPO3\CMS\Core\Console\RequestHandlerInterface`
 - `TYPO3\CMS\Core\Console\CommandRequestHandler`

Veraltete/Entfernte Funktionen

Sonstiges

- Die Mail-Funktionalität sollte nicht in der Klasse `TYPO3\CMS\Frontend\ContentObject\ContentObjectRenderer` verwendet werden.
Daher wurde die Methode `sendNotifyEmail()` als **veraltet** markiert und wird in der TYPO3 v11 entfernt werden.
- Die Methode `editOnClick()`, die zum Generieren von JavaScript `onclick`-Zielen verwendet wurde, wurde in der folgenden Klasse als **veraltet** markiert:
`TYPO3\CMS\Backend\Utility\BackendUtility`.

Veraltete/Entfernte Funktionen

RecordHistory Handling

Änderungen wurden an folgender Klasse vorgenommen:

TYPO3\CMS\Backend\History\RecordHistory:

- Die Sichtbarkeit der Eigenschaften `changeLog` und `lastHistoryEntry` wurden auf `protected` geändert (und die Funktion `public` Getter wurde auch hinzugefügt).
- Die Sichtbarkeit der Methoden `getHistoryEntry()` und `getHistoryData()` wurde auf `protected` geändert.
- Die folgenden Methoden wurden als **veraltet** markiert:
 - `createChangeLog()`
 - `shouldPerformRollback()`
 - `getElementData()`
 - `performRollback()`
 - `createMultipleDiff()`
 - `setLastHistoryEntry()`

Veraltete/Entfernte Funktionen

XML-Sprachdateien

- Das XLIFF-Format wird für Sprachdateien seit TYPO3 v4.6 verwendet.
- Die Verwendung von XML-Sprachdateien wird nun als **veraltet** markiert und löst eine Warnung/einen Fehler aus.
- Dazu gehört auch die Ausführung des folgenden XML-Parameters:
`TYPO3\CMS\Core\Localization\Parser\LocallangXmlParser`

Kapitel 5: Sonstiges

Erhöhung der Sicherheit

- Externe Links, die von TypoLink generiert wurden, oder Links die `_blank` verwenden, zeigen jetzt das Attribut `rel="noopener noreferrer"`.
- Damit soll die Sicherheit der TYPO3-Website erhöht werden:
 - Der "**noopener**" weist den Browser an, den Link zu öffnen, ohne dem neuen Browserkontext Zugriff auf das Dokument zu gewähren, das ihn geöffnet hat.
 - Der "**noreferrer**" verhindert, dass der Browser beim Navigieren zu einer anderen Seite die Seitenadresse oder einen anderen Wert über die Funktion `Referer`: HTTP Header sendet.

YAML-Dateieinbindung

- YAML-Dateien können bereits von anderen YAML-Dateien mit der folgenden Syntax eingebunden werden:

```
imports:
  - { resource: "EXT:my_extension/Configuration/FooBar/Example.yaml" }

another:
  option: true
```

- Dies wurde erweitert, um Ressourcen relativ zur aktuellen YAML-Datei zu importieren:

```
imports:
  - { resource: "subfolder/AnotherExample.yaml" }
  - { resource: "../path/to/configuration/AnotherExample.yaml" }

another:
  option: true
```

Kapitel 6: Quellen und Autoren

Quellen und Autoren

Quellen

TYPO3 News:

- <https://typo3.org/project/news/>

Release Infos:

- https://get.typo3.org/release-notes/10.x/TYPO3\CMS_10.1.0
- [TYPO3 v10 ChangeLog](#)
- [typo3/sysext/core/Documentation/Changelog/10.1/*](https://typo3.org/typo3/sysext/core/Documentation/Changelog/10.1/*)

TYPO3 Bug-/Issuetracker:

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

TYPO3 und Fluid Git Repositories:

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

Quellen und Autoren

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

