

TYP03 CMS 7.1 – Le novità

Riassunto delle funzionalità, modifiche e aggiornamenti

Creato da:

Patrick Lobacher and Michael Schams

Traduzione italiana di:

Roberto Torresani ([roberto.torresani \(at\) typo3.org](mailto:roberto.torresani@typo3.org))

TYPO3 CMS 7.1 - Le novità

Indice delle sezioni

Introduzione

BackendUI

TSconfig & TypoScript

Modifiche rilevanti

Extbase & Fluid

Funzionalità deprecate/rimosse

Fonti e autori

Introduzione

I fatti in breve

Introduzione

TYPO3 CMS 7.1 - I fatti in breve

- Data di rilascio: 24 Febbraio 2015
- Tipo di rilascio: "Sprint Release"
- Visione: Embrace, Innovate, Deliver
- Focus principale: Pulizia del Core e ottimizzazioni



Introduzione

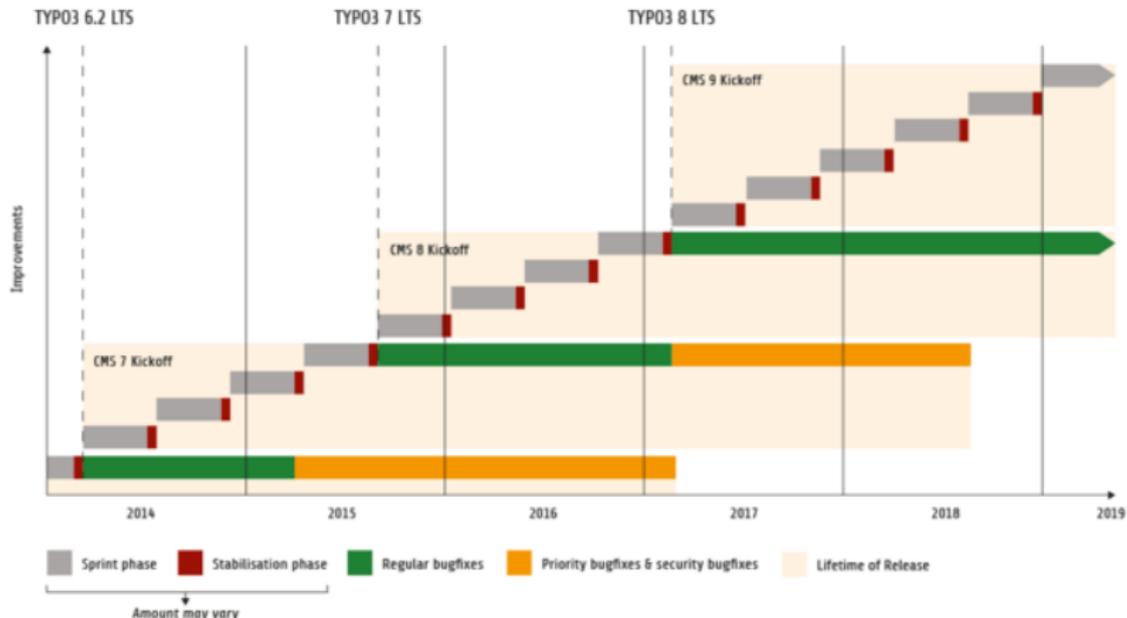
Requisiti di sistema

- PHP*: v5.5.0 - v5.6.x
- MySQL: v5.5.x - v5.6.x (no strict mode)
- Spazio disco: min 200 MB
- Impostazioni PHP:
 - `memory_limit` \geq 128M
 - `max_execution_time` \geq 240s
 - l'opzione di compilazione `-disable-ipv6` non deve essere usata
- Il Backend richiede IE \geq 9 o qualsiasi altro browser moderno

*) Altri dettagli: [Requisiti minimi PHP per TYPO3 CMS 7](#)

Introduzione

Sviluppo e tempi di rilascio



Introduzione

TYPO3 CMS Roadmap

Date di rilascio stimate e loro obiettivo principale:

- v7.0 → 02/Dec/2014 Revisione Backend Vol 1
- v7.1 → 24/Feb/2015 Pulizia core & ottimizzazioni
- v7.2 → 10/Mar/2015 Frontend
- v7.3 → 21/Apr/2015 Ecosistema Composer
- v7.4 → 09/Jun/2015 Revisione Backend Vol 2
- v7.5 → 28/Jul/2015 (*da determinare...*)
- v7.6 → 13/Oct/2015 pre-LTS inferno
- v7.7 → xx/xxx/2015 **TYPO3 CMS 7 LTS** (Long Term Release)

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

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

Introduzione

Installazione

- Procedura ufficiale di installazione su Linux/Mac OS X (DocumentRoot ad esempio /var/www/site/htdocs):

```
$ cd /var/www/site
$ wget --content-disposition get.typo3.org/7.1
$ tar xzf typo3_src-7.1.0.tar.gz
$ cd htdocs
$ ln -s ../typo3_src-7.1.0 typo3_src
$ ln -s typo3_src/index.php
$ ln -s typo3_src/typo3
$ touch FIRST_INSTALL
```

- Link simbolici in Microsoft Windows:
 - Use junction in Windows XP/2000
 - Use `mlink` in Windows Vista and Windows 7

