

TYPO3 CMS 6.2 LTS – What's New

Übersicht der neuen Funktionen, Änderungen und Verbesserungen

Patrick Lobacher und Michael Schams

TYPO3 CMS 6.2 LTS - What's New

Kapitelübersicht

Einführung

Install Tool

Responsive Images

Änderungen im Backend

TSconfig & TypoScript

Package Management

Änderungen im System

Application Programming Interface

Extbase & Fluid

Upgrade zu TYPO3 CMS 6.2 LTS

MythBuster

Quellen und Autoren

Einführung (Die Fakten)

Einführung

TYPO3 CMS 6.2 LTS: Die Fakten

■ Hauptziele:

- Reibungslose Migration
- Robuste und sichere Grundlage
- Hohe Benutzerakzeptanz
- Moderne Technologien/Interoperabilität

■ Release Manager:

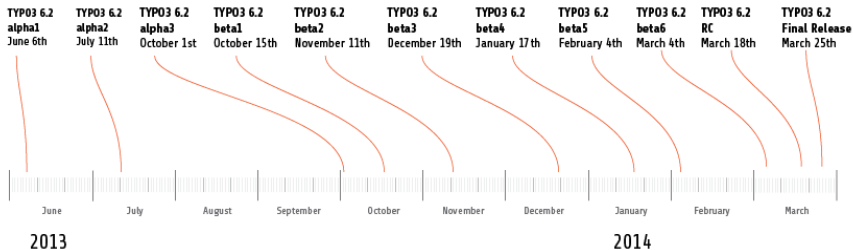
- Ernesto Baschny
ernesto.baschny (at) typo3.org
Twitter: @baschny



Einführung

TYPO3 CMS 6.2 LTS: Die Fakten

- Veröffentlichungsdatum: 25 März 2014
- Entwicklungs- und Veröffentlichungszeitplan:



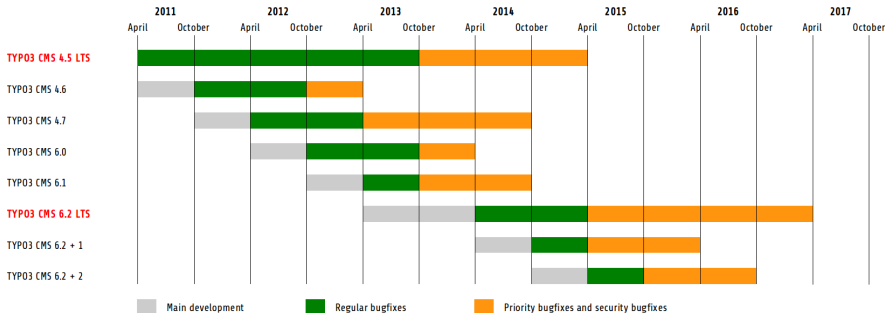
TYPO3 CMS 6.2 LTS: Die Fakten

- Systemvoraussetzungen
 - PHP v5.3.7 - v5.5.x
 - MySQL v5.1.x - v5.6.x
- Wird unterstützt bis mindestens: Ende März 2017
- TYPO3 CMS 6.2 ist ein **Long Term Support** (LTS) release (3 Jahre Support!)

Einführung

TYPO3 CMS 6.2 LTS: Die Fakten

■ Release Agenda:



Kapitel 1: Das "Install Tool"

Install Tool

Installation

- Für eine Installation wird nur noch ein Package benötigt:
`typo3_src-6.2.x.tar.gz` (Dateigröße ca. 20MB)
- "Dummy" und "Blank" Packages sind hinfällig
- Installation:
 - Extrahieren des Source Packages im Hauptverzeichnis des Web Servers
 - Erstellen der symbolischen Links, sofern erforderlich
 - Im Web Browser das Hauptverzeichnis aufrufen
 - TYPO3 Installer beginnt mit dem 1-2-3-4-5-Wizard

Install Tool

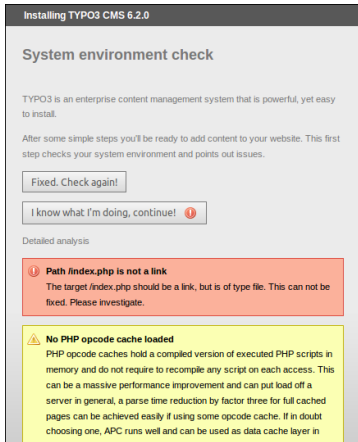
Installation

- Der Installer stellt sicher, dass alle erforderlichen Dateien und Verzeichnisse automatisch erstellt werden und am richtigen Platz sind
- Die folgenden symbolischen Links müssen existieren:
 - `typo3_src` (zeigt auf das TYPO3 Source Verzeichnis)
 - `typo3` (zeigt auf das Verzeichnis: `typo3_src/typo3`)
 - `index.php` (zeigt auf die Datei: `typo3_src/index.php`)
- Es sind keine weiteren Dateien oder Verzeichnisse für eine TYPO3 Installation notwendig
- Verzeichnis `t3lib` wurde entfernt
- Weitere Informationen im "TYPO3 Installation and Upgrade Guide"
<http://docs.typo3.org/typo3cms/InstallationGuide>

Install Tool

Neu-Entwicklung

- Es fand eine Neu-Entwicklung unter Verwendung von Fluid statt
- Der erste Schritt testet das System und meldet mögliche Probleme
- Diese können dann korrigiert oder ignoriert werden
- Ungültiger Core Setup (z.B. fehlende symbolische Links) wird ebenfalls angezeigt



Installing TYPO3 CMS 6.2.0

System environment check

TYPO3 is an enterprise content management system that is powerful, yet easy to install.

After some simple steps you'll be ready to add content to your website. This first step checks your system environment and points out issues.

Fixed. Check again!

I know what I'm doing, continue! ⓘ

Detailed analysis

ⓘ Path /index.php is not a link
The target /index.php should be a link, but is of type file. This can not be fixed. Please investigate.

⚠ No PHP opcode cache loaded
PHP opcode caches hold a compiled version of executed PHP scripts in memory and do not require to recompile any script on each access. This can be a massive performance improvement and can put load off a server in general, a parse time reduction by factor three for full cached pages can be achieved easily if using some opcode cache. If in doubt choosing one, APC runs well and can be used as data cache layer in

Install Tool

Neu-Entwicklung

- Im zweiten Schritt werden die Zugangsdaten zum Datenbankserver angegeben
- Verschiedene Verbindungstypen können ausgewählt werden
 - TCP/IP Verbindung
 - Socket Verbindung
- Alternativen zu MySQL sind ebenfalls möglich

Installing TYPO3 CMS 6.2.0

Connect to your database host

If you have not already created a username and password to access the database, please do so now. This can be done using tools provided by your host.


Username

Password

Type

Socket

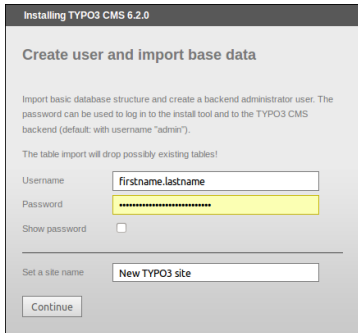
TYPO3 CMS native database implementation is based on mysql. A database abstraction layer allows to run TYPO3 CMS on different database engines like postgres. This is used rather seldom and some core parts and extensions do not fully support this. Your TYPO3 CMS experience might suffer if you choose to install the system on anything different than mysql.



Install Tool

"Neu-Entwicklung"

- Der dritte Schritt erlaubt es, die Datenbank auszuwählen oder zu erstellen (wie bereits in Versionen vor TYPO3 6.2)
- Im vierten Schritt wird das Passwort für den Benutzer "admin" gesetzt (welches ebenso das Install Tool Passwort darstellt) und der Sitename angegeben



The screenshot shows the 'Installing TYPO3 CMS 6.2.0' window with the 'Create user and import base data' step. It includes instructions on importing the database structure and creating an administrator user. The form contains fields for Username (pre-filled with 'firstname.lastname'), Password (masked with asterisks), a 'Show password' checkbox, a 'Set a site name' field (pre-filled with 'New TYPO3 site'), and a 'Continue' button.

Installing TYPO3 CMS 6.2.0

Create user and import base data

Import basic database structure and create a backend administrator user. The password can be used to log in to the install tool and to the TYPO3 CMS backend (default: with username "admin").

The table import will drop possibly existing tables!

Username:

Password:

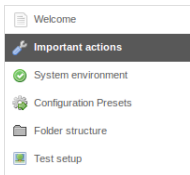
Show password:

Set a site name:

Install Tool

Cache vollständig leeren

- Mit einer neuen Funktion unter "Important actions" kann der gesamte Cache vollständig geleert werden ("Delete all cache")
- Jenes funktioniert auch, wenn der Cache ungültigen PHP Code enthält, der möglicherweise TYPO3 blockiert
- Um eine nicht-funktionierende TYPO3 Instanz zu umgehen, kann das Install Tool direkt aufgerufen werden:
`http://example.com/typo3/install`



Clear all cache

This clear all cache function works similar to the cache clearing in the backend but follows a more straight ahead approach and the according backend hooks are not executed.

This method can throw a fatal error if some broken extension is loaded. If you get a white page or a PHP error message, check your system with the broken extension test below.

Cache vollständig leeren

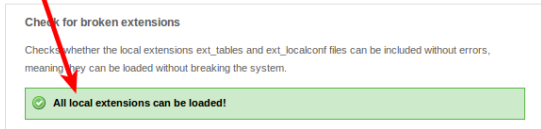
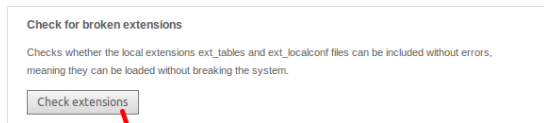
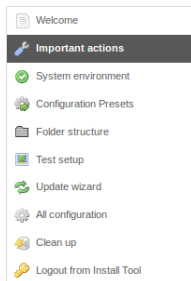
Ablauf, wenn die Funktion "Delete all cache" ausgeführt wird:

1. Inhalt des Verzeichnisses `typo3temp/Cache` wird gelöscht
2. Datenbanktabellen `cf_*` werden geleert
3. Dateien `ext_localconf.php` und `ext_tables.php` von installierten Extensions werden geladen
4. `flushCaches()` wird ausgeführt

Install Tool

Nicht-ausführbare Extensions

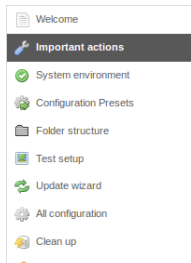
- Mit einer neuen Funktion unter "Important actions" kann nun geprüft werden, ob Extensions geladen werden können, ohne das System lahm zu legen
- Sehr nützlich zum Beispiel bei einem Update von TYPO3 4.5 zu 6.2



Install Tool

Gesalzene Passwörter

- Beim Neuerstellen von Administratorkonten über das Install Tool, werden nun **gesalzene** (salted) Passwörter verwendet (vorausgesetzt, EXT:saltedpasswords ist installiert und konfiguriert)
- Das Install Tool Passwort ist ebenfalls ein **gesalzenes** Passwort (ein existierender MD5 Hash wird beim ersten Login umgewandelt)



Create backend administrator user

You should use this function only if there are no admin users in the database, for instance if this is a blank database. After you've created the user, log in and add the rest of the user information, like email and real name.

Username:

Password:

Password again:

Install Tool

Application Context (1)

- TYPO3 ab Version 6.2 berücksichtigt nun den **Application Context** (bereits bekannt von TYPO3 Flow)
- Umgebungsvariable `TYPO3_CONTEXT` setzt diesen Kontext (Voreinstellung: `Production`, Sub-Context wie z.B. `Production/Staging` möglich)

```
# File: .htaccess
```

```
# Rules to set Application Context based on hostname:
```

```
RewriteCond %{HTTP_HOST} ^dev\.example\.com$
```

```
RewriteRule (.*) $1 [E=TYPO3_CONTEXT:Development]
```

```
RewriteCond %{HTTP_HOST} ^www\.example\.com$
```

```
RewriteRule (.*) $1 [E=TYPO3_CONTEXT:Production]
```

```
# Sets an environment variable, which is then available to TYPO3 CMS:
```

```
SetEnv TYPO3_CONTEXT Production
```

Install Tool

Vorkonfigurierte TYPO3_CONF_VAR Einstellungen

- Vorkonfigurierte TYPO3_CONF_VAR-Einstellungen können im Install Tool ausgewählt werden
- Diverse Parameter steuern zum Beispiel Debug-Ausgaben, das "deprecation log", devIPmask und weitere Log-Einstellungen
- Vorgegebene Kontexte sind "Production" and "Development" (eine eigene Konfiguration ist natürlich möglich)

Development / Production settings

TYPO3 can be run in a specific application context by using one of the built-in contexts "Production" (default), "Development" or "Testing". This can be used to provide specific configuration sets for each context. The context can be defined with the environment variable "TYPO3_CONTEXT" which is usually set through your webserver configuration (e.g. in htaccess).

However, if you don't set a context environment variable, you can still use the install tool to select a configuration preset for "Production" or "Development" context. The goal is to configure a production instance with maximum performance and no debug output that is possibly shown to users, while development instances should enable error output. The configuration preset for "Production" is set by default. As a third alternative, you can enter a custom configuration.

Production (Active)

Production settings turn off debug output, deprecation logs and set logging to warnings and errors only.

Development

Development settings enable debug output, deprecation logs and set logging to info level.

Custom configuration

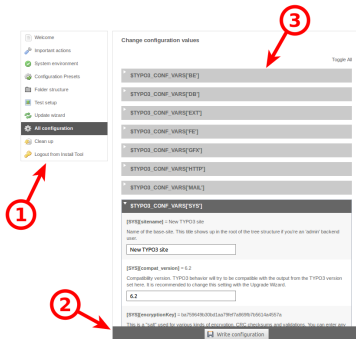
Custom configuration mixture if no other preset fits.

