

TYPO3 CMS 7.5 – Τι Νέο Υπάρχει

Σύνοψη των νέων λειτουργιών, αλλαγών και βελτιώσεων

Δημιουργήθηκε από:
Patrick Lobacher and Michael Schams
Μετάφραση στα Ελληνικά από:
Angeliki Plati

TYPO3 CMS 7.5 - Τι Νέο Υπάρχει

Σύνοψη κεφαλαίου

Εισαγωγή

Διεπαφή Χρήστη Backend

TSconfig & TypoScript

Ριζικές Αλλαγές

Extbase & Fluid

Κατηρηγμένες/Αφηρημένες Μέθοδοι

Πηγές και Συγγραφείς

Εισαγωγή Τα Γεγονότα

TYPO3 CMS 7.5 - Τα Γεγονότα

- Ημερομηνία Έκδοσης: 29 Σεπτεμβρίου 2015
- Τύπος Δημοσίευσης: **Sprint Release**
- Όραμα: Αγκαλιάζουμε, Καινοτομούμε, Παραδίδουμε
- Κύρια Εστίαση: Οριστικοποίηση



Εισαγωγή

Προδιαγραφές Συστήματος

- PHP*: v5.5.0 - v5.6.x
- MySQL: v5.5.x - v5.6.x (no strict mode)
- Χωρητ. Δίσκου: τουλάχιστον 200 MB
- Ρυθμίσεις PHP:
 - `memory_limit` \geq 128M
 - `max_execution_time` \geq 240s
 - η επιλογή `-disable-ipv6` δεν πρέπει να χρησιμοποιείται
- Για το **backend** απαιτείται η χρήση **IE \geq 9** ή οποιουδήποτε άλλου σύγχρονου **browser**

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

Εισαγωγή

Χρονοδιάγραμμα Ανάπτυξης και Δημοσίευσης



Εισαγωγή

TYPO3 CMS - Πλάνο

Εκτιμώμενες ημερομηνίες δημοσίευσης και ο πρωταρχικός τους στόχος:

- v7.0 02/Δεκ/2014 Αναμόρφωση του Backend No. 1
- v7.1 24/Φεβ/2015 Core Cleanup & Streamlining
- v7.2 28/Απρ//2015 Frontend
- v7.3 09/Ιουν/2015 Package Ecosystem, Composer and Extension Handling
- v7.4 04/Αυγ/2015 Αναμόρφωση του Backend No. 2
- v7.5 29/Σεπ/2015 Οριστικοποίηση
- v7.6 xx/xxx/2015 **TYPO3 CMS 7 LTS** (Long Term Έκδοση)

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

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

Εγκατάσταση

- Επίσημη διαδικασία εγκατάστασης σε **Linux/Mac OS X** (**DocumentRoot** για παράδειγμα `/var/www/site/htdocs`):

```
$ cd /var/www/site
$ wget --content-disposition get.typo3.org/7.5
$ tar xzf typo3_src-7.5.0.tar.gz
$ cd htdocs
$ ln -s ../typo3_src-7.5.0 typo3_src
$ ln -s typo3_src/index.php
$ ln -s typo3_src/typo3
$ touch FIRST_INSTALL
```

- Συμβολικοί σύνδεσμοι για **Microsoft Windows**:
 - Χρησιμοποιήστε `junction` για Windows XP/2000
 - Χρησιμοποιήστε `mlink` για Windows Vista and Windows 7

Αναβάθμιση σε **TYPO3 CMS 7.x**

- Η αναβάθμιση είναι δυνατή μόνο από **TYPO3 CMS 6.2 LTS**
- Η έκδοση **TYPO3 CMS < 6.2** πρέπει να αναβαθμιστεί πρώτα σε **TYPO3 CMS 6.2 LTS**
- Οδηγίες αναβάθμισης:
http://wiki.typo3.org/Upgrade#Upgrading_to_7.5
- Επίσημος οδηγός **TYPO3 Installation and Upgrading**:
<http://docs.typo3.org/typo3cms/InstallationGuide>
- Γενική προσέγγιση:
 - Ελέγξτε τις ελάχιστες απαιτήσεις συστήματος (PHP, MySQL, etc.)
 - Κάνετε ανασκόπηση του **deprecation_*.log** στην παλιά υπόσταση της **TYPO3**
 - Ανανεώστε όλες τις **extensions** στην τελευταία τους έκδοση

Αναβάθμιση σε **TYPO3 CMS 7.x**

- Γενική προσέγγιση - Συνέχεια:
 - Κάνετε **deploy** τα νέα πηγαία αρχεία και τρέξτε τον →Οδηγό Αναβάθμισης
 - Κάνετε ανασκόπηση του **startup module** για τους χρήστες **backend** (προαιρετικά)

Κεφάλαιο 1:

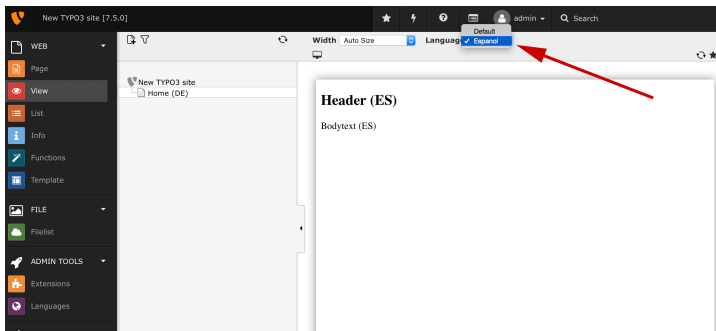
Διεπαφή Χρήστη **Backend**

Διεπαφή Χρήστη **Backend**

Επιλογέας Γλώσσας στο Δομοστοιχείο **Pageview**

Η **WEB->View** περιέχει ένα **dropdown** επιλογής γλώσσας για την προσεπισκόπηση σελίδας.