Introduzione

Aggiornamento a TYPO3 CMS 7.x

- Aggiornamenti possibili solo da TYPO3 CMS 6.2 LTS
- TYPO3 CMS < 6.2 deve essere prima aggiornato a TYPO3 CMS 6.2 LTS
- Istruzioni per l'aggiornamento:
http://wiki.typo3.org/Upgrade#Upgrading_to_7.1
- Guida ufficiale TYPO3 "TYPO3 Installation and Upgrading":
<http://docs.typo3.org/typo3cms/InstallationGuide>
- Approccio generale:
 - Verifica i requisiti minimi di sistema (PHP, MySQL, etc.)
 - Verifica **deprecation_*.log** nella vecchia istanza TYPO3
 - Aggiorna tutte le estensioni all'ultima versione
 - Imposta il nuovo sorgente ed esegui Install Tool → Upgrade Wizard
 - Verifica modulo startup per gli utenti di backend (opzionale)

Capitolo 1: Interfaccia utente Backend

BackendUI

Look & Feel: componente data/orario

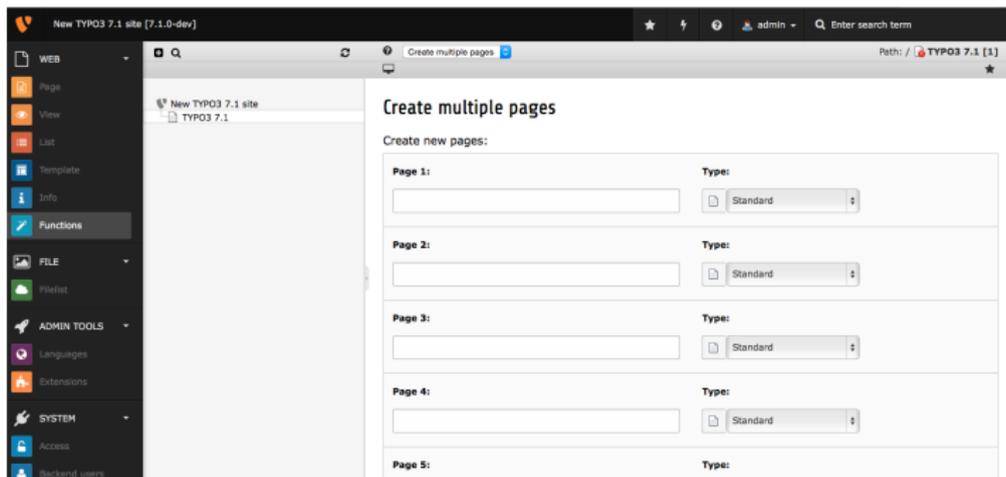
Il componente data/orario è stato sostituito con il widget basato su Bootstrap

The screenshot displays a Bootstrap-based widget for date and time selection. It features a calendar for February 2015 with the 14th highlighted in blue. To the right of the calendar is a time selection interface showing '06' and '57' separated by a colon, with up and down arrows for adjustment. Below the calendar is a text input field containing '6:57 14-2-2015' and a calendar icon. To the right of the input field is an 'Enabled' checkbox. The widget is overlaid on a page with sections titled 'Categories' and 'to Subpages'.

BackendUI

Look & Feel: Modulo Funzioni

"Crea pagine" e "Ordina pagine" sono stati spostati in Web => Functions
(in TYPO3 CMS < 7.1, erano presenti in "Web => Functions => Wizards")



Look & Feel: Modulo Accessi

Il modulo Web => Accessi permette di lasciare invariati i permessi utente/gruppo quando sono sovrascritti

Permissions: EDIT

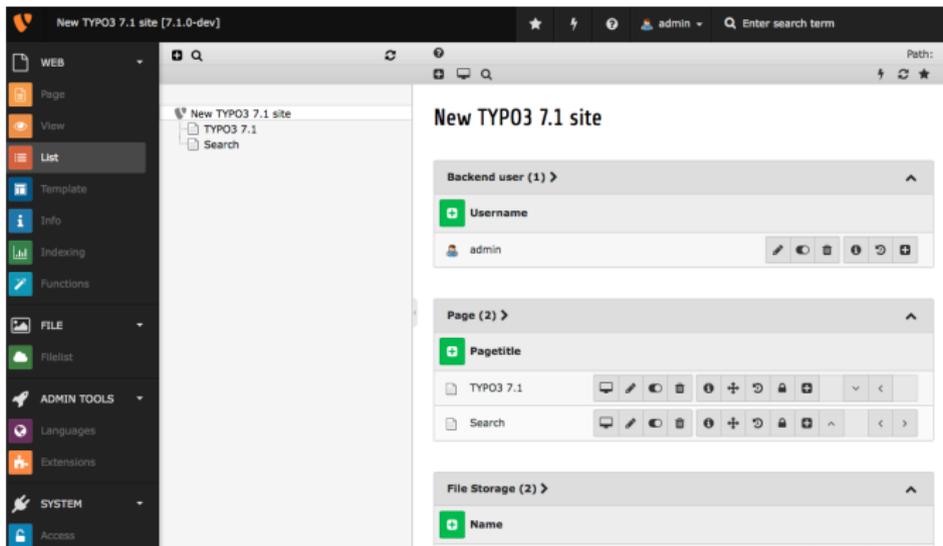
Owner

Group

	Show page	Edit content	Edit page	Delete page	New pages
Owner	<input checked="" type="checkbox"/>				
Group	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Everybody	<input type="checkbox"/>				