<input type="text"/>	BE/debug
<input type="text"/>	FE/debug
<input type="text"/>	SYS/devIPmask
<input type="text"/>	SYS/displayErrors
<input type="text"/>	SYS/enableDeprecationLog
0	SYS/isqDebug
2	SYS/systemLogLevel

Install Tool

Verbesserte Benutzerfreundlichkeit

- Die Position des Menüs auf der linken Seite ist nun fixiert und das Menü scrollt nicht mit
- Der Button "Write configuration" am Ende der Seite ist fixiert und somit immer sichtbar
- Einträge in "All Configuration" sind gruppiert (klappen bei Klick auf die Überschrift auf)



Install Tool

Benutzerfreundliche Fehlercodes

- Aussagekräftige (menschens-lesbare) Begriffe können für folgende Optionen verwendet werden:

[SYS] [errorHandlerErrors]

[SYS] [exceptionalErrors]

[SYS] [syslogErrorReporting]

[SYS] [belogErrorReporting]

(vor TYPO3 6.2 waren nur numerische Werte möglich)

- Ein Extbase ViewHelper **format.phpErrorCode** übernimmt die Umwandlung in PHP Fehlercodes

```
[SYS][errorHandlerErrors] = 30466
```

Integer: The E_* constant that will be handled by the errorhandler. Not all PHP error types can be handled! Default is E_ALL & ~(E_STRICT | E_NOTICE | E_COMPILE_WARNING | E_COMPILE_ERROR | E_CORE_WARNING | E_CORE_ERROR | E_PARSE | E_ERROR).

```
30466
```

Current PHP error code **30466** represents:

E_WARNING | E_USER_ERROR | E_USER_WARNING | E_USER_NOTICE |
E_RECOVERABLE_ERROR | E_DEPRECATED | E_USER_DEPRECATED

Install Tool

Verzeichnis-Fehler

- Fehler in der Verzeichnisstruktur werden direkt angezeigt (als "Badge", eingekreiste Zahl)

- Welcome
- Important actions
- System environment
- Configuration Presets
- Folder structure** 3
- Test setup
- Update wizard
- All configuration
- Clean up
- Logout from Install Tool

These files or folders have errors and may be automatically fixable:

❗ Path /index.php is not a link
The target /index.php should be a link, but is of type file. This can not be fixed. Please investigate.

❗ Path /typo3 is not a link
The target /typo3 should be a link, but is of type dir. This can not be fixed. Please investigate.

❗ /typo3_src should be a link, but it does not exist
Links can not be fixed by this system

⚠ /typo3temp has wrong permission
Target permission are 2770 but current permission are 0777

Install Tool

TYPO3 Core Updates

- Minor-Core-Updates (inkl. Security-Versionen) können direkt aus dem Install Tool ausgeführt werden
- Die Umgebungsvariable `TYPO3_DISABLE_CORE_UPDATER=1` unterbindet diese Funktion

The screenshot illustrates the update notification process in the TYPO3 Install Tool. It shows a sequence of three overlapping panels:

- Panel 1 (Top):** Titled "Core update", it contains the text: "The install tool can automatically update the TYPO3 CMS core to its latest minor release if certain criteria are met." Below this text is a button labeled "Check for core updates".
- Panel 2 (Middle):** Titled "Fetching list of released versions from typo3.org", it features a loading spinner icon on the left.
- Panel 3 (Bottom):** This panel displays the results of the update check. It starts with a green success message: "Fetched list of released versions". Below that is a blue informational message: "Update to release 6.2.1 is available!". At the bottom of this panel is a button labeled "Update now".

Install Tool

Diverses

- Sämtliche Formulare sind nun CSRF-geschützt
(*cross-site request forgery*)
- Das Install Tool nutzt einen vereinfachten Fluid Standalone View
- Nur unbedingt notwendige TYPO3 Funktionen werden geladen
(Extensions mit beschädigten `ext_localconf.php` oder `ext_tables.php` Dateien führen nicht mehr zum Abbruch des Install Tools)
- Neuer Startpunkt: `typo3/sysext/install/Start/Install.php`
War bisher: `typo3/install/index.php`
(es existiert eine Umleitung von der alten zur neuen Adresse)
- Damit das Install Tool auch benutzbar bleibt, wenn der Cache ungültigen PHP Code enthält, verzichtet es komplett auf Caching

Install Tool

Diverses

- Es wird geprüft, ob die PHP Option `xdebug.max_nesting_level` einen Wert von mindestens 250 aufweist (die Voreinstellung "100" führt unter Umständen zu Fehlern)
- "Relaxed permission check":
Während normalerweise die Berechtigung für das Root-Verzeichnis der Installation 2770 sein muss und der Ordner dem Web-User gehören muss, um TYPO3 zu installieren, wurde nun eine Option "targetPermissionRelaxed" eingeführt, bei der dieser Check für den Root-Folder außer Kraft gesetzt wurde, sofern es trotzdem möglich ist, die benötigten Unterverzeichnisse anzulegen.

Install Tool

Diverses

- Folgende veraltete Optionen wurden vom Install Tool entfernt (und damit auch aus der Datei `LocalConfiguration.php`):

BE/loginLabels

BE/loginNews

BE/useOnContextMenuHandler

EXT/em_mirrorListURL

EXT/em_wsdlURL

EXT/extList

EXT/extList_FE

EXT/noEdit

FE/defaultTypoScript_editorcfg

FE/simulateStaticDocuments

GFX/noIconProc

GFX/TTFLocaleConv

SYS/additionalAllowedClassPrefixes

SYS/caching/cacheBackends

SYS/caching/cacheFrontends

SYS/extCache

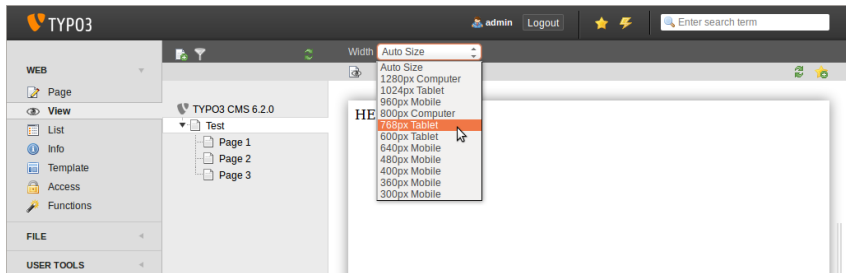
SYS/T3instID

Kapitel 2: Responsive Images

Responsive Images

Bildschirmgrößen in der Seitenvorschau

- Redakteure können nun verschiedene Bildschirmgrößen (Breiten) für die Vorschau auswählen, um responsive Websites testen zu können



Responsive Images

Bildschirmgrößen in der Seitenvorschau

- Bildschirmgrößen sind via PageTSconfig konfigurierbar:

```
mod.web_view.previewFrameWidths {  
    1780.label = <any LLL or string>  
    1780.height = 145  
}
```

- Der Schlüssel (hier: 1780) gibt die Breite an, die Höhe (height) ist optional

- Vorgegebene Größen sind in der folgenden Datei definiert:

```
typo3/sysexst/core/Configuration/DefaultConfiguration.php
```

- Labels können via PageTSconfig konfiguriert werden:

```
mod.web_view.previewFrameWidths {  
    1280.label = LLL:EXT:viewpage/Resources/Private/Language/locallang.xlf:computer  
    1024.label = LLL:EXT:viewpage/Resources/Private/Language/locallang.xlf:tablet  
}
```

Responsive Images

Responsive Bildergalerien

- Es wurden zusätzliche Attribute für responsive Bildergalerien eingeführt
- Dafür wurde das Rendering von "CSS styled content" erweitert
- Beispiel: HTML5 (`config.doctype = html5` vorausgesetzt)

TYPO3 CMS vor 6.2:

```
<div class="csc-textpic-imagewrap">...</div>
```

TYPO3 CMS ab 6.2:

```
<div class="csc-textpic-imagewrap"  
  data-csc-images="{register:imageCount}"  
  data-csc-cols="{field:imagecols}">...</div>
```

Responsive Images

Responsive Image Rendering

- cObject IMAGE kann nun so genannte "sourceCollections" rendern, um damit verschiedene Display-Auflösungen und Bildschirmgrößen zu unterstützen
- Zum Aktivieren des responsive Image Rendering für die cObjects "Bild" und "Text/Bild" müssen folgende Einstellungen gemacht werden:
`styles.content.imgtext.responsive`
`styles.content.imgtext.layoutKey`
- Gültige ("out of the box") Optionen sind:
 - `default:` standard ``-tag
 - `srcset:` ``-tag mit alternativer Bildquelle als `srcset`-Attribut
 - `picture:` `<picture>`-tag mit `source-child-tags`
 - `data:` ``-tag mit alternativer Bildquelle als `data`-Attribut

Responsive Images

Eigenschaft: `layoutKey`

- `layoutKey` definiert das Render-Layout (HTML Code, der für das ``-tag verwendet wird)
- Jede Möglichkeit repräsentiert eine unterschiedliche Lösung um HTML-Code für das IMAGE zu rendern
- Option `default` rendert das ``-tag auf herkömmliche Weise (jenes empfiehlt sich, wenn das Frontend nicht "responsive" ist)
- Wenn man ein responsive Layout implementiert, benötigt man unterschiedliche Bildgrößen
- Abhängig vom HTML Framework, dem Browser und der JavaScript Bibliothek für das progressive enhancement, verwendet man:
 - entweder eines der vordefinierten Layouts,
 - oder definiert ein neues Layout mit einem neuen `layoutKey`

Responsive Images

Eigenschaft: layoutKey

```
layoutKey = {$styles.content.imgtext.layoutKey}
layout {
  default {
    element = 
  }
  srcset {
    element = 
    source = |*|###SRC### ###SRCSETCANDIDATE###,|*|###SRC### ###SRCSETCANDIDATE###
  }
  picture {
    element = <picture>###SOURCECOLLECTION###</picture>
    source = <source src="###SRC###" media="###MEDIAQUERY###"###SELCLOSINGTAGSLASH###>
  }
  data {
    element = 
    source = data-###DATAKEY###="###SRC###"
  }
}
```

Responsive Images

Eigenschaft: `layout.[layoutKey].element`

- `###SRC###`

URL für Attribut: `src`

- `###WIDTH###`

Bildbreite (in Pixel) für Attribut: `width`

- `###HEIGHT###`

Bildhöhe (in Pixel) für Attribut: `height`

- `###PARAMS###`

Zusätzliche Parameter, wie im cObject IMAGE definiert

- `###ALTPARAMS###`

Zusätzliche alternative Parameter, wie im cObject IMAGE definiert

Responsive Images

Eigenschaft: `layout.[layoutKey].element`

- **###BORDER###**

Rahmenbreite (in Pixel) für Attribut: `border`

- **###SELCLOSINGTAGSLASH###**

Schließendes Tag, z.B. `` oder ``
(abhängig von `config.xhtmlDoctype` oder `config.doctype`)

- **###SOURCECOLLECTION###**

Zusätzliche Quellen des Bildes, abhängig von der unterschiedlichen Verwendung im Responsive Web Design.

Die Werte werden im Schlüssel `layout.[layoutKey].source` definiert

Responsive Images

Eigenschaft: `sourceCollection.[dataKey]`

- Standard `sourceCollection` von `EXT:css_styled_content`
- Es ist zu empfehlen, eine eigene `sourceCollection` zu erstellen

```
sourceCollection {
  small {
    width = 200
    srcsetCandidate = 600w
    mediaQuery = (max-device-width: 600px)
    dataKey = small
  }
  smallRetina {
    if.directReturn = 1
    width = 200
    pixelDensity = 2
    srcsetCandidate = 600w 2x
    mediaQuery = (max-device-width: 600px) AND (min-resolution: 192dpi)
    dataKey = smallRetina
  }
}
```

Responsive Images

Weitere Informationen

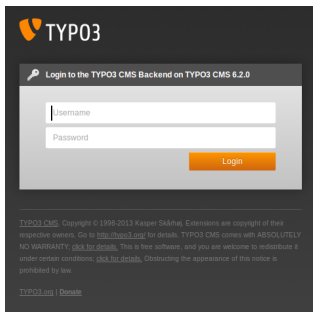
- Lauffähiges Code-Beispiel:
http://wiki.typo3.org/Responsive_Image_Rendering
- Artikel von Sven Wolfermann auf typo3.org:
<http://typo3.org/news/article/responsive-image-rendering-in-typo3-cms-62/>
- W3C Spezifikation:
<http://www.w3.org/html/wg/drafts/srcset/w3c-srcset/>
<http://www.w3.org/TR/html-picture-element/>
- Arbeitsentwurf (working draft) der "Responsive Image Community Group":
<http://responsiveimages.org>

Kapitel 3: Änderungen im Backend

Änderungen im Backend

Anmeldung am Backend

- Autofokus auf das Eingabefeld des Benutzernamens bei Aufruf des Anmeldeformulars
(HTML5 Attribute: `autofocus="autofocus"`)



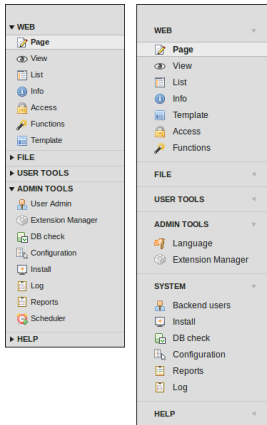
Änderungen im Backend

Allgemeines Erscheinungsbild

- Verbesserte Usability durch aufgelockertes Erscheinungsbild
- Horizontale und vertikale Abstände vieler Elemente wurden vergrößert
- Basis ist ein 12 Pixel Raster, welches verdoppelt wurde

Links: TYPO3 4.5

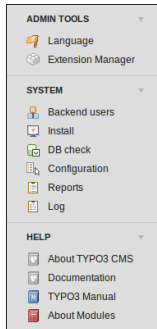
Rechts: TYPO3 6.2



Änderungen im Backend

Allgemeines Erscheinungsbild

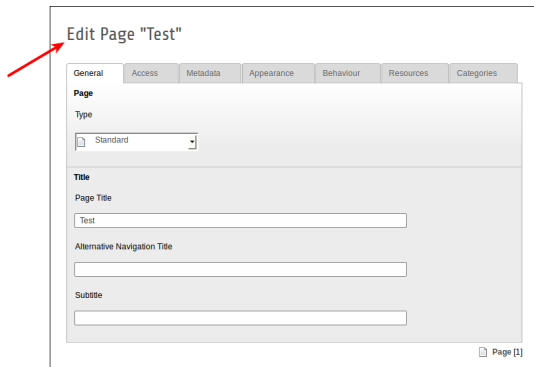
- Modul in der linken Spalte wurden neu angeordnet
- Modul "ADMINWERKZEUGE" wurde zweigeteilt:
 - **ADMINWERKZEUGE** ("Sprache" und "Erweiterungsmanager")
 - **SYSTEM** (low-level Tools, die keinen Seitenbaum anzeigen)
- Das Modul "TypoScript-Hilfe" wurde entfernt (veraltet)



Änderungen im Backend

Allgemeines Erscheinungsbild

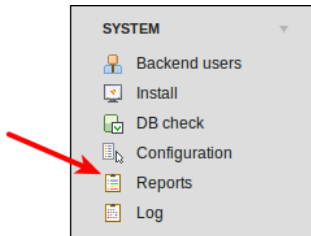
- `<h1>`-Überschriften im rechten Bereich werden nun durchgehend im TYPO3 Font dargestellt



Änderungen im Backend

Allgemeines Erscheinungsbild

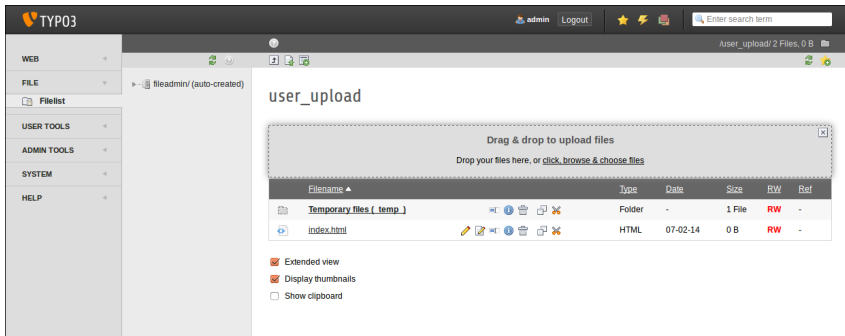
- Modul "Reports" hat ein eigenes Symbol erhalten



Änderungen im Backend

Drag&Drop in der Dateiliste

- In der Dateiliste können nun Dateien per HTML5 Drag&Drop hochgeladen werden



The screenshot shows the TYPO3 Filelist interface. The left sidebar contains navigation menus for WEB, FILE, USER TOOLS, ADMIN TOOLS, SYSTEM, and HELP. The main content area displays the 'user_upload' directory. A dashed box highlights a drag-and-drop upload area with the text: 'Drag & drop to upload files' and 'Drop your files here, or [click, browse & choose files](#)'. Below this is a table listing files and folders.