(μπορεί να απενεργοποιηθεί με `mod.SHARED.view.disableLanguageSelector = 1`)



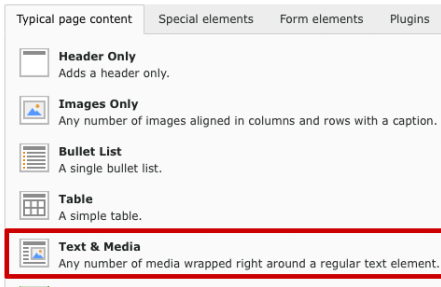
Διεπαφή Χρήστη **Backend**

Το Στοιχείο Περιεχομένου `textmedia`

Ένα νέο Στοιχείο Περιεχομένου **Text & Media** συνδυάζει τα στοιχεία `text`, `image` και `textpic`.

New content element

Please select the type of page content you wish to create:



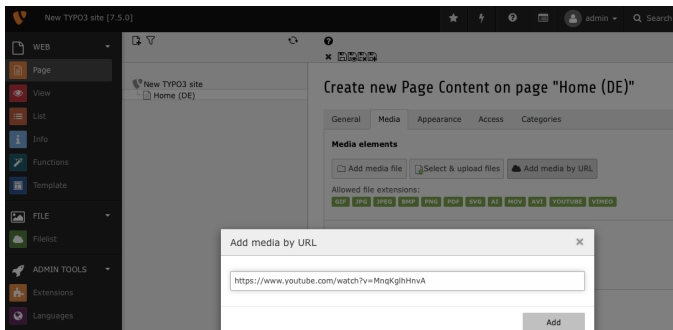
The screenshot shows a selection interface for creating new content elements. It features four tabs: 'Typical page content', 'Special elements', 'Form elements', and 'Plugins'. Under the 'Typical page content' tab, five options are listed:

- Header Only**: Adds a header only.
- Images Only**: Any number of images aligned in columns and rows with a caption.
- Bullet List**: A single bullet list.
- Table**: A simple table.
- Text & Media**: Any number of media wrapped right around a regular text element. (This option is highlighted with a red border in the image.)

Διεπαφή Χρήστη **Backend**

Αρχεία **YouTube** και **Vimeo**

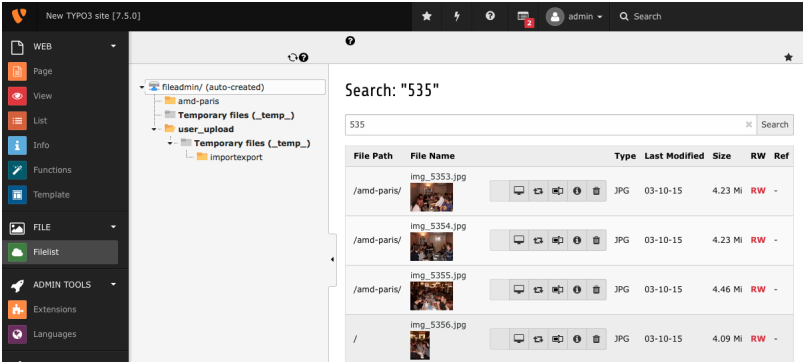
Το Στοιχείο Περιεχομένου **Text & Media** επιτρέπει στους **editors** να συμπεριλαμβάνουν εξωτερικά αρχεία **YouTube** και **Vimeo**, καθώς και τοπικά αρχεία.



Διεπαφή Χρήστη Backend

Αναζήτηση στο Δομοστοιχείο Filelist

Το δομοστοιχείο Filelist περιέχει τώρα αναζήτηση με βάση το όνομα αρχείου (αναδρομικά από τον τρέχοντα φάκελο).



The screenshot shows the TYPO3 Filelist interface. The top navigation bar includes the site name 'New TYPO3 site [7.5.0]', a search icon, and the user 'admin'. The left sidebar contains various menu items: WEB, Page, View, List, Info, Functions, Template, FILE (with Filelist selected), and ADMIN TOOLS (with Extensions and Languages). The main content area shows a file tree on the left with 'fileadmin/ (auto-created)' expanded to show 'amd-paris', 'Temporary files (_temp_)', and 'user_upload'. The 'user_upload' folder is expanded to show 'Temporary files (_temp_)' and 'importexport'. The main area displays a search for '535' with a search input field containing '535' and a search button. Below the search bar is a table of files:

File Path	File Name	Type	Last Modified	Size	RW	Ref
/amd-paris/	img_5353.jpg	JPG	03-10-15	4.23 Mi	RW	-
/amd-paris/	img_5354.jpg	JPG	03-10-15	4.23 Mi	RW	-
/amd-paris/	img_5355.jpg	JPG	03-10-15	4.46 Mi	RW	-
/	img_5356.jpg	JPG	03-10-15	4.09 Mi	RW	-

Κεφάλαιο 2: **Tsconfig & TypoScript**

TSconfig & TypoScript

Συνθήκες για Συμπεριλήψεις TypoScript

- Το INCLUDE_TYPOSCRIPT έχει τώρα μία επιπλέον (προαιρετική) ιδιότητα **condition**, η οποία συμπεριλαμβάνει το αρχείο/ευρετήριο, μόνο εάν η συνθήκη ικανοποιείται

```
// symperilamvanei TypoScript, mono ean o xrhsths einai syndedemenos:  
\latintext <INCLUDE_TYPOSCRIPT: source="FILE:EXT:my_extension/Configuration/TypoScript/feuser.ts"  
condition="[loginUser = *]">
```

```
// symperilamvanei TypoScript analoga me to perioxomeno ths efarmoghs:  
<INCLUDE_TYPOSCRIPT: source="FILE:EXT:my_extension/Configuration/TypoScript/staging.ts"  
condition="applicationContext = /~Production\\Staging\\Server\\d+$/">
```