Look & Feel: Icone nel modulo Lista

Le icone ("action buttons") nel modulo Lista sono stati divisi in due gruppi (prima le azioni primarie (read, update, delete), seguiti dalle azioni secondarie)



Capitolo 2: TSconfig & TypoScript

Tsconfig & TypeScript

StdWrap per page.headTag

- L'impostazione TypeScript `page.headTag` adesso dispone delle funzionalità `stdWrap`

```
page = PAGE
page.headTag = <head>
page.headTag.override = <head class="special">
page.headTag.override.if {
    isInList.field = uid
    value = 24
}
```

TScnfig & TypoScript

Inclusione asincrona dei file JavaScript

- I file JavaScript possono essere caricati in modo asincrono

```
page {  
  includeJS {  
    jsFile = /path/to/file.js  
    jsFile.async = 1  
  }  
}
```

- Può essere utilizzato in:
 - includeJSlibs / includeJSLibs
 - includeJSFooterlibs
 - includeJS
 - includeJSFooter

TSconfig & TypoScript

Le voci di HMENU possono essere selezionate con
`additionalWhere`

- L'oggetto TypoScript HMENU dispone della nuova funzionalità `additionalWhere`
- Questo significa che possono essere eseguite query più specifiche nel database (es. filtrare)
- Esempio:

```
lib.authormenu = HMENU
lib.authormenu.1 = TMENU
lib.authormenu.1.additionalWhere = AND author!=""
```

Tsconfig & TypoScript

Proprietà aggiuntive per il menu HMENU browse

- Due nuove proprietà per l'oggetto HMENU (option "special=browse") per selezionare le voci di menu in modo più granulare:
 - `excludeNoSearchPages`
 - `includeNotInMenu`
- Esempio:

```
lib.browsemenu = HMENU
lib.browsemenu.special = browse
lib.browsemenu.special.excludeNoSearchPages = 1
lib.browsemenu.includeNotInMenu = 1
```

TSconfig & TypeScript

Intestazioni HTTP multiple

- Header HTTP è configurabile come array (`config.additionalHeaders`)
- Questo permette di avere configurazioni multiple dell'intestazione

```
config.additionalHeaders {
  10 {
    # header string
    header = WWW-Authenticate: Negotiate

    # (optional) replace previous headers with the same name (default: 1)
    replace = 0

    # (optional) force HTTP response code
    httpResponseCode = 401
  }
  # set second additional HTTP header
  20.header = Cache-control: Private
}
```

TSconfig & TypoScript

Aggiunta l'opzione "auto" per config.absRefPrefix

- L'impostazione TypoScript `config.absRefPrefix` può essere utilizzata per l'URL rewriting. Un alternativa a `config.baseURL` (per configurare un dominio specifico), `absRefPrefix` può individuare automaticamente il dominio del sito:

```
config.absRefPrefix = auto
```

```
# ...invece di:
```

```
[ApplicationContext = Production]  
config.absRefPrefix = /
```

```
[ApplicationContext = Testing]  
config.absRefPrefix = /my_site_root/
```

Nota: La nuova opzione è sicura anche per gli ambienti multidominio per evitare meccanismi di caching duplicati.

TSconfig & TypoScript

Codice ISO di due lettere per `sys_language` (1)

- La gestione delle lingue avviene con i record memorizzati nel DB nella tabella `sys_language`, e si ha generalmente un riferimento con `sys_language_uid`
- In TYPO3 CMS 7.1, è stato introdotto il codice ISO 639-1 di due lettere:
 - Nuovo campo nel DB: `sys_language.language_isocode`
 - Nuova opzione TypoScript: `sys_language_isocode`

Nota: **ISO 639** è uno standard della International Organization for Standardization. L'elenco dei codici ISO 639-1 è disponibile a:

http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes

TScnfig & TypoScript

Codice ISO di due lettere per sys_language (2)

■ Esempio:

```
# Danese di default
config.sys_language_uid = 0
config.sys_language_isocode_default = da

[globalVar = GP:L = 1]
  # codice ISO nella tabella sys_language (uid 1)
  config.sys_language_uid = 1
  # codice ISO sovrascritto come necessario
  config.sys_language_isocode = fr
[GLOBAL]

page.10 = TEXT
page.10.data = TSFE:sys_language_isocode
page.10.wrap = <div class="main" data-language="|">
```

TSconfig & TypoScript

Personalizzare le condizioni TypoScript nel Backend

- Le condizioni personalizzate sono state introdotte nel **frontend** nella versione TYPO3 CMS 7.0
- Da TYPO3 CMS 7.1, è possibile utilizzare condizioni personalizzate anche nel **backend**
- Le condizioni devono derivare da `AbstractCondition` ed implementare il metodo `matchCondition()`
- Esempio di uso in TypoScript:

```
[BigCompanyName\TypoScriptLovePackage\MyCustomTypoScriptCondition]
```

```
[BigCompanyName\TypoScriptLovePackage\MyCustomTypoScriptCondition = 7]
```

```
[BigCompanyName\TypoScriptLovePackage\MyCustomTypoScriptCondition = 7, != 6]
```

```
[BigCompanyName\TypoScriptLovePackage\MyCustomTypoScriptCondition = {$mysite.myconstant}]
```