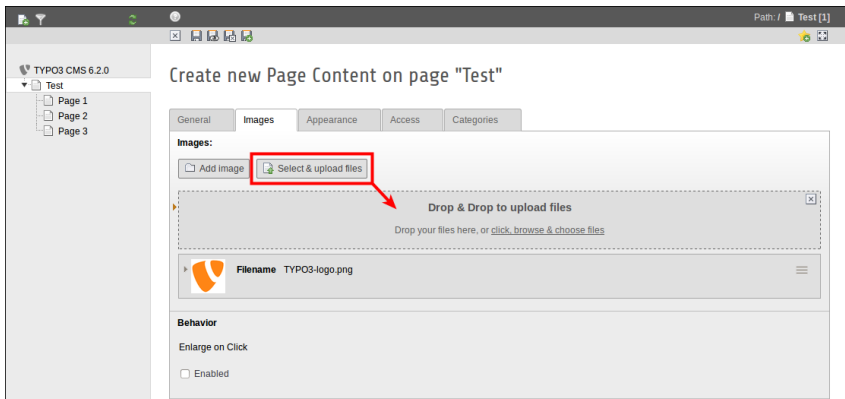
Filename	Type	Date	Size	RW	Ref
Temporary files (temp)	Folder	-	1 File	RW	-
index.html	HTML	07-02-14	0 B	RW	-

Below the table, there are three checkboxes: Extended view, Display thumbnails, and Show clipboard.

Änderungen im Backend

Drag&Drop in der Dateiliste

- ...sowie über Inhaltselemente (Button: "Select & upload files")






















The screenshot displays the TYPO3 CMS 6.2.0 backend interface. On the left, a sidebar shows the site structure with a 'Test' folder containing 'Page 1', 'Page 2', and 'Page 3'. The main content area is titled 'Create new Page Content on page "Test"'. It features several tabs: 'General', 'Images', 'Appearance', 'Access', and 'Categories'. The 'Images' tab is active, showing an 'Images:' section with two buttons: 'Add image' and 'Select & upload files'. The 'Select & upload files' button is highlighted with a red rectangular box, and a red arrow points from it to a dashed-line box labeled 'Drop & Drop to upload files'. Below this box, there is a text prompt: 'Drop your files here, or [click, browse & choose files](#)'. Underneath, a file list shows a file named 'TYPO3-logo.png' with a preview of the logo and a menu icon. At the bottom, the 'Behavior' section includes the option 'Enlarge on Click' with a checkbox labeled 'Enabled'.

Änderungen im Backend

Backend Benutzerverwaltung

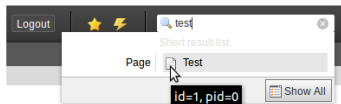
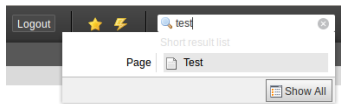
- Neben dem Benutzernamen wird nun auch der Realname dargestellt
- Klick auf den Namen führt zum Bearbeitungsformular
- Zudem gibt es nun einen Delete-Button in der Liste

Username / Real Name		Last login	
 administrator Firstname Lastname	 Compare	  	22-02-14 03:56
 ajolie Angelina Jolie	 Compare	     	Never
 bpitt Brad Pitt	 Compare	     	Never
3 Users			

Änderungen im Backend

Verbesserte Suchfunktion

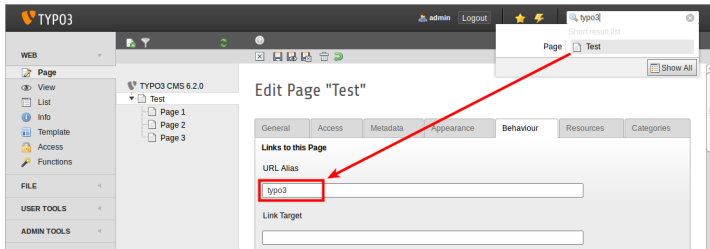
- Die Live-Suche zeigt nun UID und PID als Tooltip in der Ergebnisliste an
- Sobald man nach einer Suche das Editierformular wieder schließt, gelangt man in die Listen-Ansicht der PID (und nicht wie früher auf eine leere Seite)



Änderungen im Backend

Live Search


- In TYPO3 vor Version 6.2 durchsucht die Live-Suche für Seiten nur die Felder `title` und `uid`
- Ab TYPO3 Version 6.2 kann nun das Feld `alias` zusätzlich in der Suche einbezogen werden, sofern per `UserTSconfig` konfiguriert (`options.pageTree.searchInAlias = 1`)



Änderungen im Backend

File Abstraction Layer


- Dateiname und Titel werden nun im FAL-Element "Header" angegeben



General Images Appearance Access Categories

Images:

Add Image Select & upload files

	Title TYPO3 Logo Filename TYPO3_icon_200x200_white.png	  
---	---	---

Behavior

Enlarge on Click

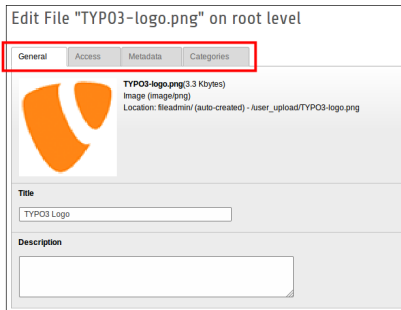
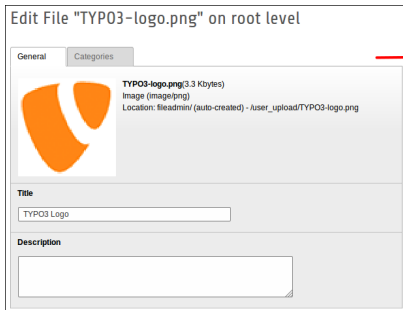
Enabled

Page Content [8]

Änderungen im Backend

File Abstraction Layer (EXT:filemetadata)

- Durch die EXT:filemetadata werden zusätzliche Metadaten eingebracht (Extension wird mitgeliefert, ist aber standardmäßig nicht installiert)



Änderungen im Backend

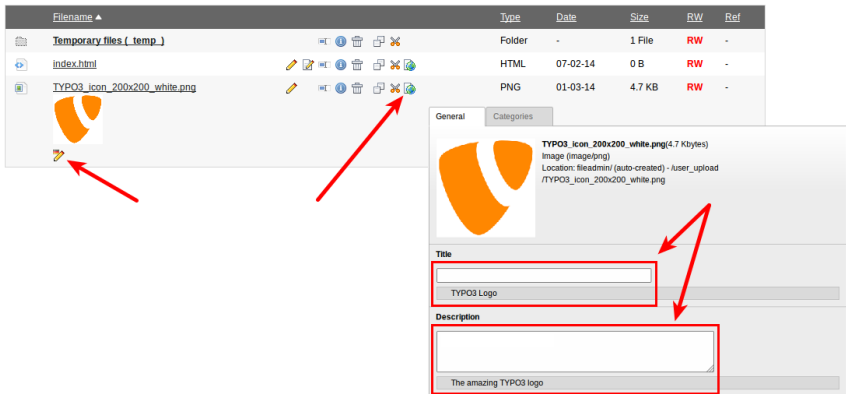
File Abstraction Layer (EXT:filemetadata)

General	Access	Metadata	Categories
Creator			
<input type="text"/>			
Creator Tool	Publisher	Source	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Geo Location			
Country	Region	City	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Latitude	Longitude		
<input type="text" value="0.0000000000000000"/>	<input type="text" value="0.0000000000000000"/>		
Metrics			
Width	Height	Unit	Color Space
<input type="text" value="68"/>	<input type="text" value="68"/>	<input type="text"/>	<input type="text"/>

Änderungen im Backend

File Abstraction Layer

- FAL Metadaten können nun in Frontend-Sprachen übersetzt werden



The screenshot displays the TYPO3 File Manager interface. On the left, a file list shows three items: a folder named 'Temporary files (.temp_)', an HTML file 'index.html', and a PNG image 'TYPO3_icon_200x200_white.png'. The image file is selected, and its metadata editor is open on the right. The metadata editor shows the file's title as 'TYPO3 Logo' and its description as 'The amazing TYPO3 logo'. Red arrows highlight the edit icon for the image file in the list and the corresponding edit icon in the metadata editor. Another red arrow points from the metadata editor back to the file list, indicating the flow of updates.

Filename	Type	Date	Size	RW	Ref
Temporary files (.temp_)	Folder	-	1 File	RW	-
index.html	HTML	07-02-14	0 B	RW	-
TYPO3_icon_200x200_white.png	PNG	01-03-14	4.7 KB	RW	-

General | Categories

TYPO3_icon_200x200_white.png (4.7 Kbytes)
Image (image/png)
Location: fileadmin/ (auto-created) - User_upload
/TYPO3_icon_200x200_white.png

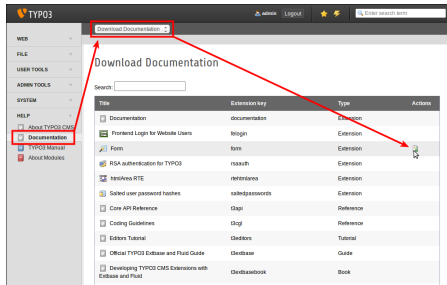
Title
TYPO3 Logo

Description
The amazing TYPO3 logo

Änderungen im Backend

Modul: Documentation

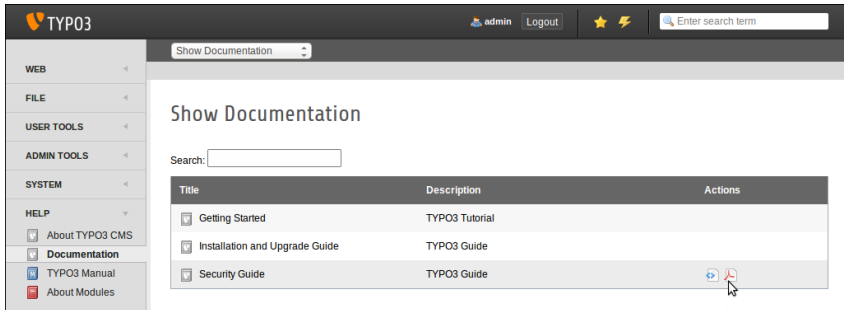
- Neues Modul "Documentation" ermöglicht es, Dokumentationen herunterzuladen
- Das Modul wird über den Erweiterungsmanager hinzugefügt, bzw. ist bei neuen Installationen bereits standardmäßig geladen
- Funktion "Download Documentation" lädt Dokumentationen nach





Änderungen im Backend

Modul: Documentation

- Im Modul "Show Documentation" kann man die Dokumentationen dann ansehen



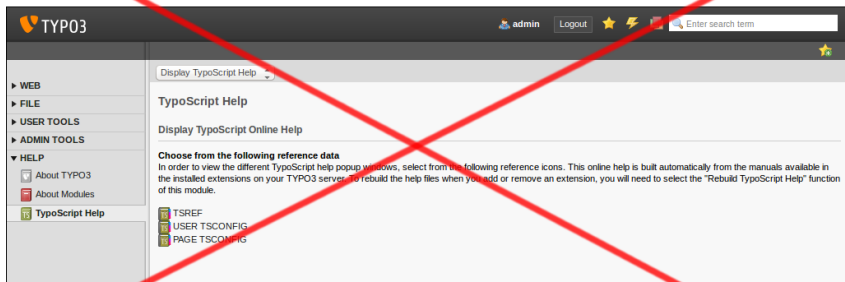
The screenshot shows the TYPO3 Backend interface for the 'Show Documentation' module. The top navigation bar includes the TYPO3 logo, the user 'admin', a 'Logout' button, a star icon, a lightning bolt icon, and a search input field with the placeholder text 'Enter search term'. The left sidebar contains a menu with categories: WEB, FILE, USER TOOLS, ADMIN TOOLS, SYSTEM, and HELP. Under the HELP category, there are links for 'About TYPO3 CMS', 'Documentation', 'TYPO3 Manual', and 'About Modules'. The main content area is titled 'Show Documentation' and features a search input field. Below the search field is a table with three columns: 'Title', 'Description', and 'Actions'. The table contains three rows of documentation entries.

Title	Description	Actions
Getting Started	TYPO3 Tutorial	
Installation and Upgrade Guide	TYPO3 Guide	
Security Guide	TYPO3 Guide	 

Änderungen im Backend

Removed: TypoScript Help

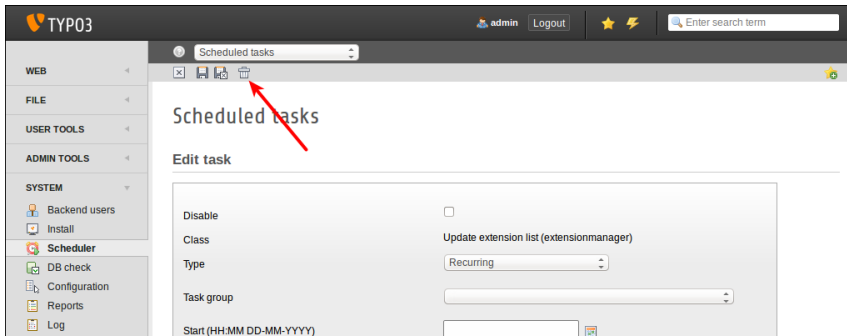
- EXT:tsconfig_help ("TypoScript-Hilfe") entfernt (enthielt veraltete Information und wurde seit längerem nicht gepflegt)



Änderungen im Backend

Planer ("Scheduler")

- Löschen von Scheduler-Tasks in der Bearbeitungsansicht
(in TYPO3 vor Version 6.2 können Tasks nur in der Listenansicht gelöscht werden)



The screenshot shows the TYPO3 Backend interface. On the left is a navigation menu with categories: WEB, FILE, USER TOOLS, ADMIN TOOLS, and SYSTEM. Under SYSTEM, the 'Scheduler' option is selected. The main content area is titled 'Scheduled tasks' and shows an 'Edit task' form. At the top of the edit form, there are four icons: a close button (X), a save button (floppy disk), a refresh button (circular arrow), and a delete button (trash can). A red arrow points to the delete icon. The form fields include: 'Disable' (checkbox), 'Class' (text input), 'Type' (dropdown menu set to 'Recurring'), 'Task group' (dropdown menu), and 'Start (HH:MM DD-MM-YYYY)' (text input).