TScnfig & TypoScript

Επιλογή-TCA: Εμφάνιση **Offset** Ημερομηνίας

- Η επιλογή TCA `disableAgeDisplay` απενεργοποιεί την εμφάνιση της ηλικίας

(για παράδειγμα: 2015-08-30 (-27 ημέρες))

```
$GLOBALS['TCA']['tt_content']['columns']['date']['config']['disableAgeDisplay'] = true;
```

- Ως προϋπόθεση, το `type` του πεδίου πρέπει να είναι `input` και το `eval` πρέπει να έχει τεθεί σε `date`

TSconfig & TypeScript

Ενσωματωμένα Αρχεία Ετικετών Γλώσσας με **TypeScript (1)**

- Τα **XLF** αρχεία γλώσσας μπορούν να διαβαστούν και να τροποποιηθούν σε έναν ενσωματωμένο πίνακα
- Αυτό ενεργοποιεί την πρόσβαση στις ετικέτες γλωσσών με **JavaScript** για παράδειγμα
- Υποστηρίζονται οι 3 ακόλουθες προαιρετικές παράμετροι:
 - **selectionPrefix**:
μόνο κλειδιά ετικετών που ξεκινούν με αυτό το πρόθεμα θα συμπεριληφθούν
 - **stripFromSelectionName**:
η γραμματοσειρά που θα αφαιρεθεί από οποιοδήποτε συμπεριληφθέν κλειδί ετικέτας

TSconfig & TypeScript

Ενσωματωμένα Αρχεία Ετικετών Γλώσσας με **TypeScript** (2)

- **errorMode:**

η κατάσταση σφάλματος αν το αρχείο δε μπόρεσε να βρεθεί:

0: syslog entry (default), 1: ignore, 3: throw an exception

TSconfig & TypeScript

Ενσωματωμένα Αρχεία Ετικετών Γλώσσας με **TypeScript (3)**

■ Παράδειγμα:

```
page = PAGE
page.inlineLanguageLabelFiles {
  someLabels = EXT:myExt/Resources/Private/Language/locallang.xlf
  someLabels.selectionPrefix = idPrefix
  someLabels.stripFromSelectionName = strip_me
  someLabels.errorMode = 2
}
```

■ Output:

```
<script type="text/javascript">
/**]
  var TYPO3 = TYPO3 || {};
  TYPO3.lang = {"firstLabel":[{"source":"first Label","target":"erstes Label"}],
"secondLabel":[{"source":"second Label","target":"zweites Label"}]};
/*]]&gt;*/
&lt;/script&gt;</pre></div><div data-bbox="22 930 277 958" data-label="Page-Footer"><p>TYPO3 CMS 7.5 - Τι Νέο Υπάρχει</p></div><div data-bbox="805 882 970 946" data-label="Page-Footer"><img alt="TYPO3 logo" data-bbox="805 882 970 946"/>The logo for TYPO3, featuring a stylized orange and white 'T' icon to the left of the text 'TYPO3' in a bold, sans-serif font.</div>
```

TScnfig & TypoScript

Προεπισκόπηση του Χώρου Εργασίας με TScnfig

- Το TYPO3 CMS παράγει συνδέσμους προεπισκόπησης μόνο για πίνακες `tt_content`, `pages` και `pages_language_overlay` εκ προεπιλογής
- Αυτό μπορεί τώρα να ρυθμιστεί χρησιμοποιώντας PageTScnfig:

```
# xrhsimopihse th sel. 123 gia proepiskophsh eggrafwn workspaces (genika)
options.workspaces.previewPageId = 123
```

```
# xrhsimopihse to pid kathe eggrafhs gia proepiskophsh (genika)
options.workspaces.previewPageId = field:pid
```

```
# xrhsimopihse th sel. 123 gia proepiskophsh eggrafwn workspaces (gia pinaka tx_myext_table)
options.workspaces.previewPageId.tx_myext_table = 123
```

```
# xrhsimopihse to pid kathe eggrafhs gia proepiskophsh (gia pinaka tx_myext_table)
options.workspaces.previewPageId.tx_myext_table = field:pid
```

TSconfig & TypoScript

Ποιότητα Εικόνας της `sourceCollection`

- Η ποιότητα εικόνας κάθε μίας καταχώρισης της `sourceCollection` **mpore'i t'wra na diamorfwje'i**
- Αυτή η ρύθμιση έχει προτεραιότητα έναντι της διαμόρφωσης στο **Install Tool**
(όπως είναι αποθηκευμένη στο αρχείο `LocalConfiguration.php`)
- Παράδειγμα:

```
# gia mikres eikones retina
tt_content.image.20.1.sourceCollection.smallRetina.quality = 80

# gia megales eikones retina
tt_content.image.20.1.sourceCollection.largeRetina.quality = 65
```

TSconfig & TypeScript

Καταμέτρηση Στοιχείων σε μία Λίστα

- Μία νέα ιδιότητα `returnCount` έχει προστεθεί στην ιδιότητα `stdWrap split`
- Αυτό επιτρέπει την καταμέτρηση του αριθμού των στοιχείων σε μία λίστα διαχωρισμένη με κόμματα
- Ο ακόλουθος κώδικας επιστρέφει 9 για παράδειγμα:

```
1 = TEXT
1 {
  value = x,y,z,1,2,3,a,b,c
  split.token = ,
  split.returnCount = 1
}
```


TScnfig & TypoScript

Διαχείριση των Backend Layouts (1)

- Η διαχείριση των backend layouts έχει απλοποιηθεί για το frontend
- Η νέα επιλογή pagelayout μπορεί να χρησιμοποιηθεί στην TypoScript
- Παράδειγμα:

```
page.10 = FLUIDTEMPLATE
page.10 {
    file.stdWrap.cObject = CASE
    file.stdWrap.cObject {
        key.data = pagelayout
        default = TEXT
        default.value = EXT:sitepackage/Resources/Private/Templates/Home.html
        3 = TEXT
        3.value = EXT:sitepackage/Resources/Private/Templates/1-col.html
        4 = TEXT
        4.value = EXT:sitepackage/Resources/Private/Templates/2-col.html
    }
}
```