TScnfig & TypoScript

Personalizzare le icone in PageTScnfig

- Le coppie etichetta/valore dei campi di selezioni possono già essere configurate con l'opzione `addItem`s in PageTScnfig
- Ora è possibile personalizzare anche le **icone** di questi campi
 - Opzione 1: utilizzando `addItem`s e la sottoproprietà `.icon`
 - Opzione 2: utilizzando `altIcons` (in generale su tutti gli elementi)
- Esempio:

```
TCEFORM.pages.doktype.addItem {
    10 = My Label
    10.icon = EXT:t3skin/icons/gfx/i/pages.gif
}
TCEFORM.pages.doktype.altIcons {
    10 = EXT:myext/icon.gif
}
```

Tsconfig & TypoScript

Estendere l'elemento browser con nuovi alberi

- La nuova opzione `.append` in `UserTsconfig` permette all'amministratore di **aggiungere** nuovi alberi, invece di sostituire le configurazioni con i punti dell'albero dell'utente
- Esempio:

```
options.pageTree.altElementBrowserMountPoints = 20,31  
options.pageTree.altElementBrowserMountPoints.append = 1
```

Tsconfig & TypoScript

Override delle etichette di checkbox e radiobutton

- Le etichette dei radiobutton e checkbox possono essere sovrascritti
- Esempio:

```
// campo con un singolo checkbox (usa ".default")
TCEFORM.pages.hidden.altLabels.default = nuova etichetta
TCEFORM.pages.hidden.altLabels.default = LLL:path/to/languagefile.xlf:individualLabel

// campo con vari checkbox (0, 1, 2, 3...)
TCEFORM.pages.l18n_cfg.altLabels.0 = nuova etichetta del primo checkbox
TCEFORM.pages.l18n_cfg.altLabels.1 = nuova etichetta del secondo checkbox
TCEFORM.pages.l18n_cfg.altLabels.2 = nuova etichetta del terzo checkbox
...
```

TScnfig & TypoScript

Varie (1)

- Larghezza e altezza del widget Browser possono essere configurati in UserTScnfig:

```
options.popupWindowSize = 400x900  
options.RTE.popupWindowSize = 200x200
```

- PageTScnfig: una nuova configurazione di RTE può essere utilizzata per configurare il target di default per i link di un certo tipo:

```
buttons.link.[type].properties.target.default
```

Dove [type] può essere page, file, url, mail o spec
(le estensioni possono fornire altri tipi)

TSconfig & TypoScript

Varie (2)

- I titoli delle sezioni dei risultati della ricerca sono di default dei link. E' possibile disabilitare questi link e visualizzare delle sezioni di semplice testo.

```
plugin.tx_indexedsearch.linkSectionTitles = 0
```

- Ora getData può accedere ai dati field (non solo agli array come GPVar e TSFE):

```
10 = TEXT  
10.data = field:fieldname|level1|level2
```

- Impostazione TypoScript config.pageTitle dispone delle funzionalità stdWrap

```
# trasforma in maiuscolo il valore di <title>  
page = PAGE  
page.config.pageTitle.case = upper
```

Capitolo 3: Modifiche rilevanti

Modifiche rilevanti

TCA: Numero massimo di caratteri nell'elemento testo

- Il tipo `text` in TCA supporta ora l'attributo HTML5 `maxlength` per limitare la lunghezza di un testo (nota: il ritorno a capo è solitamente contato come due caratteri)

```
'teaser' => array(  
  'label' => 'Teaser',  
  'config' => array(  
    'type' => 'text',  
    'cols' => 60,  
    'rows' => 2,  
    'max' => '30' // <-- maxlength  
  )  
)
```

Da notare, che non tutti i browser supportano questo attributo. Vedi [Browser Support List](#) per dettagli.

Modifiche rilevanti

Nuova implementazione SplFileInfo

- Nuova classe: TYPO3\CMS\Core\Type\File\FileInfo
- Questa classe estende la classe SplFileInfo, che consente il recupero delle informazioni meta dai file

```
$fileIdentifier = '/tmp/foo.html';  
$fileInfo = GeneralUtility::makeInstance(  
    \TYPO3\CMS\Core\Type\File\FileInfo::class,  
    $fileIdentifier  
);  
echo $fileInfo->getMimeType(); // output: text/html
```

- Implementazioni personalizzate possono utilizzare il seguente hook:

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
    [\TYPO3\CMS\Core\Type\File\FileInfo::class]['mimeTypeGuessers']
```

Modifiche rilevanti

UserFunc nelle condizioni di visualizzazione di TCA

- `userFunc displayCondition` permette di controllare in qualsiasi condizione o stato immaginabile
- Se una situazione non può essere controllata con nessuno dei controlli disponibili, gli sviluppatori possono creare le proprie user function (ritorna TRUE/FALSE per visualizzare/nascondere appropriati campi TCA)

```
$GLOBALS['TCA']['tt_content']['columns']['bodytext']['displayCond'] =  
    'USER:Vendor\\Example\\User\\ElementConditionMatcher->  
        checkHeaderGiven: any: more: information';
```

Modifiche rilevanti

API per il modulo Twitter Bootstrap (1)

- Due nuove API per creare/rimuovere popup modali:
 - `TYPO3.Modal.confirm(title, content, severity, buttons)`
 - `TYPO3.Modal.dismiss()`
- Le opzioni `title` e `content` sono richieste
- Le opzioni `buttons.text` e `buttons.trigger` sono anche richieste, se è usato `buttons`
- Esempio 1:

```
TYPO3.Modal.confirm(  
    'The title of the modal',           // title  
    'This the body of the modal', // content  
    TYPO3.Severity.warning             // severity  
);
```

Modifiche rilevanti

API per il modulo Twitter Bootstrap (2)

■ Esempio 2:

```
TYPO3.Modal.confirm('Warning', 'You may break the internet!',
    TYPO3.Severity.warning,
    [
        {
            text: 'Break it',
            active: true,
            trigger: function() { ... }
        },
        {
            text: 'Abort!',
            trigger: function() {
                TYPO3.Modal.dismiss();
            }
        }
    ]
);
```

Modifiche rilevanti

Archivio API Javascript (1)

- L'accesso alla configurazione utente del BE (`$BE_USER->uc`) può essere gestito in JavaScript usando semplici coppie chiave-valore
- Inoltre, il [localStorage](#) di HTML5 può essere utilizzato per memorizzare i dati nel browser dell'utente (client-side)
- Due nuovi oggetti globali TYPO3:
 - `top.TYPO3.Storage.Client`
 - `top.TYPO3.Storage.Persistent`
- Ogni oggetto ha le API seguenti:
 - `get(key)`: recupera il dato
 - `set(key, value)`: scrive il dato
 - `isset(key)`: verifica se la chiave viene usata
 - `clear()`: svuota tutto l'archivio dei dati

Modifiche rilevanti

Archivio API Javascript (2)

■ Esempio:

```
// leggi il valore della chiave 'startModule'  
var value = top.TYPO3.Storage.Persistent.get('startModule');  
  
// scrivi il valore 'web_info' come chiave di 'start_module'  
top.TYPO3.Storage.Persistent.set('startModule', 'web_info');
```

Modifiche rilevanti

Inline Rendering dei Checkbox

- L'impostazione `inline` dei checkbox per "cols" può essere usata per visualizzare i checkbox direttamente uno vicino all'altro per ridurre lo spazio utilizzato

```
'weekdays' => array(  
  'label' => 'Weekdays',  
  'config' => array(  
    'type' => 'check',  
    'items' => array(  
      array('Mo', ''),  
      array('Tu', ''),  
      array('We', ''),  
      array('Th', ''),  
      array('Fr', ''),  
      array('Sa', ''),  
      array('Su', '')  
    ),  
    'cols' => 'inline'  
  )  
)  
,  
...  
...
```

Modifiche rilevanti

Registrazione dei Content Object

- Una nuova opzione globale per registrare e/o estendere/sovrascrivere cObjects come ad esempio TEXT è stata introdotta
- Una lista di tutti i cObjects presenti è disponibile come:

```
$GLOBALS['TYPO3_CONF_VARS']['FE']['ContentObjects']
```

- Esempio: registra un nuovo cObject EXAMPLE

```
$GLOBALS['TYPO3_CONF_VARS']['FE']['ContentObjects']['EXAMPLE'] =  
    Vendor\MyExtension\ContentObject\ExampleContentObject::class;
```

- La classe registrata deve essere una sottoclasse di
TYPO3\CMS\Frontend\ContentObject\AbstractContentObject
- Registra la tua classe nella directory
typo3conf/myextension/Classes/ContentObject/
per essere disponibile per un futuro meccanismo di autocaricamento

Modifiche rilevanti

Hooks e Signals (1)

- Un nuovo hook è stato aggiunto alla fine di `PageRepository->init()`, per permettere un intervento sulla visibilità delle pagine

- Registra l'hook come di seguito:

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
    [\TYPO3\CMS\Frontend\Page\PageRepository::class]['init']
```

- La classe dell'hook deve implementare la seguente interfaccia:

```
\TYPO3\CMS\Frontend\Page\PageRepositoryInitHookInterface
```

Modifiche rilevanti

Hooks e Signals (2)

- Un nuovo hook è stato aggiunto a `PageLayoutView` per modificare il rendering del footer di un elemento di contenuto.

- Esempio:

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
    ['cms/layout/class.tx_cms_layout.php']['tt_content_drawFooter'];
```

- La classe dell'hook deve implementare la seguente interfaccia:

```
\TYPO3\CMS\Backend\View\PageLayoutViewDrawFooterHookInterface
```

Modifiche rilevanti

Hooks e Signals (3)

- Un nuovo hook è stato aggiunto come post processor di `BackendUtility::countVersionsOfRecordsOnPage`
- Questo può essere utilizzato per visualizzare lo stato dei workspace in un albero di pagine per esempio
- Registra l'hook come di seguito:

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']  
  ['t3lib/class.t3lib_befunc.php']['countVersionsOfRecordsOnPage'][] =  
  'My\Package\HookClass->hookMethod';
```

Modifiche rilevanti

Hooks e Signals (4)

- Un nuovo signal è stato aggiunto alla fine del metodo `DataPreprocessor::fetchRecord()`
- Questo può essere utilizzato per modificare l'array `regTableItems_data` per esempio, al fine di visualizzare i dati modificati in `TCEForms`

```
$this->getSignalSlotDispatcher()->dispatch(  
    \TYPO3\CMS\Backend\Form\DataPreprocessor::class,  
    'fetchRecordPostProcessing',  
    array($this)  
);
```