Änderungen im Backend

Planer ("Scheduler")

- Scheduler-Tasks können nun mit einer Beschreibung versehen werden (als Subheadline oder Tooltip, siehe nächste Seite)

Edit task

Disable	<input type="checkbox"/>
Class	Update extension list (extensionmanager)
Type	Recurring
Task group	
Start (HH:MM DD-MM-YYYY)	<input type="text"/>
End (HH:MM DD-MM-YYYY)	<input type="text"/>
Frequency (seconds or cron command)	86400
Allow Parallel Execution	<input type="checkbox"/>
Description	This task updates the extension list once a day

Änderungen im Backend

Planer ("Scheduler")

- Taskbeschreibung als Subheadline
(jenes muss in der Extensionkonfiguration aktiviert werden)

	ID	Task	Type	Frequency	Parallel Execution	Last Execution	Next Execution
<input type="checkbox"/>	1	Update extension list (extensionmanager) This task updates the extension list once a day	Recurring	86400	No	-	24-02-14 00:00

Execute selected tasks

- Taskbeschreibung als Tooltip ("Hover")

	ID	Task	Type	Frequency	Parallel Execution	Last Execution	Next Execution
<input type="checkbox"/>	1	Update extension list (extensionmanager)	Recurring	86400	No	-	24-02-14 00:00

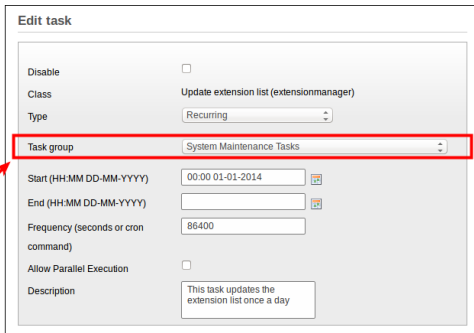
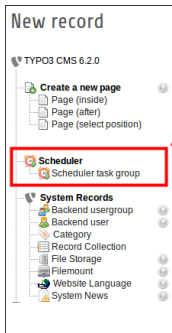
Execute selected tasks

This task updates the extension list once a day

Änderungen im Backend

Planer ("Scheduler")

- Scheduler-Tasks können nun gruppiert werden
- Dazu legt man auf der Root-Page (UID: 0) "Scheduler task groups" an und wählt im Task die entsprechende Gruppe aus



Änderungen im Backend

Systemextension: Form

- Neuer Post-Prozessor für das cObject FORM: **redirect** (Weiterleitung nach Abschicken der Mail)
- Das Feld wird über die TypoScript-Funktion `typolink` ausgewertet, das bedeutet, man kann z.B. eine ID für eine interne Seite oder eine URL angeben

The screenshot shows the configuration interface for a Form in the TYPO3 backend. It features three tabs: 'Elements', 'Options', and 'Form'. The 'Form' tab is active, displaying a tree view of configuration sections: 'BEHAVIOUR', 'PREFIX', 'ATTRIBUTES PROPERTIES', and 'POST PROCESSORS'. The 'POST PROCESSORS' section is expanded, showing a dropdown menu with 'Redirect' selected. A red arrow points to this dropdown. Below this, the 'Send email' section is visible, with fields for 'Email address of the recipient' (recipient@test.com), 'Email address of the sender' (sender@test.com), and 'Subject' (test subject), each with a green checkmark. A 'Remove' button is present. At the bottom, the 'Redirect' section is expanded, showing a 'Destination to' field with the value '123' and a 'redirect to:' label. A red arrow points to this section, and a 'Remove' button is located below it.

Änderungen im Backend

Modul: List

- Zusätzliche Spalten "UID" und "PID" in der Listenansicht für nicht-Administratoren

The screenshot displays the 'Template (1)' list view in the TYPO3 Backend. The table header includes columns for '[Ref]', '[uid]', and '[pid]'. The data row shows 'NEW SITE' with values '1' and '1' in the '[uid]' and '[pid]' columns, respectively. A red box highlights these values, with a red arrow pointing to the field selection menu below. In this menu, the fields '[uid]' and '[pid]' are highlighted, with another red arrow pointing to them.

Template Title	[Ref]	[uid]	[pid]
NEW SITE	-	1	1

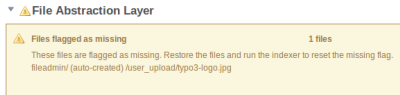
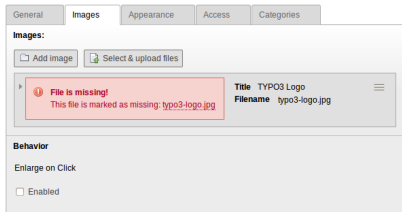
Field selection menu:

- Include Static Templates After Basis Templates
- Setup
- Description
- Static Template Files from TYPO3 Extensions
- Versioning Label
- [uid]
- [pid]
- [tstamp]
- [crdate]
- [cruser_id]

Änderungen im Backend

File Abstraction Layer

- Bemerkt der FAL-Indexer eine nicht mehr vorhandene Datei, wird ein entsprechender Hinweis ausgegeben und ein Flag "missing" in der Datenbanktabelle "sys_file" gesetzt
- Im "Reports" Modul wird jenes ebenfalls als Warnung angezeigt
- Sobald die Datei wieder gefunden wird, werden Hinweis und Flag zurückgesetzt



Änderungen im Backend

Neuer Menütyp: Kategorien (1)

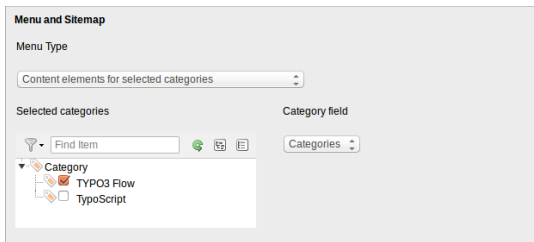
- Im Inhaltselement "Menü und Sitemap" kann ein Menü, basierend auf Kategorien ausgewählt werden

The screenshot displays the TYPO3 Backend interface. On the left, the 'Special elements' tab is active, showing a list of content elements. The 'Special Menu' element is highlighted with a red rectangular box. A red arrow points from this box to the 'Menu and Sitemap' configuration dialog on the right. The dialog shows the 'Menu Type' dropdown set to 'Pages for selected categories'. Below, the 'Selected categories' section includes a search field 'Find Item' and a list of categories: 'Category' (expanded), 'TYPO3 Flow' (checked), and 'TypoScript' (unchecked). A 'Category field' dropdown is also visible on the right side of the dialog.

Änderungen im Backend

Neuer Menütyp: Kategorien (1)

- Weiterer neuer Menütyp: "Content elements for selected categories"



Änderungen im Backend

Kategorien

- Kategorien können nun manuell sortiert werden (bisher waren diese immer alphabetisch sortiert)

▼ Category (2)	
Title	[Ref]
TYPO3 Flow	1
TypoScript	-



Änderungen im Backend

Kategorien

- Bei Backend-Benutzern und -Gruppen kann nun definiert werden, welche Kategorien diese sehen können

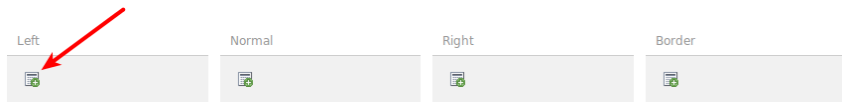
The screenshot displays the 'Mounts and Workspaces' configuration page in the TYPO3 Backend. At the top, there are tabs for 'General', 'Access Rights', 'Mounts and Workspaces', 'Options', and 'Access'. The 'Workspace permissions' section is active, showing a checked checkbox for 'Edit Live (Online)'. Below this, a vertical ellipsis indicates a scrollable list. The 'Category Mounts' section is visible, showing a tree view where the 'Category' folder is expanded, revealing sub-items 'TYPO3 Flow' and 'TypoScript'. In the bottom right corner, the user 'Backend user [2]' is identified.

Änderungen im Backend

Usability

- Das Icon "Neues Inhaltselement" ist nun immer sichtbar, sofern die Spalte noch leer ist

Test Page



Änderungen im Backend

Funktionen

- Wenn im Modul "Funktionen" mehrere Seiten angelegt werden, kann ausgewählt werden, dass diese vorerst nicht im Menü sichtbar sein sollen

Create multiple pages

Create new pages:

Page 1:	<input type="text"/>	Type:	<input type="text" value="Standard"/>
Page 2:	<input type="text"/>	Type:	<input type="text" value="Standard"/>
Page 3:	<input type="text"/>	Type:	<input type="text" value="Standard"/>

Place new pages after the existing subpages

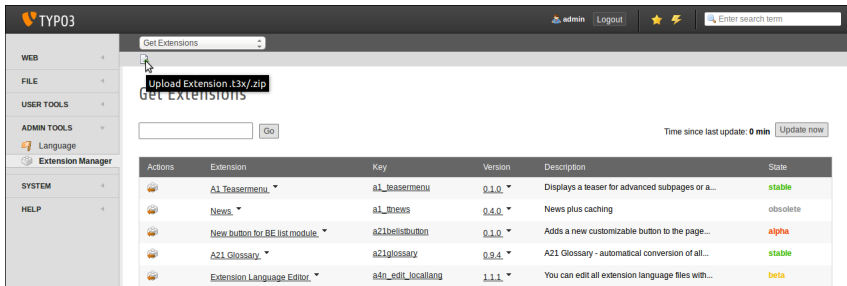
Hide new pages

Hide new pages in menus

Änderungen im Backend

Erweiterungsmanager

- Extensions können nun auch im Untermenü "Erweiterungen hinzufügen" hochgeladen werden



The screenshot shows the TYPO3 Backend interface. The top navigation bar includes the TYPO3 logo, user information (admin), a Logout button, and a search bar. The left sidebar contains a menu with categories like WEB, FILE, USER TOOLS, ADMIN TOOLS, SYSTEM, and HELP. The 'Extension Manager' is selected in the sidebar. The main content area shows the 'Get Extensions' dropdown menu, which is open, revealing the 'Upload Extension .t3x/.zip' option. Below the dropdown is a search input field with a 'Go' button and a 'Time since last update: 0 min' indicator with an 'Update now' button. A table of installed extensions is visible below the search area.

Actions	Extension	Key	Version	Description	State
	A1 Teasermenu	a1_teasermenu	0.1.0	Displays a teaser for advanced subpages or a...	stable
	News	a1_tnews	0.4.0	News plus caching	obsolete
	New button for BE list module	a21bellsbutton	0.1.0	Adds a new customizable button to the page...	alpha
	A21 Glossary	a21glossary	0.9.4	A21 Glossary - automatical conversion of all...	stable
	Extension Language Editor	a4n_edit_loclang	1.1.1	You can edit all extension language files with...	beta

Änderungen im Backend

Recycler

- Einträge im Recycler lassen sich nun nach dem letzten Zugriff sortieren, um das Auffinden von Einträgen zu erleichtern

Recycler

The recycler allows you to select any deleted data and undelete it. You can undelete recursive if the parent of the element is deleted too.

Search: Depth: Infinite Table: Page Content

<input type="checkbox"/>	UID	PID	Records	Table	Last edit
<input type="checkbox"/>	▶ 5	1	test content element 1	Page Content	23-02-1
<input type="checkbox"/>	▶ 6	1	test content element 2	Page Content	23-02-1
<input type="checkbox"/>	▶ 7	1	test content element 3	Page Content	23-02-1

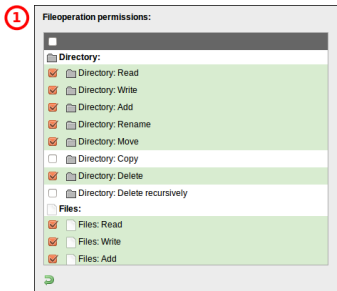
Sort Ascending
Sort Descending
Columns

Page 1 of 1 | Displaying records 1 - 3 of 3 | Delete | Undelete

Änderungen im Backend

Zugriffsrechte auf Dateien und Ordner

- Rechte auf Dateien/Ordner können nun für Benutzer/Gruppen sehr detailliert konfiguriert werden (1)
- Jenes ist zwar bereits seit TYPO3 6.0 möglich, bisher allerdings nur mittels UserTSconfig (2)



Änderungen im Backend

OpenID (1)

- Die OpenID Konfiguration für Backend-Benutzer ist nun über einen Wizard wesentlich einfacher zu bewerkstelligen
- Hierfür ist natürlich die Systemextension EXT:openid erforderlich

Edit Backend user "brad.pitt" on root level

General | Access Rights | Mounts and Workspaces | Options

Disable:

Username:

OpenID identifier:

Password:

OpenID registration

Add an OpenID to your backend user. This OpenID can then be used to log in the TYPO3 backend.