(η συνέχεια στην επόμενη σελίδα)

Διαχείριση των Backend Layouts (2)

- ...όπου `key.data = pagelayout` αντικαθιστά τον ακόλουθο κώδικα:

```
field = backend_layout
ifEmpty.data = levelfield:-2,backend_layout_next_level,slide
ifEmpty.ifEmpty = default
```

TSconfig & TypoScript

Διάφορα

- Η λειτουργία `stdWrap bytes` έχει προστεθεί στο **TYPO3 CMS 7.4**
- Η δυνατότητα να τεθεί η `base` έχει προστεθεί στο **TYPO3 CMS 7.5**, η οποία επιτρέπει να προσδιοριστεί αν θα χρησιμοποιηθεί μία βάση του 1000 ή του 1024 για τον υπολογισμό

```
bytes.labels = " | K| M| G"  
bytes.base = 1000
```

Κεφάλαιο 3: Ριζικές Αλλαγές

Στοιχεία Περιεχομένου βασισμένα σε **Fluid (1)**

- Η νέα επέκταση συστήματος **Fluid-based Content Elements** έχει υλοποιηθεί
- Τα **Fluid templates** χρησιμοποιούνται για την απεικόνιση στοιχείων περιεχομένου και όχι η **TypoScript**
- Θα μπορούσε να είναι μια εναλλακτική για την **CSS Styled Content** κάποια στιγμή στο μέλλον
- Συμπεριλάβετε τα ακόλουθα στατικά **templates** προκειμένου να χρησιμοποιήσετε αυτή τη λειτουργία:
 - Στοιχεία Περιεχομένου (`fluid_styled_content`)
 - Στοιχεία Περιεχομένου CSS (προαιρετικά) (`fluid_styled_content`)

Στοιχεία Περιεχομένου βασισμένα σε **Fluid (2)**

- Επιπρόσθετα, το ακόλουθο **PageTSconfig template** πρέπει να προστεθεί στις ιδιότητες της σελίδας:
Fluid-based Content Elements (fluid_styled_content)
- Αντικαταστήστε τα προεπιλεγμένα **templates** προσθέτοντας τα δικά σας μονοπάτια στο **TypoScript setup**:

```
lib.fluidContent.templateRootPaths.50 = EXT:site_example/Resources/Private/Templates/  
lib.fluidContent.partialRootPaths.50 = EXT:site_example/Resources/Private/Partials/  
lib.fluidContent.layoutRootPaths.50 = EXT:site_example/Resources/Private/Layouts/
```

Στοιχεία Περιεχομένου βασισμένα σε **Fluid (3)**

- Μετακινήστε από την **CSS Styled Content** στην **Fluid-based Content Elements**:
 - Απεγκαταστήστε την επέκταση `css_styled_content`
 - εγκαταστήστε την επέκταση `fluid_styled_content`
 - Χρησιμοποιήστε τον **Upgrade Wizard** στο **Install Tool** για να μετακινήσετε τα Στοιχεία Περιεχομένου `text`, `image` και `textpic` στα `textmedia`

Σημείωση: Η **Fluid-based Content Elements** βρίσκεται ακόμα σε αρχικό στάδιο και ριζικές αλλαγές είναι δυνατό να συμβούν μέχρι το **TYPO3 CMS 7 LTS**. Επίσης, μερικά **conflicts** σχετικά με την **CSS Styled Content** πιθανόν να υπάρχουν ακόμα.

Μέθοδος **SELECTmmQuery**

- Η νέα μέθοδος **SELECT_mm_query** έχει προστεθεί στην κλάση **DatabaseConnection**
- Έχει εξαχθεί από το **exec_SELECT_mm_query** για να διαχωρίσει το «χτίσιμο» και την εκτέλεση των **M:M queries**.
- Αυτό καθιστά δυνατό το «χτίσιμο» **query** στο αφηρημένο επίπεδο της βάσης δεδομένων

```
$query = SELECT_mm_query('*', 'table1', 'table1_table2_mm', 'table2', 'AND table1.uid = 1',  
'', 'table1.title DESC');
```


Βελτιστοποίηση Πινάκων της Βάσης Δεδομένων σε **MySQL**

- Νέο χρονοπρογραμματισμένο **task** για το «τρέξιμο» της εντολής **MySQL OPTIMIZE TABLE** σε επιλεγμένους πίνακες της Β.Δ.
- Αυτή η εντολή αναδιοργανώνει τον φυσικό χώρο αποθήκευσης των δεδομένων πινάκων και των σχετικών δεδομένων ευρετηρίου για να μειώσει το χώρο αποθήκ. και να βελτιώσει την αποδοτικότητα **I/O**
- Υποστηρίζονται οι ακόλουθοι τύποι πινάκων:
MyISAM, InnoDB και ARCHIVE
- Η χρήση αυτού του **task** με **DBAL** και άλλα **DBMS** δεν υποστηρίζεται λόγω του ότι οι εντολές που χρησιμοποιούνται είναι μόνο για **MySQL**

Σημείωση: η βελτιστοποίηση πινάκων είναι μία εντατική **I/O** διεργασία. Επίσης, στην **MySQL < 5.6.17** η διεργασία κλειδώνει τους πίνακες για όσο «τρέχει», το οποίο μπορεί να επηρεάσει το **website**.