Modifiche rilevanti

Hooks e Signals (5)

- Un nuovo signal è stato aggiunto, che permette ulteriori elaborazioni sull'inizializzazione di un oggetto mailer, es. registrare un Swift Mailer plugin

```
$signalSlotDispatcher = \TYPO3\CMS\Core\Utility\GeneralUtility::makeInstance(  
    \TYPO3\CMS\Extbase\SignalSlot\Dispatcher::class  
);  
  
$signalSlotDispatcher->connect(  
    \TYPO3\CMS\Core\Mail\Mailer::class,  
    'postInitializeMailer',  
    \Vendor\Package\Slots\MailerSlot::class,  
    'registerPlugin'  
);
```

Modifiche rilevanti

Multipli UID in PageRepository::getMenu()

- Il metodo PageRepository::getMenu() ora accetta un array, per poter definire più pagine root.

```
$pageRepository = new \TYPO3\CMS\Frontend\Page\PageRepository();  
$pageRepository->init(FALSE);  
$rows = $pageRepository->getMenu(array(2, 3));
```

Capitolo 4: Extbase & Fluid

Extbase & Fluid

PaginateViewHelper

- Da TYPO3 CMS 7.1, PaginateViewHelper accetta in input collections dei seguenti tipi:
 - QueryResultInterface
 - ObjectStorage
 - ArrayAccess
 - array

- Esempio:

```
<f:widget.paginate objects="{blogs}" as="paginatedBlogs">
  <f:for each="{paginatedBlogs}" as="blog">
    <h4>{blog.title}</h4>
  </f:for>
</f:widget.paginate>
```

ContainerViewHelper carica i moduli RequireJS

- ContainerViewHelper può caricare i moduli RequireJS attraverso l'attributo `includeRequireJsModules`
- Esempio:

```
<f:be.container pageTitle="Extension Module" loadJQuery="true"
  includeRequireJsModules="{
    0: 'TYPO3\CMS\Extension\Module1',
    1: 'TYPO3\CMS\Extension\Module2',
    2: 'TYPO3\CMS\Extension\Module3',
    3: 'TYPO3\CMS\Extension\Module4'
  }" >
```

Metodo `has()` in `ObjectAccess`

- Nel loro utilizzo in Fluid, `object.property` e `object.isProperty` supportano l'uso dei seguenti metodi:
 - `isProperty()`
 - `getProperty()`
- Novità da TYPO3 CMS 7.1: `hasProperty()`
- la chiamata al metodo `$object->hasProperty()` se `object.hasProperty` è utilizzata in Fluid

Caricamento multiplo di file con FormUpload-ViewHelper

- FormUpload-Viewhelper supporta il nuovo attributo `multiple`, che permette di caricare più file nello stesso momento.

```
<f:form.upload property="files" multiple="multiple" />
```

Nota: gli sviluppatori hanno bisogno di preparare il valore in ingresso per la mappatura della proprietà da parte del proprio `TypeConverter`!

Capitolo 5: Funzionalità deprecate/rimosse

Funzionalità deprecate/rimosse

`$TYPO3_CONF_VARS[SYS][compat_version]`

- L'opzione `$TYPO3_CONF_VARS[SYS][compat_version]` (gestita negli aggiornamenti nel wizard dell'Install Tool) è stato rimosso
- Tutti i controlli su `GeneralUtility::compat_version` sono ora fatti dalla costante `TYP03_branch`

Nota: Adesso le condizioni TypoScript, che verificavano il vecchio `compat_version` hanno un comportamento differente!

Funzionalità deprecate/rimosse

Stile inline del tag <blockquote>

- CSS Styled Content visualizza il tag <blockquote> usando le indicazioni TypoScript in `lib.parseFunc_RTE`

- Le righe seguenti sono state rimosse senza sostituzioni:

```
lib.parseFunc_RTE.externalBlocks.blockquote.callRecursive.tagStdWrap.HTMLparser = 1
lib.parseFunc_RTE.externalBlocks.blockquote.callRecursive.tagStdWrap.HTMLparser.tags.blockquote.
  overrideAttribs = style="margin-bottom:0;margin-top:0;"
```

- Con il risultato che lo stile inline
"margin-bottom:0;margin-top:0;"
è stato rimosso

Nota: lo stile del tag <blockquote> potrebbe essere differente dopo l'aggiornamento a TYPO3 CMS 7.1

Funzionalità deprecate/rimosse

Workspaces: campo `disable_autocreate`

- Il campo deprecato `disable_autocreate` è stato rimosso da `EXT:workspaces`
- Se un estensione TYPO3 utilizza questo campo, sarà restituito un errore SQL

Funzionalità deprecate/rimosse

Funzionalità `include_once`

- La funzionalità di inclusione file PHP con il modulo funzioni (es. modulo Info) con un array `include_once` è stato rimosso
- Questo vale per i seguenti moduli:
 - Web => Page
 - Web => Page - New Content Element Wizard
 - Web => Functions
 - Web => Info
 - Web => Template
 - Web => Recycler
 - User => Task Center
 - System => Scheduler