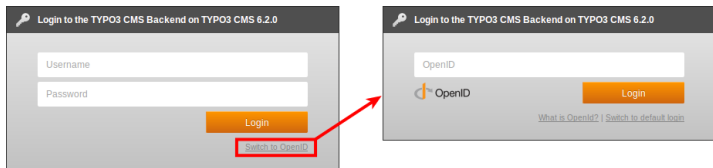
OpenID Identifier



Änderungen im Backend

OpenID (2)

- Die OpenID Konfiguration für Backend-Benutzer ist nun über einen Wizard wesentlich einfacher zu bewerkstelligen
- Hierfür ist natürlich die Systemextension EXT:openid erforderlich

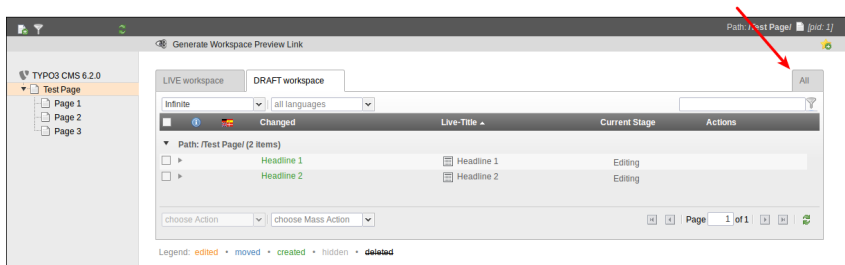


- Weitere Informationen über OpenID:
<http://openid.net>

Änderungen im Backend

Arbeitsumgebung

- Benutzer können nun selbst entscheiden, an wen die Benachrichtigungen gesendet werden sollen
- Der Reiter "All" ist nun für alle Benutzer (nicht nur für Admins) sichtbar



Legend: edited • moved • created • hidden • deleted

Kapitel 4: TScnfig & TypoScript

Tsconfig & TypoScript

TypoScript inkludieren

- Sämtliche Dateien eines Verzeichnisses (rekursiv) inkludieren:

```
<INCLUDE_TYPOSCRIPT: source="DIR:directory">
```

```
<INCLUDE_TYPOSCRIPT: source="DIR:EXT:myextension/res/setup">
```

- Reihenfolge der Inkludierung:

alphabetisch, zuerst Dateien, dann Verzeichnisse

- Mit der Option `extensions="..."` werden nur Dateien mit dieser Endung berücksichtigt

```
<INCLUDE_TYPOSCRIPT: source="DIR:directory" extensions="ts">
```

- Standardmäßig werden nur `ts-`, `t3-`, `t3s-`, `t3c-`, `txt-` Dateien inkludiert

- Diese Liste ist im Install Tool konfigurierbar:

```
$TYPO3_CONF_VARS['SYS']['tsfile_ext']
```

TSconfig & TypeScript

TypeScript inkludieren

- INCLUDE_TYPOSCRIPT unterstützt nun auch relative Pfade
- Pfad der ersten Inkludierung **muss** absolut sein
- ./ verweist auf das aktuelle Verzeichnis der letzten Inkludierung
- ../ verweist auf das Elternverzeichnis des letzten Inkludierung
- Zum Beispiel:

```
<INCLUDE_TYPOSCRIPT: source="FILE:directory/typoscript/setup.ts">
```

```
<INCLUDE_TYPOSCRIPT: source="FILE:./filename.ts">
```

```
<INCLUDE_TYPOSCRIPT: source="FILE:../filename.ts">
```

```
<INCLUDE_TYPOSCRIPT: source="FILE:../directory/filename.ts">
```

TSconfig & TypeScript

strPad

- Option `stdWrap` wurde zu `strPad` Eigenschaften hinzugefügt

```
page = PAGE
page.10 = TEXT
page.10 {
  value = Hello World!
  strPad {
    length = 5
    length {
      current = 1
      setCurrent.data = TSFE:page|uid
      setCurrent.wrap = | + 80
      prioriCalc = 1
    }
    padWith = .
  }
}
```

TSconfig & TypoScript

`_DEFAULT_PI_VARS`

- `stdWrap` wurde zu `_DEFAULT_PI_VARS` hinzugefügt
- `_DEFAULT_PI_VARS` dienen dazu, Default-Werte für piVars (zu einer Extension gehörende GET/POST-Variablen) zu setzen
- TYPO3 vor Version 6.2:

```
plugin.tt_news._DEFAULT_PI_VARS {  
    year = 2013  
}
```

- TYPO3 ab Version 6.2:

```
plugin.tt_news._DEFAULT_PI_VARS {  
    year.stdWrap.data = date:Y  
}
```

TScnfig & TypoScript

Debug Ausgaben

- Debug Register und Page per TypoScript ausgeben:

```
GLOBALS['TSFE']->register
```

```
GLOBALS['TSFE']->page
```

- Zum Beispiel:

```
10 = LOAD_REGISTER
```

```
10.variable = value
```

```
20 = TEXT
```

```
20.data = debug:register
```

```
30 = TEXT
```

```
30.data = debug:page
```

SYS_LASTCHANGED	1376804898
variable	wert
uid	1
pid	0
t3ver_old	0
t3ver_id	0
t3ver_wsid	0
t3ver_label	
t3ver_state	0
t3ver_stage	0
t3ver_count	0
t3ver_tstamp	0
t3ver_move_id	0
t3_origuid	0
tstamp	1376804898
sorting	256
deleted	0

Tsconfig & TypeScript

Filelinks

- Filelinks bieten eine Beschreibung, einen Titel und einen alternativen Titel für jede Datei. Auf sämtliche drei Werte kann nun per Register zugegriffen werden:

```
register:description  
register:titleText  
register:altText
```

- Zum Beispiel:

```
# filelinks  
tt_content.uploads.20 {  
  # link description instead of filename  
  labelStdWrap.data = register:description  
  # output alternative text  
  itemRendering.20.data = register:titleText  
}
```

TSconfig & TypeScript

stdWrap function: replacement (1)

- Option `replace` der `stdWrap`-Funktion `replacement` unterstützt nun `optionSplit`
- Beispiel 1:

```
10 = TEXT
10.value = TYPO3_inspires_people_to_share
10.replacement.10 {
    search = _
    replace = 1 || 2 || 3
    useOptionSplitReplace = 1
}
```

Ausgabe:

TYPO31inspires2people3to3share

Tsconfig & TypeScript

stdWrap function: replacement (2)

- Option `replace` der `stdWrap`-Funktion `replacement` unterstützt nun `optionSplit`
- Beispiel 2:

```
10 = TEXT
10.value = TYPO3 inspires people to share
10.replacement.10 {
    search = #(TYPO3|people|share)#i
    replace = ${1} CMS || all ${1} || collaborate and ${1}
    useOptionSplitReplace = 1
    useRegExp = 1
}
```

Ausgabe:

TYPO3 CMS inspires all people to collaborate and share

Tsconfig & TypeScript

cObject FILE

- cObject FILES wurde um die Register FILE_NUM_CURRENT und FILES_COUNT erweitert
- Zum Beispiel:

```
10 = FILES
10 {
  references {
    table = tt_news
    uid.field = uid
    fieldName = media
  }
  renderObj = COA
  renderObj {
    10 = TEXT
    10.value = Renders first file twice
    10.if.isFalse.data = register:FILE_NUM_CURRENT
    20 = TEXT
    20.value = file {register:FILE_NUM_CURRENT} of {register:FILES_COUNT}
    20.insertData = 1
  }
}
```

TSconfig & TypoScript

Menü-Typ: Categories

- Es gibt nun die Möglichkeit, ein Kategorien-Menü in TypoScript zu erstellen
- Beispiel:

```
page.20 = HMENU
page.20 {
    special = categories
    special {
        # comma-separated list of categories
        value = 1
        # sort by title (stdWrap)
        sorting = title
        # sorting "asc" or "desc" (stdWrap)
        order = desc
        1 = TMENU
        1.NO {
            allWrap = <li> | </li>
        }
    }
}
```

Tsconfig & TypeScript

CSS and JavaScript files

- `splitChar` kann nun auf `allWrap` Eigenschaften angewandt werden
- "wrap" funktioniert nun wie die `stdWrap.wrap` Methode
- Standard `splitChar`-Zeichen ist das Pipe-Symbol: `|`
- Diese Erweiterung gilt für:
 - `includeCSS`
 - `includeJSlibs`
 - `includeJSFooterlibs`
 - `includeJS`
 - `includeJSFooter`

Tsconfig & TypeScript

Conditions

- Condition `userFunc` erlaubt die Angabe von mehreren Argumenten

- TYPO3 vor Version 6.2:

```
[userFunc = user_function(argument1)]
```

- TYPO3 ab Version 6.2:

```
[userFunc = user_function(argument1, argument2, ...)]
```

- Zum Beispiel:

```
[userFunc = user_match(checkSubnet, 192.168)]
```

```
function user_match($command, $subnet) {  
    switch($command) {  
        case 'checkSubnet':  
            if (strstr(getenv('REMOTE_ADDR'), $subnet)) { ... }  
    }  
}
```

Tsconfig & TypoScript

Conditions

- Application Context kann nun auch per Condition abgefragt werden
- Wildcards "+", "*", sowie reguläre Ausdrücke werden unterstützt
- Zum Beispiel:

```
[applicationContext = Development/Debugging, Development/Profiling]
# TYPO3 site in development stage
[global]
```

```
[applicationContext = Production*]
# TYPO3 site in production stage
# for example "Production/Live" or "Production/Staging"
[global]
```

```
[applicationContext = /^TestServer\d+$/]
# TYPO3 site on TestServer1 or TestServer2 or TestServer3, etc.
[global]
```


Tsconfig & TypoScript

Conditions

- Mittels devIP kann auf die im Install Tool eingetragene devIpMask geprüft werden
- Zum Beispiel:

```
[IP = devIP]
  page.10 = TEXT
  page.10.value = Hello Developer!
[global]
```

Datensätze ohne Standard Sprache

- Neue Option `includeRecordsWithoutDefaultTranslation` ermittelt Datensätze, die keine Entsprechung in der Default-Sprache haben, aber deren `languageField` der aktuellen Sprache entspricht
- Zum Beispiel:

```
pageContent = CONTENT
pageContent {
  table = tt_content
  select.includeRecordsWithoutDefaultTranslation = 1
  ...
}
```

TSconfig & TypeScript

cObject FILES

- cObject FILES verfügt nun über die Eigenschaften `begin` und `maxItems`
- Zum Beispiel:

```
page.10 = FILES
page.10 {
  references {
    table = pages
    uid.data = page:uid
    fieldName = media
  }

  # retrieve up to 5 files, beginning at the first (0):
  begin = 0
  maxItems = 5

  renderObj = TEXT
  renderObj {
    data = file:current:size
    wrap = <p>File size:<strong>|</strong></p>
  }
}
```

TSconfig & TypeScript

Seitenbaum manipulieren

- Bestimmte Doktypes können vom Seitenbaum ausgeschlossen werden
- Jenes erfolgt über die Einstellung `excludeDoktypes` in der `UserTSconfig` und ist somit Benutzer und Gruppen-spezifisch
- Zum Beispiel:

```
# exclude "folder" pages  
options.pageTree.excludeDoktypes = 254
```

```
# exclude "folder" and "standard" pages  
options.pageTree.excludeDoktypes = 254,1
```

Tsconfig & TypoScript

Module im BE ausblenden

- Man kann nun einzelne Module im Backend ausblenden
- Der Zugriff darauf wird allerdings nicht eingeschränkt (hierfür verwendet man nach wie vor die Berechtigungseinstellungen für BE-Benutzer und -Gruppen)

- Zum Beispiel:

```
options.hideModules = file, help
options.hideModules.web := addToList(func,info)
options.hideModules.system = BelogLog
```

TScnfig & TypoScript

Domain für Vorschau

- Im PageTScnfig ist es nun möglich, eine Domain anzugeben, die für die Anzeige verwendet wird
- Jenes ist beispielsweise für Multi-Domain-Sites nützlich
- Zum Beispiel:

```
TCEMAIN.viewDomain = example.com
```

Conditions in Backend-Layouts

- Innerhalb von Backend-Layouts können nun auch Conditions verwendet werden

```
backend_layout {  
  colCount = 2  
  rowCount = 1  
  rows {  
    1 {  
      columns {  
        1.name = Main  
        1.colPos = 0  
        2.name = Right  
        2.colPos = 1  
      }  
    }  
  }  
}
```

```
[PIDupinRootline = 123]  
# remove right column in branch of page ID 123  
backend_layout.rows.1.columns.2 >  
[global]
```

Tsconfig & TypoScript

Diverses

- Der "Passwort vergessen" Link kann nun über die Option `showForgotPassword` gezielt ein- und ausgeschaltet werden. Jenes ist sinnvoll, wenn sich mehrere Login-Formulare auf einer Seite befinden.
- HTTP Antwort beinhaltet standardmäßig den Header `Content-length`
 - Jenes läßt sich durch folgende Option beeinflussen:
`config.enableContentLengthHeader`
 - Beschleunigt das Rendering, wenn "Pipelining" im Apache aktiviert ist
- Ergebnisliste der `EXT:indexed_search` hat nun `stdWrap`-Eigenschaften (Option: `plugin.tx_indexedsearch.resultlist_stdWrap`)

Kapitel 5: Package Management

Package Management

Package Manager

- Der **Package Manager** von TYPO3 Flow wurde zu TYPO3 CMS portiert
- Planung wurde bereits während der TYPO3 CMS 6.1 Entwicklung begonnen
- Vereinheitlicht die Paket-Formate
- Extensions sind damit nur eine spezielle Art von "Packages"
- Hauptziele des Projekts:
 - Saubere API für Package Management
 - Unterstützung von Vendor Namespace
 - Unterstützung von Composer Packages
 - Unterstützung von Flow Packages
 - Überarbeitung des Autoloaders

Package Management

Integration des Package Managers

- Entfernung des Schlüssels `$TYPO3_CONF['EXT']['extListArray']` aus der Datei `typo3conf/LocalConfiguration.php`
- Inhalt der Datei `LocalConfiguration.php` wird umkopiert nach `typo3conf/LocalConfiguration.beforePackageStatesMigration.php`
- Datei `typo3conf/PackageStates.php` enthält:
 - Status der Packages (aktiv/inaktiv)
 - Positionen der Extensions im Filesystem
- Extensions in folgenden Verzeichnissen werden automatisch erkannt:

`typo3/sysex/`
`typo3/ext/`
`typo3/contrib/`

`typo3conf/ext/`
`Packages/ (rekursiv)`

Package Management

Integration des Package Managers

- Jede Extension erhält zwei zusätzliche Dateien:
 - `composer.json`
 - `Classes/Package.php`
- Wenn die Extension "required" ist, wird jenes über ein `protected` Flag in der Datei `composer.json` gesetzt
- Wenn die Extension "required" ist, wird jendes über die Eigenschaft `protected $protected` in der Datei `Classes/Package.php` gesetzt
- Fehlt die Datei `PackageStates.php`, wird sie neu erstellt und enthält Extensions, bei denen die Eigenschaft auf `TRUE` steht
- Autoloader bekommt ein eigenes Caching-Backend
- Weitere Informationen:
<http://wiki.typo3.org/Blueprints/Packagemanager>

Package Management

Integration des Package Managers

Beispiel: typo3conf/PackageManager.php

```
return array ('packages' =>
    array (
        'core' =>
            array (
                'manifestPath' => '',
                'composerName' => 'typo3/cms/core',
                'state' => 'active',
                'packagePath' => 'typo3/sysexst/core/',
                'classesPath' => 'Classes/',
            ),
        'workspaces' =>
            array (
                'manifestPath' => '',
                'composerName' => 'typo3/cms/workspaces',
                'state' => 'inactive',
                'packagePath' => 'typo3/sysexst/workspaces/',
                'classesPath' => 'Classes/',
            ),
        ...
    ),
    'version' => 4,
);
```

Package Management

Integration des Package Managers

Beispiel: composer.json

```
{
  "name": "typo3/cms-indexed-search",
  "type": "typo3-cms-framework",
  "description": "TYPO3 Core",
  "homepage": "http://typo3.org",
  "license": ["GPL-2.0+"],
  "version": "6.2.0",
  "require": {
    "typo3/cms-core": "*"
  },
  "replace": {
    "indexed_search": "*"
  }
}
```

Package Management

Integration des Package Managers

- Packages können auch zur Laufzeit aktiviert werden:

```
$GLOBALS['TYPO3_CONF_VARS']['EXT']['runtimeActivatedPackages'] =  
array( packageKey );
```

- Der Schlüssel wird unmittelbar nach der Initialisierung des Package Management aktiviert