Χειρισμός των **Online** Μέσων (1)

- Τα εξωτερικά μέσα (**online media**) υποστηρίζονται εκ προεπιλογής τώρα
- Ως παραδείγματα, η υποστήριξη των βίντεο **YouTube** και **Vimeo** έχει υλοποιηθεί στον πυρήνα
- Οι πόροι μπορούν να προστεθούν ως **URLs** χρησιμοποιώντας το στοιχείο περιεχομένου **Text & Media**, για παράδειγμα
- Η αντίστοιχη **helper** κλάση φέρνει τα **data** και παρέχει μία εικόνα που θα χρησιμοποιηθεί ως προεπισκόπηση αν είναι διαθέσιμη

Ριζικές Αλλαγές

Χειρισμός των **Online** Μέσων (2)

Οι ακόλουθες συντάξεις **URL** είναι δυνατές:

YouTube:

`youtu.be/<code>`

`www.youtube.com/watch?v=<code>`

`www.youtube.com/v/<code>`

`www.youtube-nocookie.com/v/<code>`

`www.youtube.com/embed/<code>`

Vimeo:

`vimeo.com/<code>`

`player.vimeo.com/video/<code>`

Χειρισμός των **Online** Μέσων (3)

- Η πρόσβαση στους πόρους χρησιμοποιώντας **Fluid** μπορεί να επιτευχθεί ως εξής:

```
<!-- enable js api and set no-cookie support for YouTube videos -->  
<f:media file="{file}" additionalConfig="{enablejsapi:1, 'no-cookie': true}" ></f:media>
```

```
<!-- show title and uploader for YouTube and Vimeo before video starts playing -->  
<f:media file="{file}" additionalConfig="{showinfo:1}" ></f:media>
```

- Επιλογές προσαρμοσμένης διαμόρφωσης για βίντεο **YouTube**:
autoplay, controls, loop, enablejsapi, showinfo, no-cookie
- Επιλογές προσαρμοσμένης διαμόρφωσης για βίντεο **Vimeo**:
autoplay, loop, showinfo

Χειρισμός των **Online** Μέσων (4)

- Για να εγγραφείτε στην δική σας υπηρεσία **online media**, θα χρειαστείτε `OnlineMediaHelper` κλάση που υλοποιεί την `OnlineMediaHelperInterface` και μια κλάση `FileRenderer` που υλοποιεί την `FileRendererInterface`

```
// register your own online video service (the used key is also the bind file extension name)
$GLOBALS['TYPO3_CONF_VARS']['SYS']['OnlineMediaHelpers']['myvideo'] =
    \MyCompany\Myextension\Helpers\MyVideoHelper::class;

$rendererRegistry = \TYPO3\CMS\Core\Resource\Rendering\RendererRegistry::getInstance();
$rendererRegistry->registerRendererClass(
    \MyCompany\Myextension\Rendering\MyVideoRenderer::class
);

// register an custom mime-type for your videos
$GLOBALS['TYPO3_CONF_VARS']['SYS']['FileInfo']['fileExtensionToMimeType']['myvideo'] =
    'video/myvideo';

// register your custom file extension as allowed media file
$GLOBALS['TYPO3_CONF_VARS']['SYS']['mediafile_ext'] .= ',myvideo';
```

Δρομολόγηση στο **Backend**

- Μία νέα συνιστώσα δρομολόγησης έχει προστεθεί στο **TYPO3 backend** η οποία διαχειρίζεται τη διευθέτηση των διάφορων κλήσεων/δομοστοιχείων μέσα στο **TYPO3 CMS**
- Οι διαδρομές μπορούν να οριστούν στην ακόλουθη κλάση: `Configuration/Backend/Routes.php`

```
return [  
    'myRouteIdentifier' => [  
        'path' => '/document/edit',  
        'controller' => Acme\MyExtension\Controller\MyExampleController::class . '::methodToCall'  
    ]  
];
```

- Η κληθείσα μέθοδος περιέχει αντικείμενα αιτήματος και απόκρισης συμβατά με **PSR-7**:

```
public function methodToCall(ServerRequestInterface $request, ResponseInterface $response) {  
    ...  
}
```

Ριζικές Αλλαγές

Ορισμός Αυτόματης Φόρτωσης στο `ext_emconf.php`

- Οι επεκτάσεις μπορούν τώρα να παρέχουν έναν ή περισσότερους **PSR-4** ορισμούς στο αρχείο `ext_emconf.php`
- Αυτό ήταν ήδη δυνατόν στο `composer.json`, αλλά με αυτό το νέο χαρακτηριστικό, οι προγραμματιστές επεκτάσεων δε χρειάζεται πια να παρέχουν ένα αρχείο `composer` για αυτό το σκοπό

```
$EM_CONF[$_EXTKEY] = array (  
    'title' => 'Extension Skeleton for TYPO3 CMS 7',  
    ...  
    'autoload' =>  
        array(  
            'psr-4' => array(  
                'Helhum\\ExtScaffold\\' => 'Classes'  
            )  
        )  
    );
```

(προτεινόμενος τρόπος για την εγγραφή κλάσεων στην TYPO3)

Νέος Παραγωγός Εικονιδίων (1)

- Η λογική της χρήσης εικονιδίων, μεγεθών εικονιδίων και **overlays** (επικαλύψεων) εικονιδίων είναι τώρα συγκεντρωμένη μέσα στην κλάση **IconFactory**
- Ο νέος παραγωγός θα αντικαταστήσει σταδιακά το **API** εικονιδίων
- Όλα τα εικονίδια του πυρήνα θα εγγραφούν άμεσα στην κλάση **IconRegistry**

Νέος Παραγωγός Εικονιδίων (2)

- Οι επεκτάσεις πρέπει να χρησιμοποιούν την `IconRegistry::registerIcon()` για να παρακάμψουν υπάρχοντα εικονίδια ή για να προσθέσουν επιπλέον εικονίδια στον παραγωγό εικονιδίων:

```
IconRegistry::registerIcon(  
    $identifier,  
    $iconProviderClassName,  
    array $options = array()  
);
```

Νέος Παραγωγός Εικονιδίων (3)

- Ο πυρήνας του **TYPO3 CMS** υλοποιεί 3 κλάσεις παροχής εικονιδίων: **BitmapIconProvider**, **FontawesomeIconProvider** και **SvgIconProvider**