Funzionalità deprecate/rimosse

Impostazioni `config.meaningfulTempFilePrefix`

- In TYPO3 CMS < 7.1, il nome dei file delle immagini generato con GIFBUILDER poteva essere personalizzato con opzioni TypoScript: `config.meaningfulTempFilePrefix` (GIFBUILDER usava un codice hash come nome del file)
- Questa opzione è stata rimossa (nomi dei file nella directory `typo3temp/GB/` mostrano il nome originale del file come primo elemento in modo automatico)

Funzionalità deprecate/rimosse

Rimossi file

- I seguenti **file** sono stati rimossi:
 - typo3/file_edit.php
 - typo3/file_newfolder.php
 - typo3/file_rename.php
 - typo3/file_upload.php
 - typo3/show_rechis.php
 - typo3/listframe_loader.php
- Le loro funzionalità sono state spostate nei moduli di backend, es. `typo3/file_edit.php` in `BackendUtility::getModuleUrl('file_edit');`

Funzionalità deprecate/rimosse

ExtJS DateTimePicker

- Il componente ExtJS `Ext.ux.DateTimePicker` è stato rimosso e sostituito con un componente alternativo di Twitter Bootstrap (vedi capitolo "BackendUI")
- Le estensioni di sistema di TYPO3 CMS interessate sono `EXT:belog` o `EXT:scheduler` per esempio

Nota: le estensioni che si basano sul componente vecchio `Ext.ux.DateTimePicker` molto probabilmente non funzioneranno!

Funzionalità deprecate/rimosse

Access List Render Mode

- Le seguenti **variabili** sono state rimosse:
`$GLOBALS[TYPO3_CONF_VARS][BE][accessListRenderMode]`
- I corrispondenti campi nelle tabelle TCA `be_users` e `be_groups` sono impostati al valore di default "checkbox"
- Questi possono essere cambiati nel file `typo3conf/extTables.php`:

```
$GLOBALS['TCA']['be_users']['columns']['file_permissions']['config']['renderMode'] = 'singlebox';  
$GLOBALS['TCA']['be_users']['columns']['userMods']['config']['renderMode'] = 'singlebox';
```

```
$GLOBALS['TCA']['be_groups']['columns']['file_permissions']['config']['renderMode'] = 'singlebox';  
$GLOBALS['TCA']['be_groups']['columns']['pagetypes_select']['config']['renderMode'] = 'singlebox';  
$GLOBALS['TCA']['be_groups']['columns']['tables_select']['config']['renderMode'] = 'singlebox';  
$GLOBALS['TCA']['be_groups']['columns']['tables_modify']['config']['renderMode'] = 'singlebox';  
$GLOBALS['TCA']['be_groups']['columns']['non_exclude_fields']['config']['renderMode'] = 'singlebox';  
$GLOBALS['TCA']['be_groups']['columns']['userMods']['config']['renderMode'] = 'singlebox';
```

Funzionalità deprecate/rimosse

Elemento di contenuto "Mailform"

- Le funzionalità Mailform, fornite all'oggetto FORM, sono state rimosse dal core
- Sono disponibili nell'estensione di compatibilità EXT:compatibility6 se richieste
- Le seguenti opzioni sono state marcate come deprecate:

```
$TYPO3_CONF_VARS] [FE] [secureFormmail]  
$TYPO3_CONF_VARS] [FE] [strictFormmail]  
$TYPO3_CONF_VARS] [FE] [formmailMaxAttachmentSize]
```

- I seguenti metodi con TypoScriptFrontendController sono stati rimossi:

```
protected checkDataSubmission()  
protected sendFormmail()  
public extractRecipientCopy()  
public codeString()  
protected roundTripCryptString()
```

Funzionalità deprecate/rimosse

Funzionalità modificate (1)

- EXT: `indexed_search` è attivo appena l'estensione è installata. Il risultato è che le opzioni TypoScript `config.index_enable = 1` e `config.index_externals = 1` diventano attive automaticamente
- L'opzione TSconfig `web_func.menu.wiz` cambia in `web_func.menu.functions`
- Le estensioni, che si integrano nella toolbar in alto a destra, devono implementare la nuova interfaccia:
`TYPO3\CMS\Backend\Toolbar\ToolbarItemInterface`
e devono essere registrate in:
`$GLOBALS['TYPO3_CONF_VARS']['BE']['toolbarItems']`

Funzionalità deprecate/rimosse

Funzionalità modificate (2)

- I file `typo3/js/tree.js` sono stati sostituiti da `EXT:backend/Resources/Public/JavaScript/LegacyTree.js` (basato su jQuery)
- La variabile `$GLOBALS['WEBMOUNTS']` è stata sostituita da `$GLOBALS['BE_USER']->returnWebmounts()`
- Il supporto a `.t3-table` e `.t3-button` sono stati rimossi (Le classi Twitter Bootstrap gestiscono ora l'aspetto grafico)
- Le bandiere delle nazioni (immagini PNG) sono state spostate da `typo3/gfx/flags/` e `typo3/sysex/t3skin/images/flags/` in: `typo3/sysex/core/Resources/Public/Icons/flags/`

Funzionalità deprecate/rimosse

Funzionalità modificate (3)

- I template CSS Styled Content TypoScript da TYPO3 CMS 4.5 a 6.1 sono stati rimossi
- I seguenti cObjects TypoScript sono stati spostati nell'estensione di compatibilità EXT:compatibility6:

SEARCHRESULTS	COLUMNS	OTABLE	CLEARGIF
IMGTEXT	CTABLE	HRULER	

- L'elemento di contenuto search è stato spostato nell'estensione di compatibilità EXT:compatibility6
- Le seguenti opzioni TCA sono state rimosse:

_PADDING	_VALIGN	_DISTANCE
----------	---------	-----------

Funzionalità deprecate/rimosse

Opzione TypoScript andWhere

- L'opzione TypoScript andWhere è stata marcata come deprecata
- Gli Integrators devono usare la proprietà where e markers al suo posto:

```
page.30 = CONTENT
page.30 {
    table = tt_content
    select {
        pidInList = this
        orderBy = sorting
        where {
            dataWrap = sorting>{field:sorting}
        }
    }
}
```

```
page.60 = CONTENT
page.60 {
    table = tt_content
    select {
        pidInList = 73
        where = header != ###whatever###
        orderBy = ###sortfield###
        markers {
            whatever.data = GP:first
            sortfield.value = sor
            sortfield.wrap = |ting
        }
    }
}
```

Funzionalità deprecate/rimosse

Entry point deprecati

- I seguenti entry points sono stati classificati come deprecati:

- typo3/tce_file.php
- typo3/move_el.php
- typo3/tce_db.php
- typo3/login_frameset.php
- typo3/sysextd/cms/layout/db_new_content_el.php
- typo3/sysextd/cms/layout/db_layout.php

- Al loro posto, vanno utilizzati i seguenti:

```
\TYPO3\CMS\Backend\Utility\BackendUtility::getModuleUrl('<parameter>')
```

Where *<parameter>* could be:

```
tce_file, move_element, tce_db, login_frameset,  
new_content_element, web_layout
```

Funzionalità deprecate/rimosse

Varie (1)

- L'opzione TypoScript `config.xhtmlDoctype = xhtml_2` è stata marcata da rimuovere in TYPO3 CMS 8

- I seguenti metodi sono stati marcati come deprecati:

```
TYPO3\CMS\Backend\Utility\BackendUtility::implodeTSParams()  
TYPO3\CMS\Backend\Controller::makeLoginBoxImage()
```

- Il seguente metodo è stato marcato come deprecato:

```
LocalImageProcessor::getTemporaryImageWithText()
```

...e sostituito da:

```
TYPO3\CMS\Core\Imaging\GraphicalFunctions::getTemporaryImageWithText()
```

- Le proprietà StdWrap `textStyle` e `tableStyle` sono state marcate come deprecate

Funzionalità deprecate/rimosse

Varie (2)

- L'opzione TypoScript `page.includeJSlibs` è stata rinominata in `page.includeJSLibs` ("L" maiuscola) e la vecchia opzione marcata come deprecata
- La condizione TypoScript `device` è stata marcata come deprecata
- Il metodo `DocumentTable::table()` è stato marcato come deprecato (gli sviluppatori devono usare Fluid per questo)
- I seguenti metodi sono stati marcati come deprecati:

```
TYPO3\CMS\Frontend\Controller\
```

```
    TypoScriptFrontendController::doXHTML_cleaning()
```

...ed di conseguenza l'opzione TypoScript `config.xhtml_cleaning`

- Il seguente hook è stato marcato come deprecato:

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

Funzionalità deprecate/rimosse

Varie (3)

- I seguenti metodi sono stati marcati come deprecati:

```
TypoScriptTemplateObjectBrowserModuleFunctionController::  
    verify_TSubjects()  
ExtendedTemplateService::ext_getKeyImage()  
ConfigurationForm::ext_getKeyImage()
```

- L'esecuzione di `contentObject->COBJECT()` è stata marcata come deprecata
(usare `$cObj->cObjGetSingle('...', $conf)`; al suo posto)
- L'accesso diretto a `FormEngine::$renderReadOnly` è stato marcato come deprecato
(usare `AbstractFormElement::setRenderReadOnly(TRUE)`; al suo posto)

Funzionalità deprecate/rimosse

Varie (4)

- I seguenti metodi di FormEngine sono stati marcati come deprecati:
 - FormEngine::insertDefStyle
 - FormEngine::getAvailableLanguages()
 - FormEngine::sL()
 - FormEngine::renderVDEFDiff()
 - FormEngine::getLL()
 - FormEngine::getTSCpid()
 - FormEngine::getSingleField_typeFlex_langMenu()
 - FormEngine::getSingleField_typeFlex_sheetMenu()
 - FormEngine::getSpecConfFromString()

Capitolo 6: Fonti e autori

Fonti e autori

Fonti (1)

TYPO3 News:

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

Note sulla release:

- http://wiki.typo3.org/TYPO3\CMS_7.1.0
- [INSTALL.md](#) e [Changelog](#)
- [typo3/sysext/core/Documentation/Changelog/7.1/*](http://typo3.org/typo3/sysext/core/Documentation/Changelog/7.1/*)

TYPO3 Bug-/Segnalazioni:

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

Archivio TYPO3 Git:

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

TYPO3 CMS What's New Slides:

Patrick Lobacher

(Ricerca, raccolta informazioni e versione tedesca)

Michael Schams

(Leader del progetto e versione inglese)

Tradotto da:

Andrey Aksenov, Paul Blondiaux, Sergio Catala, Ben van't Ende,
Michel Mix, Sinisa Mitrovic, Nena Jelena Radovic e Roberto Torresani

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

Licensed under Creative Commons BY-NC-SA 3.0