Kapitel 6: Änderungen im System

Änderungen im System

Normalize.css

- Das Backend nutzt nun `normalize.css`, das dafür sorgt, dass Browser Elemente einheitlich unter Zuhilfenahme von modernen Webstandards dargestellt werden
- HTML5-ready und eine Alternative zu traditionellen CSS Reset Lösungen
- Ziele von `normalize.css` sind:
 - Beibehaltung sinnvoller Voreinstellungen, anstatt diese nur zu überschreiben
 - Normalisierung von Styles für diverse HTML Elemente
 - Behebung von Bugs und übliche Browser-Inkonsistenzen
 - Usability-Steigerung durch raffinierte Verbesserungen
 - Gute Dokumentation und Inline-Comments

Änderungen im System

TCA: displayCond Options BIT und !BIT

- TCA displayCond bitweise gegen Multi-Value-Felder testen
BIT: Bit ist gesetzt, !BIT: Bit ist nicht gesetzt

Angenommen sei
folgendes TCA:

```
'content' => array(  
  'label' => '...',  
  'config' => array(  
    'type' => 'check',  
    'items' => array(  
      array('Content A', ''),  
      array('Content B', ''),  
      array('Content C', ''),  
    ),  
  ),  
)  
,
```

Anwendungs-
beispiele:

```
'content_a' => array(  
  'label' => '...',  
  'displayCond' => 'FIELD:content:BIT:1',  
  'config' => array(  
    'type' => 'text',  
  )  
)  
,  
'content_b' => array(  
  'label' => '...',  
  'displayCond' => 'FIELD:content:!BIT:2',  
  'config' => array(  
    'type' => 'text',  
  )  
)  
,
```

Änderungen im System

Automatisches Update für Sprachen

- Ein Extbase Command Controller ermöglicht das automatische Update von Sprachen für Extensions:

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']['extbase']  
  ['commandControllers'][] =  
  'TYPO3\CMS\Lang\Command\LanguageCommandController';
```

- Ein möglicher Aufruf wäre:
`typo3/cli_dispatch.phpsh extbase language:update de,en,fr`
- Eine komma-separierte Liste von Locales (e.g. `de,en,fr`) aktualisiert lediglich diese Sprachen
- Ohne dieses Argument werden alle Sprachen aktualisiert, die im Modul "Sprache" ausgewählt sind

Änderungen im System

System Extension: ReST Manuals

- Manuals sämtlicher System Extensions wurde zu reStructuredText migriert
- OpenOffice Manuals werden nicht länger benötigt und wurde entfernt
- ReST ist ein leicht zu lesendes, "what-you-see-is-what-you-get", Text-basiertes Format
- ReST Dateien von System Extensions liegen unter:
`typo3/sysext/<extensionkey>/Documentation/*`
- Weitere Information unter:
<http://de.wikipedia.org/wiki/ReStructuredText>
<http://wiki.typo3.org/ReST>

Änderungen im System

Eigene Übersetzungsserver für Extensions

- Durch XLIFF und einem neuen Signal/Slot können nun eigene Übersetzungsserver für Extensions eingesetzt werden (nächste Slide zeigt ein Beispiel)
- Eine mögliche Lösung für einen Übersetzungsserver: **Pootle**
 - Online Translation Management Tool
 - geschrieben in Python/Django
 - original entwickelt und veröffentlicht von translate.org.za
 - freigegeben unter GNU GPL

Änderungen im System

Eigene Übersetzungsserver für Extensions

Beispiel: EXT:myextension/localconf.php

```
/**
 * @var \TYPO3\CMS\Extbase\SignalSlot\Dispatcher $signalSlotDispatcher
 */
$signalSlotDispatcher =
    \TYPO3\CMS\Core\Utility\GeneralUtility::makeInstance(
        'TYPO3\CMS\Extbase\SignalSlot\Dispatcher');

$signalSlotDispatcher->connect(
    'TYPO3\CMS\Lang\Service\UpdateTranslationService',
    'postProcessMirrorUrl',
    'Company\Extension\Slots\CustomMirror',
    'postProcessMirrorUrl'
);
```

Änderungen im System

Eigene Übersetzungsserver für Extensions

Beispiel: EXT:myextension/Classes/Slots/CustomMirror.php

```
<?php
namespace Company\Extensions\Slots;
class CustomMirror {

    /**
     * @var string
     */
    protected static $extKey = 'myextension';

    public function postProcessMirrorUrl($extensionKey, &$mirrorUrl) {
        if ($extensionKey === self::$extKey) {
            $mirrorUrl = 'http://example.com/typo3-packages/';
        }
    }
}
```

Änderungen im System

Eigene Übersetzungsserver für Extensions

Vorausgesetzte Datei/Verzeichnis-Struktur auf dem Server:

```
http://example.com/typo3-packages/  
  '-- <first-letter-of-extension-key>  
    '-- <second-letter-of-extension-key>  
      '-- <extension-key>-l10n  
        |-- <extension-key>-l10n-de.zip  
        |-- <extension-key>-l10n-fr.zip  
        |-- <extension-key>-l10n-it.zip  
      '-- <extension-key>-l10n.xml
```

Zum Beispiel:

```
http://example.com/typo3-packages/m/y/myextension-l10n/myextension-l10n.xml
```


Änderungen im System

Eigene Übersetzungsserver für Extensions

Beispiel: <extension-key>-l10n.xml

```
<?xml version="1.0" standalone="yes" ?>
<TERlanguagePackIndex>
  <meta>
    <timestamp>1374841386</timestamp>
    <date>2013-07-26 14:23:06</date>
  </meta>
  <languagePackIndex>
    <languagepack language="de">
      <md5>1cc7046c3b624ba1fb1ef565343b84a1</md5>
    </languagepack>
    <languagepack language="fr">
      <md5>f00f73ae5c43cb68392e6c508b65de7a</md5>
    </languagepack>
    <languagepack language="it">
      <md5>cd59530ce1ee0a38e6309544be6bcb3d</md5>
    </languagepack>
  </languagePackIndex>
</TERlanguagePackIndex>
```

Änderungen im System

Automatisierter t3d Import

- Extensions können während ihrer Installation nun automatisch **t3d Packages** importieren
- t3d Packages beinhalten zum Beispiel Daten, Relationen, Dateien, usw.
- Die t3d Datei muss `data.t3d` lauten und sich in folgendem Verzeichnis befinden: `EXT:myextension/Initialisation/`
- Der Import geschieht nur einmalig, selbst wenn die Extension später erneut installiert werden sollte

Änderungen im System

Automatisierter Datei Import

- Extensions können während ihrer Installation nun automatisch **Dateien** importieren
- Dateien werden bereit gestellt unter:
EXT:myextension/Initialisation/Files/...
- Dateien werden während der Extension-Installation kopiert nach:
fileadmin/<extensionkey>/
- Der Import geschieht nur einmalig, selbst wenn die Extension später erneut installiert werden sollte

Änderungen im System

Extensions als Repositories nutzen

- Manchmal hängen Extensions von Extensions ab, die verändert und/oder angepasst wurden, oder von Extensions, die nicht im offiziellen TYPO3 Extension Repository (TER) veröffentlicht wurden
- Um diese Abhängigkeiten aufzulösen, können Extensions nun andere Extensions enthalten:
`EXT:myextension/Initialisation/Extensions/...`
- Während der Installation werden diese kopiert nach:
`typo3conf/ext/`
- Anschließend werden weitere Abhängigkeiten der Extension aufgelöst

Änderungen im System

(De-)Installation von Extensions über CLI

- Extensions können nun via Command Line Interface (CLI), also über die Kommandozeile, installiert und de-installiert werden
- Beispiele:

```
typo3/cli_dispatch.phpsh extbase extension:install <extensionkey>
```

```
typo3/cli_dispatch.phpsh extbase extension:uninstall <extensionkey>
```
- Anmerkung: Backend-Benutzer **_cli_lowlevel** muss hierfür existieren!

Änderungen im System

TCA: Kaskadierendes Löschen von Datensätzen

- Im TCA gibt es nun eine Einstellung, die es erlaubt, das kaskadierende Löschen von Kind-Datensätzen ein- bzw. auszuschalten
- Dafür muss die Relation vom Typ **"inline"** sein
- Standardwert ist TRUE
(d.h. beim Löschen eines Datensatzes werden die Kind-Datensätze automatisch mitgelöscht)
- Beispiel (Deaktivierung der Löschung von Kind-Datensätzen):

```
...
'type' => 'inline',
'foreign_table' => ...,
  'behaviour' => array(
    'enableCascadingDelete' => 0
  )
...
)
...
```

Änderungen im System

Mehrere Kategorien pro Tabelle

- Bisher konnte nur ein `makeCategorizable()`-Aufruf per Tabelle ausgeführt werden
(weitere Aufrufe überschrieben die vorhergehenden)
- Seit TYPO3 CMS 6.2 können beliebig viele solcher Felder existieren
- Beispiel:

```
\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::makeCategorizable(  
    $extensionKey,  
    $tableName,  
    $fieldName = 'categories',  
    $options = array(  
        'label' => 'my category'  
    )  
);
```

- Mit `$options` kann ein eigenes Label für das Feld eingebracht werden

Änderungen im System

Data Provider für Backend Layouts

- In TYPO3 vor Version 6.2 werden Backend Layouts ausschließlich in der Datenbank gespeichert
- Seit TYPO3 ab Version 6.2 können sogenannte *Data Provider* definiert werden
(Extensions können somit ihre eigenen Backend Layout Definitionen liefern)
- Data Provider müssen das folgende Interface implementieren:
TYPO3\CMS\Backend\View\BackendLayout\DataProviderInterface
- und werden wie folgt registriert:

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
    ['BackendLayoutDataProvider'][$_EXTKEY] = 'Classname';
```


Änderungen im System

Backend Layout Data Providers

- Zudem gibt es neue API-Befehle für das Backend Layout Handling, z.B.:

```
'itemsProcFunc' => 'TYPO3\\CMS\\Backend\\View\\  
BackendLayoutView->addBackendLayoutItems'
```

```
getBackendLayoutView()->getSelectedCombinedIdentifier($id);  
getBackendLayoutView()->getSelectedBackendLayout();
```

- PageTSconfig Option, um Backend Layouts von der Zuweisung auszuschließen:

```
options.backendLayout.exclude = default_1, my_extension__headerLayout
```

Änderungen im System

Multiple Value Selector (1)

- Ein Filter ermöglicht es, Einträge in einem Multi-Select Feld auszublenden, um die Suche deutlich zu vereinfachen
- Dafür muss das TCA entsprechend angepasst werden (z.B. in der Datei `typo3conf/ext/Tables.php`):

```
$GLOBALS['TCA']['fe_users']['columns']['usergroup']['config']
    ['enableMultiSelectFilterTextfield'] = TRUE;

$GLOBALS['TCA']['fe_users']['columns']['usergroup']['config']
    ['multiSelectFilterItems'] = array(

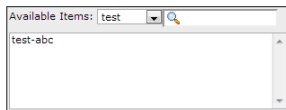
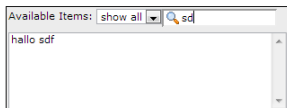
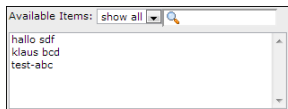
        array('', 'show all'), // no filter
        array('test', 'test'), // first value: filter, second value: label

        array(
            'TYPO3',
            'LLL:EXT:myext/Resources/Private/Language/locallang_db.xlf:tx_myext.label.typo3'
        ),
    );
```

Änderungen im System

Multiple Value Selector (2)

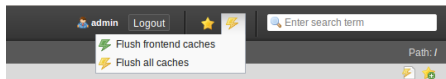
- Zwei Optionen stehen zur Verfügung:
 - Auswahl aus einer Dropdown Box mit vordefinierten Begriffen
 - Textfeld zur Eingabe eines Such/Filter-Begriffes
- Das Ergebnis könnte dann beispielsweise wie folgt aussehen:



Änderungen im System

Cache Gruppen (1)

- Der TYPO3 Core nutzt nun zwei verschiedene Cache Gruppen:
 - **System Caches:** class loading cache, configuration cache, l10n_cache, extbase_object, extbase_reflection etc.
 - **Frontend Caches:** cHash cache, page cache, page section cache
- In TYPO3 vor 6.2 leerte *clear all caches* sämtliche Caches (nicht ideal)
- In TYPO3 ab 6.2 macht der Core Gebrauch von zwei Cache Gruppen: "**pages**" für alle Seiten-relavanten Caches und "**system**", welches für Compile-Time und Konfigurations-Caches zuständig ist



Änderungen im System

Cache Groups (2)

- Entsprechende Konfigurationsoption:

(in den Dateien: LocalConfiguration.php/DefaultConfiguration.php)

```
'cache_hash' => array(
    'frontend' => 'TYPO3\CMS\Core\Cache\Frontend\VariableFrontend',
    'backend' => 'TYPO3\CMS\Core\Cache\Backend\Typo3DatabaseBackend',
    'options' => array(),
    'groups' => array('pages', 'all')
),
```

- "*Flush all caches*" leert nicht mehr System Caches
(nur "*Clear Configuration Cache*" oder das Install Tool leert diese)
- Neue userTSconfig Option ermöglicht es Nicht-Admins System Caches zu leeren:

```
options.clearCache.system = 1
```

BREAKING CHANGE!

Änderungen im System

TCA: Anzahl Aktivierte Checkboxes

- TCA erlaubt nun die Prüfung/Limitierung der Anzahl aktivierter Checkboxes
 - `maximumRecordsChecked`:
Limitiert die maximale Anzahl Datensätze von der selben Tabelle
 - `maximumRecordsCheckedInPid`:
Limitiert die maximale Anzahl Datensätze auf der selben PID (parent ID)
- Wählt der Backend Benutzer weitere Checkboxes nach dem Erreichen der maximal zulässigen Anzahl aus, so werden diese wieder deaktiviert
- Example:

```
$tcaConfiguration = array(  
    'type' => 'check',  
    'eval' => 'maximumRecordsChecked',  
    'validation' => array(  
        'maximumRecordsChecked' => 5  
    )  
);
```

Änderungen im System

TCA: MM_oppositeUsage Eigenschaft

- Wenn ein sys_category Datensatz kopiert wird, wird eine neue MM-Referenz angelegt, allerdings ohne den "fieldname" zu setzen
- Der Wert wird prinzipiell von der Gegenseite mittels MM_match_fields definiert, auf den man allerdings bislang nicht zugreifen konnte
- Um dieses Problem zu lösen, wurde die neue Eigenschaft MM_oppositeUsage eingeführt:

```
'config' => array(
    'allowed' => '*',
    'MM' => 'tx_myextension_first_second_mm',
    'MM_oppositeUsage' => array(
        'tt_content' => array('somefield'),
        'tx_myextension_domain_model' => array('some_property'),
    ),
),
```

Änderungen im System

Diverses

- **Eigene Record-List:**

Es ist nun möglich, eine eigene Record-List im Element-Browser zu verwenden und die Standard Record-List damit zu überschreiben.