- Παράδειγμα χρήσης:

```
$iconFactory = GeneralUtility::makeInstance(IconFactory::class);  
$iconFactory->getIcon(  
    $identifier,  
    Icon::SIZE_SMALL,  
    $overlay,  
    IconState::cast(IconState::STATE_DEFAULT)  
)->render();
```

- Έγκυρες τιμές για **Icon::SIZE_...** είναι: **SIZE_SMALL**, **SIZE_DEFAULT** και **SIZE_LARGE**
- Έγκυρες τιμές για **Icon::STATE_...** είναι: **STATE_DEFAULT** και **STATE_DISABLED**

Νέος Παραγωγός Εικονιδίων (4)

- Ο πυρήνας του **TYPO3 CMS** παρέχει έναν **Fluid ViewHelper** ο οποίος διευκολύνει τη χρήση εικονιδίων μέσα σε μία **Fluid view**:

```
{namespace core = TYPO3\CMS\Core\ViewHelpers}

<core:icon identifier="my-icon-identifier"></core:icon>

<!-- use the "small" size if none given -->
<core:icon identifier="my-icon-identifier"></core:icon>
<core:icon identifier="my-icon-identifier" size="large"></core:icon>
<core:icon identifier="my-icon-identifier" overlay="overlay-identifier"></core:icon>

<core:icon identifier="my-icon-identifier" size="default" overlay="overlay-identifier">
</core:icon>

<core:icon identifier="my-icon-identifier" size="large" overlay="overlay-identifier">
</core:icon>
```

Άγκιστρα (**hooks**) και Σήματα

- Ένα νέο σήμα έχει προστεθεί στον **LinkValidator**, το οποίο επιτρέπει επιπλέον επεξεργασία μετά την οριστικοποίηση μιας συγκεκριμένης εγγραφής (π.χ. λήψη δεδομένων περιεχομένου από διαμόρφωση **plugin** σε εγγραφή)
- Εγγραφή του σήματος στο αρχείο `ext_localconf.php`:

```
$signalSlotDispatcher = \TYPO3\CMS\Core\Utility\GeneralUtility::makeInstance(  
    \TYPO3\CMS\Extbase\SignalSlot\Dispatcher::class  
);  
  
$signalSlotDispatcher->connect(  
    \TYPO3\CMS\Linkvalidator\LinkAnalyzer::class,  
    'beforeAnalyzeRecord',  
    \Vendor\Package\Slots\RecordAnalyzerSlot::class,  
    'beforeAnalyzeRecord'  
);
```

Η **JumpUrl** ως Επέκταση Συστήματος (1)

- Η παραγωγή και ο χειρισμός των **JumpURLs** έχει μετακινηθεί σε μία νέα επέκταση συστήματος `jumpurl`
- Νέα άγκιστρα έχουν εισαχθεί, τα οποία επιτρέπουν την παραγωγή και τον χειρισμό ειδικών (**custom**) **URL** (βλ. επόμενη διαφάνεια)

BREAKING CHANGE!

Η **JumpUrl** ως Επέκταση Συστήματος (2)

- Άγκιστρο 1: χειρισμός των **URLs** κατά την παραγωγή συνδέσμων

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']['urlProcessing']['urlHandlers']  
    ['myext_myidentifier']['handler'] = \Company\MyExt\MyUrlHandler::class;  
  
// class needs to implement the UrlHandlerInterface:  
class MyUrlHandler implements \TYPO3\CMS\Frontend\Http\UrlHandlerInterface {  
    ...  
}
```

- Άγκιστρο 2: χειρισμός των **URLs**

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']['urlProcessing']['urlProcessors']  
    ['myext_myidentifier']['processor'] = \Company\MyExt\MyUrlProcessor::class;  
  
// class needs to implement the UrlProcessorInterface:  
class MyUrlProcessor implements \TYPO3\CMS\Frontend\Http\UrlProcessorInterface {  
    ...  
}
```

Διεπαφή Γραμμής Εντολών (CLI)

- Η κλήση του `typo3/cli_dispatch.phpsh` μέσω της γραμμής εντολών δείχνει τώρα ένα έγχρωμο μήνυμα, αν ένα άκυρο ή μη CLI κλειδί δωθεί ως πρώτη παράμετρος
- Οι ελεγκτές εντολών του **Extbase** μπορούν τώρα να βρίσκονται σε αυθαίρετους υποφακέλους μέσα στον φάκελο **Command**
- Παράδειγμα:

Ο ελεγκτής στο αρχείο:

```
my_ext/Classes/Command/Hello/WorldCommandController.php
```

... μπορεί να κληθεί μέσω CLI:

```
typo3/cli_dispatch.sh extbase my_ext:hello:world <arguments>
```

Διάφορα (1)

- Τα κουμπιά μετακίνησης του **TCA** τύπου **group** μπορούν τώρα να απενεργοποιηθούν κατηγορηματικά χρησιμοποιώντας την επιλογή `hideMoveIcons = TRUE`
- Η μέθοδος `makeCategorizable` έχει επεκταθεί με μία νέα παράμετρο `override` για να θέσει μία νέα διαμόρφωση κατηγορίας για έναν ήδη εγγεγραμμένο συνδυασμό πίνακα/πεδίου
- Παράδειγμα:

```
\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::makeCategorizable(  
    'css_styled_content', 'tt_content', 'categories', array(), TRUE  
);
```

Η τελευταία παράμετρος (εδώ: `TRUE`) επιβάλλει την παράκαμψη (η προεπιλεγμένη τιμή είναι `FALSE`).

Ριζικές Αλλαγές

Διάφορα (2)

- Μία νέα συνάρτηση παράγει ένα μοναδικό **ID**

```
$uniqueId = \TYPO3\CMS\Core\Utility\StringUtility::getUniqueId('Prefix');
```

- Ο τύπος αρχείου **.typoscript** έχει προστεθεί στη λίστα των έγκυρων αρχείων τύπου **plain text**

- Νέα επιλογή διαμόρφωσης καθορίζει επεκτάσεις αρχείων από αρχεία μέσω των τεστ

```
$GLOBALS['TYPO3_CONF_VARS']['SYS']['mediafile_ext'] =  
'gif,jpg,jpeg,bmp,png,pdf,svg,ai,mov,avi';
```

BREAKING CHANGE!