- **Mehr Untergruppen:**

Das Attribute subgroup in der Datenbank-Tabelle be_groups wurde von varchar(250) zu text geändert, was die mögliche Anzahl von Untergruppen für Backend Benutzer deutlich steigert.

- **Zusammengeführte TS/Template Extensions:**

Technisch gesehen war "WEB > Template" bisher auf mehrere Extensions verteilt (tstemplate_ceditor, tstemplate_info, tstemplate_objbrowser und tstemplate_analyzer). Diese wurden nun in eine einzige Extension zusammengeführt: "tstemplate".

Änderungen im System

Diverses

■ **label_userFunc_options:**

Die TCA-Option `label_userFunc_options` wurde zur Klasse `BackendUtility` hinzugefügt.

■ **Dateinamen von Extensions:**

Wenn eine Extension im Extension Manager heruntergeladen wird, enthält der Dateiname nun den aktuellen Zeitstempel (Jahr, Monat, Tag und Zeit):

```
<extensionKey>_<version>_<timestamp>.zip  
myextension_1.0.0_201403252359.zip
```

■ **EXT:saltedpasswords:**

Die `EXT:saltedpasswords` ist nun für das Backend standardmäßig aktiviert. Jenes erzwingt `salted` Passwörter für die Anmeldung am Backend. Das Install Tool prüft die Konfiguration und passt sie entsprechend an, sofern erforderlich.

Änderungen im System

Diverses

- **SignalSlots können Parameter manipulieren:**

Parameter, die an einen SignalSlot-Dispatcher übergeben werden, können nun verändert werden und der Dispatcher liefert diese modifizierten Argumente auch wieder zurück.

- **Vorschau einer Arbeitsumgebung:**

Sämtliche HTTP Parameter werden nun in der Arbeitsumgebungs-Vorschau (Workspace Preview) berücksichtigt. Jenes war ein Problem in TYPO3 CMS vor 6.2, wenn Extensions nicht wie erwartet funktionierten, da sie auf bestimmte Parameter angewiesen waren.

- **TCEforms Placeholder feature:**

Das bereits in TYPO3 CMS 4.7 eingeführte *Placeholder* Feature der TCEforms funktioniert nun rekursive (z.B. `__row|uid_foreign|field`).

Änderungen im System

Diverses

■ **Icons mit hoher Auflösung:**

Der Sprite-Generator unterstützt nun Icons mit doppelter Auflösung. Er erzeugt eine zweite Datei, die das Suffix @x2.png trägt, und die die hohe Auflösung enthält. Über CSS3 wird dafür gesorgt, dass die hochauflösenden Icons auf Geräten dargestellt werden, die jenes unterstützen, ohne die Performance auf anderen Geräten zu beeinflussen.

■ **Proxy NTLM Authentifizierung:**

Unterstützung von Proxy NTLM Authentifizierung (**NT LAN Manager**: eine Sammlung von Microsoft Sicherheitsprotokollen). Jenes kann über das Install Tool aktiviert werden:

```
$GLOBALS['TYPO3_CONF_VARS']['SYS']['curlProxyNTLM']
```

(Anmerkung: dieses Feature wurde vor über 8 Jahren vorgeschlagen :-)

Änderungen im System

Diverses

■ **cookieHttpOnly standardmäßig:**

Um Session Cookies nur noch über das HTTP Protokoll zugänglich zu machen, ist `cookieHttpOnly` nun standardmäßig aktiviert.

Das bedeutet, dass die Cookies "fe_typo_user" und "be_typo_user" nicht mehr über Scriptsprachen (z.B. JavaScript) zugänglich sind, was den Schutz vor XSS (*Cross Site Scripting*) Angriffen verbessert. Allerdings unterstützen ältere Browser diese Technologie nicht.

■ **Aufgeräumte Datenbanktabellen:**

Die folgenden Attribute wurden von der Tabelle `tt_content` entfernt (diese werden schon seit TYPO3 4.0 nicht mehr verwendet): `text_align`, `text_face`, `text_size`, `text_color`, `text_properties`.

Änderungen im System

Diverses

- **HTML Tidy entfernt:**

Die *HTML Tidy* Funktionalität wurde vom TYPO3 Core entfernt, kann aber durch die Extension EXT:tidy ohne Probleme wiederhergestellt werden.

- **dontSetCookie entfernt:**

Da das Cookie "fe_typo_user" nur gesetzt wird, sofern benötigt (und nicht mehr immer), wurde die Install Tool Option dontSetCookie irrelevant und daher entfernt.

- **"Wizard" Scripts entfernt:**

Die folgenden "Wizard" Scripts wurden entfernt: typo3/wizard_add.php, typo3/wizard_colorpicker.php, typo3/wizard_edit.php, typo3/wizard_forms.php, typo3/wizard_list.php, typo3/wizard_rte.php, typo3/wizard_table.php

Kapitel 7: Application Programming Interface (API)

Application Programming Interface

Hook: `tsfe::checkEnableFields`

- In TYPO3 CMS vor 6.2 kann das Feature "auf Unterseiten ausdehnen" (*extend to subpages*) nicht in eigenen Extensions verwendet werden, um zusätzliche Regeln für die Sichtbarkeit von Seiten bereitzustellen (Liste der zu prüfenden Felder ist hard-coded in `tsfe::checkEnableFields()`)
- In TYPO3 CMS ab 6.2 kann durch einen neuen Hook diese Liste beliebig erweitert werden

- Klasse:

```
\TYPO3\CMS\Frontend\Controller\TypoScriptFrontendController  
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
    ['tslib/class.tslib_fe.php']['hook_checkEnableFields']
```

Application Programming Interface

Hook: `checkFlexFormValue` im `DataHandler`

- In TYPO3 CMS vor 6.2 wird vor dem Speichern von FlexForm-Daten nicht geprüft, ob der Wert selbst bereits aus der Datenbank gelöscht wurde
- Jenes ist problematisch, wenn `SwitchableControllerActions` (Extbase) verwendet werden: alte Actions, die es nicht mehr gibt, müssen manuell aus der Datenbank entfernt werden
- In TYPO3 CMS ab 6.2 können mit einem neuen Hook FlexForm-Daten vor dem Speichern verändert werden

- Klasse:

```
\TYPO3\CMS\Core\DataHandling\DataHandler  
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
    ['t3lib/class.t3lib_tcemain.php']['checkFlexFormValue']
```

- Methode:

```
checkFlexFormValue_beforeMerge()
```


Application Programming Interface

Hook für den Header im PageLayoutController

- Mit diesem Hook kann der Header einer Seite im Modul "Web > Page" erweitert werden
- Der Hook wird ausgeführt, bevor der Seiteninhalt gerendert wird
- Klasse:

```
\TYPO3\CMS\Backend\Controller\PageLayoutController
```

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']
```

```
['cms/layout/db_layout.php']['drawHeaderHook']
```

- Methode:
callUserFunction()

Application Programming Interface

IRRE: Defaults für erzeugte Werte

- Für IRRE Datensätze gibt es eine neue TCA-Option für "inline" Felder
- Über `foreign_record_defaults` können Default-Werte für erzeugte Datensätzen gesetzt werden

```
'config' => array(  
    'type' => 'inline',  
    'foreign_table' => 'tt_content',  
    'foreign_record_defaults' => array(  
        'CType' => 'image'  
    ),  
)
```

Beispiel oben: `tt_content`-Elemente, die für dieses IRRE Feld erstellt wurden, sind standarmäßig **image**-Inhaltselemente (jedoch können Redakteure diese natürlich vor dem Speichern ändern)

Application Programming Interface

Arbeitsumgebung (1)

- In TYPO3 CMS vor 6.2 kann das Modul Arbeitsumgebung ("Workspaces") nur durch das Überschreiben von PHP und JavaScript Komponenten erweitert werden
- In TYPO3 CMS ab 6.2 ist es nun möglich, die Definitionen und das Verhalten der dargestellten Spalten des Moduls zu manipulieren
- Einige Beispiele auf den folgenden Slides...

Application Programming Interface

Arbeitsumgebung (2)

Beispiel (Datei ext_localconf.php):

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
    ['t3lib/class.t3lib_tcemain.php']['processCmdmapClass']['workspaces_logger'] =  
    'Vendor\\WorkspacesLogger\\Hook\\DataHandlerHook';
```

Beispiel (Datei ext_tables.php):

```
\TYPO3\CMS\Workspaces\Service\AdditionalColumnService::getInstance()->register(  
    'WorkspacesLogger_StageChange',  
    'Vendor\\WorkspacesLogger\\DataProvider'  
);  
  
\TYPO3\CMS\Workspaces\Service\AdditionalResourceService::getInstance()->addJavaScriptResource(  
    'WorkspacesLogger',  
    'EXT:myextension/Resources/Public/JavaScript/StageChange.js'  
);
```

Application Programming Interface

Arbeitsumgebung (3)

Beispiel (Datei Vendor\WorkspacesLogger\Hook\DataHandlerHook):

```
<?php
namespace Vendor\WorkspacesLogger\Hook;
use TYPO3\CMS\Core\SingletonInterface;

class DataHandlerHook implements SingletonInterface {

    const TABLE_Name = 'tx_workspaceslogger_event';
    const EVENT_SetStage = 91;

    /**
     * hook that is called when no prepared command was found
     */
    public function processCmdmap($command, $table, $id, $value, &$commandIsProcessed,
        \TYPO3\CMS\Core\DataHandling\DataHandler $tceMainObj) {
        ...
        $action = (string) $value['action'];
        if ($command === 'version' && $action === 'setStage' && $commandIsProcessed) {
            ...
        }
    }
}
```

Application Programming Interface

PSR-3 kompatibles Logging

- Die TYPO3 CMS 6.2 Logging API ist nun PSR-3 kompatibel
- PSR-3 soll einen Standard für das Logging in PHP definieren (PHP Framework Interop Group)
- Das Hauptziel von PSR-3 ist es, *"dass Bibliotheken ein LoggerInterface Objekt erhalten und in einer einfachen und universellen Weise Protokoll-Einträge in dieses schreiben"*
- LoggerInterface enthält Methoden, wie beispielsweise `debug()`, `warning()`, `notice()`, `alert()`, `error()`, etc.
- Weitere Informationen:
<http://www.php-fig.org/psr/3/>

CSRF-geschützte Ajax Calls

- Ajax Calls im TYPO3 Backend können nun wirksam gegen CSRF (*Cross-Site Request Forgery*) geschützt werden, indem deren Handler registriert wird

```
\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::registerAjaxHandler(  
    'TxMyExt::process',  
    '\Vendor\MyExt\AjaxHandler->process'  
);
```

- URL für die Ajax ID beinhaltet einen CSRF Token, der im `ajax.php`-Dispatcher überprüft wird

```
$ajaxUrl = \TYPO3\CMS\Core\Utility\BackendUtility::getAjaxUrl('TxMyExt::process');
```

- Die Variable kann im JavaScript Kontext auf der Seite verwendet werden

```
var ajaxUrl = TYPO3.settings.MyExt.ajaxUrl;
```

Application Programming Interface

Diverses

- Neue Methode `canBeInterpretedAsFloat()`
in der Klasse `MathUtility`
(analog zu: `canBeInterpretedAsInteger()`)
- Neuer Enumeration-Type (ohne die Verwendung von Modulen von
Drittanbietern): `\TYPO3\CMS\Core\Type\Enumeration`

Zum Beispiel verwendet in:

```
\TYPO3\CMS\Core\Versioning\VersionState
```

...und dann im Code mittels:

```
new VersionState(VersionState::DEFAULT_STATE);
```


Kapitel 8: Extbase & Fluid

ObjectManager->getScope()

- Über die Methode ObjectManager->getScope() kann nun überprüft werden, ob eine Klasse vom Typ **Prototype** oder **Singleton** ist

```
/**
 * @var \TYPO3\CMS\Extbase\Object\ObjectManagerInterface
 * @inject
 */
protected $objectManager;

$this->objectManager->getScope($propertyTargetClassName) ===
    \TYPO3\CMS\Extbase\Object\Container\Container::SCOPE_PROTOTYPE

$this->objectManager->getScope($propertyTargetClassName) ===
    \TYPO3\CMS\Extbase\Object\Container\Container::SCOPE_SINGLETON
```

Ermittlung des PageTypes in URIs

- Das Attribut `pageType` ist nicht mehr erforderlich, wenn spezielle Seitentypen gerendert werden

TYPO3 vor Version 6.2:

```
<f:link.action arguments="{blog: blog}" pageType="{settings.plaintextPageType}"
  format="txt">[plaintext]</f:link.action></li>
```

- Über eine neue TypeScript Option `formatToPageTypeMapping` kann ein globales Mapping erreicht werden

```
plugin.tx_myextension {
  view.formatToPageTypeMapping {
    txt = 99
    pdf = 123
  }
}
```

TYPO3 ab Version 6.2:

```
<f:link.action arguments="{blog: blog}"
  format="txt">[plaintext]</f:link.action></li>
```

Extbase & Fluid

Object Type Converter (1)

- Der in TYPO3 Flow eingeführte **Object Type Converter** wurde nach Extbase portiert
- Damit ist es möglich, Arrays in nicht-persistente Objekte zu konvertieren
- Einige Beispiele auf den folgenden Slides...

Extbase & Fluid

Object Type Converter (2)

GET request

```
http://example.com/index.php?id=299
&tx_myextension[action]=list
&tx_myextension[controller]=Entity
&tx_myextension[demand][title]=foo
&tx_myextension[demand][relation]=1
```

Entity controller: initializeListAction()

```
use [Vendor]\myextension\Domain\Dto\Demand;
public function initializeListAction() {
    /**
     * @var PropertyMappingConfiguration $demandConfiguration
     */
    $demandConfiguration = $this->arguments['demand']->getPropertyMappingConfiguration();
    $demandConfiguration->allowAllProperties()->forProperty('relation')->allowAllProperties()->
        setTypeConverterOption(
            'TYPO3\CMS\Extbase\Property\TypeConverter\PersistentObjectConverter',
            PersistentObjectConverter::CONFIGURATION_CREATION_ALLOWED,
            TRUE
        );
}
```

Extbase & Fluid

Object Type Converter (3)

Entity controller: listAction()

```
use [Vendor]\myextension\Domain\Dto\Demand;
/**
 * @var PropertyMappingConfiguration $demandConfiguration
 */
public function listAction(Demand $demand = NULL) {
    $entities = $this->entityRepository->findAll();
    $this->view->assign('entities', $entities);
}
```

Model: [Vendor]\myextension\Domain\Dto\Demand.php

```
namespace [Vendor]\myextension\Domain\Dto;
use [Vendor]\myextension\Domain\Model\Relation;
class Demand {
    protected $relation;
    /**
     * @param \TYPO3Friends\MapperExample\Domain\Model\Relation $relation
     */
    public function setRelation($relation) {
        $this->relation = $relation;
    }
}
```

Verketteten von QuerySettings

- Das Chaining (Verketteten) von QuerySettings ist nun auch mit den Optionen `setIncludeDeleted` und `setIgnoreEnableFields` möglich, die mit TYPO3 CMS 6.0 eingeführt wurden:

```
$query->getQuerySettings()  
->setRespectStoragePage(FALSE)  
->setRespectSysLanguage(FALSE)  
->setIgnoreEnableFields(TRUE)  
->setIncludeDeleted(TRUE);
```

Extbase & Fluid

RawQueryResults per Query

- Bisher war es möglich, mittels folgendem Statement dafür zu sorgen, dass keine Objekte vom QueryManager rekonstruiert werden, sondern dass das Query-Ergebnis "roh" zurückgegeben wird:

```
$query->getQuerySettings()->setReturnRawQueryResult(TRUE);
```

- Diese Möglichkeit gibt es nicht mehr zentral, sondern man gibt dies per Query im execute-Statement mittels TRUE an:

```
$query->execute(TRUE);
```


Extbase & Fluid

Rekursive Validierung

- Extbase verwendet nun die von TYPO3 Flow portierte sogenannte **"Rekursive Validierung"**
- Dies bedeutet, dass bei Erzeugung von verschachtelten Objekten (Objekt-Baum) durch den Property-Mapper auch die inneren Objekte validiert werden und nicht nur das äußere Objekt, wie bisher
- Zudem ist es nun möglich, leere (optionale) Werte zuzulassen. Will man daher explizit erreichen, dass eine Eigenschaft angegeben werden muss, so muss der `NotEmptyValidator` verwendet werden

BREAKING CHANGE!