Κεφάλαιο 4: Extbase & Fluid

Extbase & Fluid

Φιλτράρισμα σύμφωνα με την Κρισιμότητα για την `FlashMessageQueue`

- Στο TYPO3 CMS < 7.5, όλα τα μηνύματα της `FlashMessageQueue` μπορούν μόνο να προσκομιστούν ή/και να αφαιρεθούν
- Στο TYPO3 CMS >= 7.5, αυτό μπορεί να γίνει για έναν συγκεκριμένο βαθμό κρισιμότητας:

```
FlashMessageQueue::getAllMessages($severity);  
FlashMessageQueue::getAllMessagesAndFlush($severity);  
FlashMessageQueue::removeAllFlashMessagesFromSession($severity);  
FlashMessageQueue::clear($severity);
```

Προσθήκη Υποστήριξης Query για between

- Υποστήριξη για `between` έχει προστεθεί στο αντικείμενο `Extbase Query`
- Δεν υπάρχει κάποιο πλεονέκτημα επίδοσης λόγω του ότι το `DBMS` μετατρέπει το `between` εσωτερικά ούτως ή άλλως: `min <= expr AND expr <= max`
- Το νέο χαρακτηριστικό του `Extbase` αναπαράγει τη συμπεριφορά του `DBMS` μέσω της κατασκευής μίας λογικής συνθήκης `AND`, και έτσι αυτό λειτουργεί σε όλα τα `DBMS`

```
$query->matching(  
    $query->between('uid', 3, 5)  
);
```

Πολλαπλές Ουρές **FlashMessage**

- Είναι τώρα δυνατό να υλοποιηθούν πολλαπλές **FlashMessageQueues**:

```
$queueIdentifier = 'myQueue';  
$this->controllerContext->getFlashMessageQueue($queueIdentifier);
```

- Πρόσβαση χρησιμοποιώντας **Fluid** λειτουργεί ως εξής:

```
<f:flashMessages queueIdentifier="myQueue" ></f:flashMessages>
```

ViewHelper Μέσων (1)

- Προκειμένου να εμφανιστεί εύκολα ένα **video**, **audio** και όλοι οι άλλοι τύποι αρχείων με μία εγγεγραμμένη κλάση **Renderer** στο **frontend**, έχει υλοποιηθεί ένας **MediaViewHelper**
- Ο **MediaViewHelper** αρχικά ελέγχει αν υπάρχει κάποιος **Renderer** για το δοθέν αρχείο - αν όχι, εμφανίζει μία εικόνα
- Παραδείγματα:

```
<code title="Image Object">  
  <f:media file="{file}" width="400" height="375" ></f:media>  
</code>
```

```
<output>  
    
</output>
```

ViewHelper Μέσων (2)

■ Παραδείγματα (συνέχεια):

```
<code title="MP4 Video Object">
  <f:media file="{file}" width="400" height="375" ></f:media>
</code>
```

```
<output>
  <video width="400" height="375" controls>
    <source src="fileadmin/user_upload/my-video.mp4" type="video/mp4">
  </video>
</output>
```

```
<code title="MP4 Video Object with loop and autoplay option set">
  <f:media file="{file}" width="400" height="375"
    additionalConfig="{loop: '1', autoplay: '1'}" ></f:media>
</code>
```

```
<output>
  <video width="400" height="375" controls loop>
    <source src="fileadmin/user_upload/my-video.mp4" type="video/mp4">
  </video>
</output>
```

Extbase & Fluid

Επέκταση Συστήματος `form` (1)

- Η επέκταση συστήματος `form` (συμπεριλαμβανομένου του προσαρμοσμένου μοντέλου δεδομένων, της λογικής του `controller`, της επικύρωσης ιδιοτήτων, των `views` και του `templating`) έχει υιοθετηθεί για την υποστήριξη της στοίβας του `Extbase/Fluid MVC`
- Αυτό επιτρέπει καλύτερη προσαρμογή και έλεγχο της παραγόμενης συμπεριφοράς και του `markup`, απλώς με την τροποποίηση των `Fluid templates` ή με την αξιοποίηση προσαρμοσμένης λογικής `view helper`
- Κάθε στοιχείο φόρμας χρησιμοποιεί το δικό του `Partial`, το οποίο μπορεί επίσης να διαμορφωθεί με την επιλογή `TypoScript partialPath = ...`

Επέκταση Συστήματος form (2)

- Υπάρχουν οι τρεις ακόλουθοι νέοι **ViewHelpers**:
 - `AggregateSelectOptionsViewHelper` (για ετικέτες `optgroup`)
 - `SelectViewHelper` (για ετικέτες `optgroup`)
 - `PlainMailViewHelper` (για την εμφάνιση `mail` απλού κειμένου)
- Επιπρόσθετα, υπάρχουν τρεις **Views**:
 - `show` (η ίδια η φόρμα)
 - `confirmation` (η σελίδα επιβεβαίωσης)
 - `postProcessor/mail` (το `email`)
- Τα μονοπάτια των **template** και η ορατότητα των πεδίων μπορεί να προσαρμοσθεί για κάθε **View** ξεχωριστά

Extbase & Fluid

Επισήμανση @cli

- Με τη χρήση του νέας επισήμανσης @cli, οι εντολές σε έναν **Extbase CommandController** μπορούν να μαρκριστούν ως **CLI-εντολές** μόνο
- Αυτές οι εντολές εξαιρούνται από την επιλογή της εντολής **scheduler**
- Τυπικές περιπτώσεις χρήσης είναι εντολές όπως `extbase:help:help` για παράδειγμα

Κεφάλαιο 5:

Κατηρηγμένες/Αφηρημένες Μέθοδοι

Κατηρηγμένες/Αφηρημένες Μέθοδοι

Μέθοδοι σχετικές με **Slash** στην **GeneralUtility**

- Οι ακόλουθες μέθοδοι της κλάσης **GeneralUtility** έχουν σημειωθεί ως κατηρηγμένες:
 - `GeneralUtility::addSlashesOnArray()`
 - `GeneralUtility::stripSlashesOnArray()`
 - `GeneralUtility::slashArray()`

Κατηρηγμένες/Αφηρημένες Μέθοδοι

Σταθερές και Μέθοδοι σχετικές με τη **CLI**

- Η λογική σχετικά με τα συνηθισμένα **script** βασισμένα σε **CLI** με την επιλογή **CLIkey** έχουν μεταφερθεί στον **cliRequestHandler** και για αυτό, η ακόλουθη μέθοδος έχει σημειωθεί ως κατηρηγμένη:

```
BackendUserAuthentication->checkCLIuser()
```

- Επιπρόσθετα, οι ακόλουθες σταθερές και καθολικές παράμετροι έχουν σημειωθεί ως κατηρηγμένες:

```
const TYPO3_cliKey  
const TYPO3_cliInclude  
$GLOBALS['MCONF']['name']  
$GLOBALS['temp_cliScriptPath']  
$GLOBALS['temp_cliKey']
```

IconUtility

- Η διεπαφή `IconUtilityOverrideResourceIconHookInterface` καθώς και η κλάση `IconUtility` θα αφαιρεθούν από το **TYPO3 CMS 8**
- Όλες οι δημόσιες μέθοδοι αυτής της κλάσης έχουν σημειωθεί ως κατηρηγμένες:

<code>IconUtility::skinImg()</code>	<code>IconUtility::getIcon()</code>
<code>IconUtility::getSpriteIcon()</code>	<code>IconUtility::getSpriteIconForFile()</code>
<code>IconUtility::getSpriteIconForRecord()</code>	<code>IconUtility::getSpriteIconForResource()</code>
<code>IconUtility::getSpriteIconClasses()</code>	

(κάποιες μέθοδοι έχουν μεταφερθεί στην `IconFactory`)

- Επιπρόσθετα, η ακόλουθη ρύθμιση του `PageTSConfig` έχει σημειωθεί ως κατηρηγμένη:

```
mod.wizards.newContentElement.wizardItems.*.elements.*.icon
```

Κατηρηγμένες/Αφηρημένες Μέθοδοι

Κατηρηγμένες Μέθοδοι του `HtmlParser`

- Η λειτουργία αντικατάστασης `marker` έχει μεταφερθεί από την `core/Classes/Html/HtmlParser.php` στη δική της κλάση `core/Classes/Service/MarkerBasedTemplateService.php`
- Οι ακόλουθες μέθοδοι μέσα στην `HtmlParser` θα αφαιρεθούν στο **TYPO3 CMS 8** και έχουν σημειωθεί ως κατηρηγμένες τώρα:

`HtmlParser::getSubpart()`

`HtmlParser::substituteSubpart()`

`HtmlParser::substituteSubpartArray()`

`HtmlParser::substituteMarker()`

`HtmlParser::substituteMarkerArray()`

`HtmlParser::substituteMarkerAndSubpartArrayRecursive()`

Κατηρηγμένες/Αφηρημένες Μέθοδοι

Η Επέκταση Συστήματος `form`

- Η επέκταση συστήματος `form` είναι τώρα βασισμένη στο `Extbase/Fluid`
- Γί αυτό όλες οι ρυθμίσεις `layout` βασισμένες σε `TypoScript` έχουν σημειωθεί ως κατηρηγμένες και η χρήση `layout` δεν συνιστάται πλέον:

```
10 = FORM
10 {
    layout {
        containerWrap = <div><elements ></elements></div>
        elementWrap = <div><element ></element></div>
    }
}
```


Κατηρηγμένες/Αφηρημένες Μέθοδοι

Κατηρηγμένοι **ViewHelpers** και Μέθοδοι

- Οι ακόλουθοι **ViewHelpers** έχουν σημειωθεί ως κατηρηγμένοι:

```
\TYPO3\CMS\Fluid\ViewHelpers\Be\Buttons\IconViewHelper  
\TYPO3\CMS\Backend\ViewHelpers\SpriteManagerIconViewHelper
```

- Οι ακόλουθες μέθοδοι έχουν σημειωθεί ως κατηρηγμένες:

```
BackendUtility::getExcludeFields()  
BackendUtility::getExplicitAuthFieldValues()  
BackendUtility::getSystemLanguages()  
BackendUtility::getRegisteredFlexForms()  
BackendUtility::exec_foreign_table_where_query()  
BackendUtility::replaceMarkersInWhereClause()
```

Κεφάλαιο 6: Πηγές και Συγγραφείς

Πηγές και Συγγραφείς

Πηγές (1)

Νέα της **TYPO3**:

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

Πληροφορίες Δημοσίευσης:

- http://wiki.typo3.org/TYPO3_CMS_7.5.0
- [INSTALL.md](#) and [ChangeLog](#)
- typo3/sysex/core/Documentation/Changelog/7.5/*

Αναφορές Προβλημάτων στην **TYPO3**:

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

TYPO3 Git Αποθήκες (**Repositories**):

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

TYPO3 CMS Διαφάνειες Τι Νέο Υπάρχει:

Patrick Lobacher

(Έρευνα, Συγκέντρωση Πληροφορίας και Έκδοση στα Γερμανικά)

Michael Schams

(Επικεφαλής Έργου και Έκδοση στα Αγγλικά)

Μεταφράσεις από:

**Andrey Aksenov, Sergio Catala, Jigal van Hemert, Michel Mix,
Sinisa Mitrovic, Angeliki Plati, Nena Jelena Radovic, Roberto Torresani**

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

Υπό την άδεια της Creative Commons BY-NC-SA 3.0