Application Context

- Der aktuelle Application Context (wird als Umgebungsvariable `TYPO3_CONTEXT` oder im Install Tool gesetzt) kann wie folgt abgefragt werden:

```
\TYPO3\CMS\Core\Core\Bootstrap::getInstance()->getContext();
```

```
\TYPO3\CMS\Core\Utility\GeneralUtility::getContext();
```

ViewHelper: Image

- Im Fluid ViewHelper **image** ist das Attribut `title` nun optional

Beispiel:

```
<f:image src="background.jpg" alt="Text" />
```

TYPO3 vor Version 6.2:

```

```

TYPO3 ab Version 6.2:

```

```

ViewHelper: textfield und textarea

- Die beiden ViewHelper **form.textfield** und **form.textarea** können nun mit den Argumenten `autofocus` und `placeholder` ausgestattet werden

Beispiel ("placeholder"):

```
<f:form.textfield
  id="powermail_field_{field.marker}"
  ...
  placeholder="{field.title -> vh:string.RawAndRemoveXss()}"
  ...
  name="field[{field.uid}]"
  required="{field.mandatory}" />
```

ViewHelper: switch

- Es gibt einen neuen Fluid ViewHelper **switch**, der der gleichnamigen PHP Funktion ähnelt:

```
<f:switch expression="{person.gender}">
  <f:case value="male">Mr.</f:case>
  <f:case value="female">Mrs.</f:case>
  <f:case default="TRUE">Mrs. or Mr.</f:case>
</f:switch>
```

- **Hinweis:** die exzessive Verwendung dieses ViewHelpers könnte auf eine schlechte Architektur hindeuten. So könnte das Beispiel oben mit den entsprechenden Partialen "title.male.html" und "title.female.html" auch wie folgt gelöst werden:

```
<f:render partial="title.{person.gender}" />
```

ViewHelper: fileSize

- Dieser ViewHelper konvertiert eine Byte-Angabe (in Form eines Integer-Wertes) in ein Menschen-lesbares Format

Beispiel 1 (fileSize = 1263616):

```
fileSize -> f:format.bytes()
```

Ausgabe: "1234 KB"

Beispiel 2 (fileSize = 1263616):

```
fileSize -> f:format.bytes(  
    decimals: 2,  
    decimalSeparator: '.',  
    thousandsSeparator: ',',  
)
```

Ausgabe: "1,234.00 KB"

ViewHelper: format.date

- Standardwert des ViewHelper **format.date** ist der im Install Tool konfigurierte Wert
`$GLOBALS['TYPO3_CONF_VARS']['SYS']['ddmmyy']`
- Ist kein Wert im Install Tool gesetzt, wird "Y-m-d" verwendet (Jahr, Monat, Tag)

Extbase & Fluid

ViewHelper Backend Container

- Der Fluid ViewHelper backend container (`be.container`) wurde überarbeitet:

`typo3/sysext/fluid/Classes/ViewHelpers/Be/ContainerViewHelper.php`

Überholt (deprecated):

- `$addCssFile` (wird abgelöst von: `$includeCssFiles`)
- `$addJsFile` (wird abgelöst von: `$includeJsFiles`)

Neu:

- `$loadJQuery`
- `$includeCssFiles`
- `$includeJsFiles`
- `$addJsInlineLabels`

Extbase & Fluid

ViewHelper: button.icon

- Fluid ViewHelper **button.icon** ist nicht mehr "experimentell"
- Liefert ein Button-Icon zurück (ggf. mit Link)

```
<f:be.buttons.icon  
    icon="actions-document-new" title="Create new Foo" />
```

```
<f:be.buttons.icon uri="{f:uri.action(action:'new')}"  
    icon="actions-document-new" title="Create new Foo" />
```

- Das Attribut `icon` unterstützt derzeit über 310 Werte!

Siehe:

```
$GLOBALS['TBE_STYLES']['spriteIconApi']['coreSpriteImageNames']
```

...in der Datei:

```
typo3/systext/core/ext_tables.php
```

Option `addQueryStringMethod`

- Option `addQueryString` unterstützt lediglich **GET**-Argumente, die dann an den erzeugten Link angehängt werden
- **POST**-Argumente (wie sie z.B. bei Widgets vorkommen) funktionieren mit dieser Option allerdings nicht
- Über die neue Option `addQueryStringMethod` kann nun definiert werden, welche Parameter verwendet werden sollen: GET (default), POST, GET/POST oder POST/GET
- Folgende ViewHelper wurden mit dieser Option ausgestattet:
 - `link.action`
 - `link.page`
 - `uri.action`
 - `uri.page`
 - `widget.link`
 - `widget.uri`
 - `widget.paginate`

Extbase & Fluid

Fluid: Template Fallback Pfad

- Fluid erlaubt nun "Fallback" Pfade für Templates, Partials und Layouts:
templateRootPaths, partialRootPaths, layoutRootPaths
- Reihenfolge der Verzeichnisse, die geprüft werden: höchster Index zuerst, dann Verzeichnisse mit entsprechend kleinerem Index

```
plugin.tx_myextension {  
    view {  
        templateRootPath = EXT:myextension/Resources/Private/Templates/  
    }  
}
```

```
plugin.tx_myextension {  
    view {  
        templateRootPath >  
        templateRootPaths {  
            10 = fileadmin/myextension/Templates/  
            20 = EXT:myextension/Resources/Private/Templates/  
        }  
    }  
}
```

Kapitel 9: Upgrade zu TYPO3 CMS 6.2 LTS

Upgrade zu TYPO3 CMS 6.2 LTS

Generelles Vorgehen

- Anleitung zum Upgrade:
http://wiki.typo3.org/Upgrade#Upgrading_to_6.2
- Offizielles TYPO3 Guide "TYPO3 Installation and Upgrading":
<http://docs.typo3.org/typo3cms/InstallationGuide>
- Generelles Vorgehen:
 - Prüfen, ob das System die Mindestvoraussetzungen erfüllt (PHP, MySQL, usw.)
 - **deprecation_*.log** der alten TYPO3 Installation durchsehen
 - Sämtliche Extensions auf den aktuellste Stand bringen (auf TYPO3 6.2 Kompatibilität achten)
 - Installation: siehe Kapitel "Install Tool" in dieser Präsentation

Upgrade zu TYPO3 CMS 6.2 LTS

Upgrade von TYPO3 CMS 4.5 LTS

- Viele TYPO3 Installationen werden von der bisherigen LTS zur neuen LTS Version aktualisieren (4.5 zu 6.2)
- Smooth Migration Project:
 - Hat sich zum Ziel gesetzt, die Migration von 4.5 zu 6.2 so unkompliziert wie möglich zu gestalten
 - Liefert Dokumentationen und sammelt Infos über mögliche Probleme in Extensions usw.
 - <http://forge.typo3.org/projects/typo3cms-smoothmigration>
- EXT:typo3-upgradereport:
 - Hauptsächlich von Steffen Ritter entwickelt
 - Wird in einer TYPO3 CMS 4.5 LTS installiert und durchläuft etliche Tests
 - Mithilfe zur Optimierung der Extension ist natürlich erwünscht
 - <https://github.com/nxpthx/typo3-upgradereport>

Upgrade zu TYPO3 CMS 6.2 LTS

Infos speziell für Redakteure

- Fasst die wichtigsten Änderungen zwischen TYPO3 CMS 4.5 LTS und TYPO3 CMS 6.2 LTS zusammen
- Zielgruppe: speziell Redakteure (nicht-technische End-Benutzer)
- Soll außerdem Agenturen unterstützen:
 - Zur Vorbereitung auf Supportanfragen ihrer Kunden
 - Zur Unterstützung bei Workshops, Seminaren, Schulungen, usw.
- Dokument "**What's New for Editors**" kann kostenlos heruntergeladen werden:

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

Kapitel 10: TYPO3 CMS 6.2 LTS – MythBuster

Mythen über TYPO3 CMS 6.2

- TYPO3 CMS 6.2 LTS wird das letzte TYPO3 CMS Release sein → **falsch!**
Die Wahrheit ist, dass trotz der Veröffentlichung von [TYPO3 Neos](#) die Entwicklung von TYPO3 CMS fortgesetzt wird und dass es auch in Zukunft weitere Versionen geben wird.

- Der TYPO3 Core wurde in 6.x komplett überarbeitet → **falsch!**
Die Wahrheit ist, dass mit TYPO3 CMS 6.0 das PHP Konzept von "[Namespaces](#)" eingeführt wurde, was neue Klassennamen zur Folge hat. Allerdings stellt eine Kompatibilitätsschicht in TYPO3 sicher, dass Entwickler noch weiterhin die bisherigen, veralteten Klassennamen ohne Probleme in ihren Extensions verwenden können.

Mythen über TYPO3 CMS 6.2

- Extensions, die für TYPO3 CMS 4.5 entwickelt wurden, funktionieren in 6.2 nicht mehr **→ falsch!**
Die Wahrheit ist, dass die Core API nicht vollständig geändert wurde und eine Abwärtskompatibilität aufweist, die der [Deprecation Policy](#) entspricht. Die meisten Extensions, die für TYPO3 CMS 4.5 entwickelt wurden, sind auch in TYPO3 CMS 6.2 lauffähig, ohne dass Modifikationen notwendig sind (oder nur geringe Anpassungen).
- TemplaVoilà ist in TYPO3 6.2 nicht mehr lauffähig **→ falsch!**
Die Wahrheit ist, dass die Entwicklergemeinschaft an einer Version von TemplaVoilà arbeitet, die auch mit TYPO3 CMS 6.2 kompatibel ist. Allerdings wird TemplaVoilà nicht weiterentwickelt und TYPO3 Integriatoren sollten für zukünftige Projekte Alternativen in Erwägung ziehen.

Mythen über TYPO3 CMS 6.2

- `tslib_pibase`-Extensions funktionieren nicht mehr → **falsch!**
Die Wahrheit ist, dass die Klasse `tslib_pibase` auch in TYPO3 6.2 existiert, aber aufgrund der zuvor erwähnten Namespace-Konvention einen neuen Namen trägt: `\TYPO3\CMS\Frontend\Plugin\AbstractPlugin`.
Ein Klassenalias stellt sicher, dass auch der bisherige Name verwendet werden kann (Abwärtskompatibilität).
- DAM Records können nicht nach FAL migriert werden → **falsch!**
Fakt ist, dass DAM nicht von TYPO3 6.x unterstützt wird. Allerdings bietet FAL eine API, die dazu bestimmt ist, sämtliche DAM-Funktionalität abzubilden. Es gibt außerdem eine [DAM-to-FAL Migration Extension](#).

Mythen über TYPO3 CMS 6.2

- Upgrade von 4.5 zu 6.2 mit einem Upgrade-Wizard → **falsch!**
Gerüchten zufolge soll das "Smooth Migration" Projekt einen Upgrade-Wizard bereit stellen, der eine TYPO3 Version 4.5 automatisch zu 6.2 aktualisiert. Die Wahrheit ist, dass das Projekt das Ziel verfolgt, Information und Dokumentationen zu liefern, um TYPO3 Integriatoren bei dem Migrationsprozess zu unterstützen.
- TYPO3 6.2 benötigt wesentlich bessere Hardware → **falsch!**
Gerüchten zufolge soll TYPO3 6.2 zehn Mal langsamer als 4.5 sein. Die Wahrheit ist, dass die Performance in den meisten Fällen identisch ist und sich die [Mindestanforderungen](#) von TYPO3 CMS kaum geändert haben. Allerdings sollten Systemadministratoren einen Hardware-Upgrade generell in Erwägung ziehen, da TYPO3 4.5 bereits im Januar 2011 veröffentlicht wurde - also vor über 3 Jahren!

Kapitel 11: Quellen und Autoren

Quellen und Autoren

Quellennachweis

TYPO3 News:

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

Release Notes:

- http://wiki.typo3.org/TYPO3_6.2
- <http://typo3.org/download/release-notes/typo3-6-2-release-notes/>
- NEWS.txt und ChangeLog

TYPO3 Git Repositories:

- <https://git.typo3.org/TYPO3v4/Core.git>
- <https://git.typo3.org/TYPO3v4/CoreProjects/MVC/extbase.git>
- <https://git.typo3.org/TYPO3v4/CoreProjects/MVC/fluid.git>

Quellen und Autoren

Quellennachweis

TYPO3 Bug-/Issuetracker:

- <http://forge.typo3.org/projects/typo3v4-core/issues>

Weitere Quellen:

- Responsive Image Community Group
<http://responsiveimages.org>
- Package Manager (Blueprint)
<http://wiki.typo3.org/Blueprints/Packagemanager>
- Normalize.css
<http://nicolas.github.io/normalize.css/>
- PHP Logging standard PSR-3
<http://www.php-fig.org/psr/psr-3/>
- "LTS Smooth Migration" Projekt
<http://forge.typo3.org/projects/typo3cms-smoothmigration>
- TYPO3 CMS 4.5 zu 6.2 Upgrade Report
<https://github.com/nxpthx/typo3-upgradereport>

Quellen und Autoren

TYP03 CMS What's New Slides:

Patrick Lobacher (Recherche und Informationsdokumentation)
Michael Schams (Englische/deutsche Version und Project Leader)

Übersetzungen von:

Andrey Aksenov, Paul Blondiaux, Sergio Catala,
Philippe Hérault, Sinisa Mitrovic, Michel Mix,
Roberto Torresani, Ric van Westhreenen, Christiaan Wiesenekker

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

Lizensiert unter Creative Commons BY-NC-SA 3.0

