

TYP03 CMS 7.1 – Qué hay Nuevo

Resumen de las nuevas características, cambios y mejoras

Creado por:

Patrick Lobacher y Michael Schams

Traducción en Español por:

Sergio Catala

TYPO3 CMS 7.1 - Qué hay Nuevo

Resumen de Capítulos

Introducción

Interfaz del Usuario del Backend

TSconfig & TypoScript

Cambios En Profundidad

Extbase & Fluid

Funciones obsoletas/eliminadas

Fuentes y Autores

Introducción

Los Hechos

Introducción

TYPO3 CMS 7.1 - Los Hechos

- Fecha de lanzamiento: 24 Febrero 2015
- Tipo de lanzamiento: "Lanzamiento Sprint"
- Visión: Adoptar, Innovar, Lanzar
- Foco principal: Limpieza del Núcleo y Coordinación



Introducción

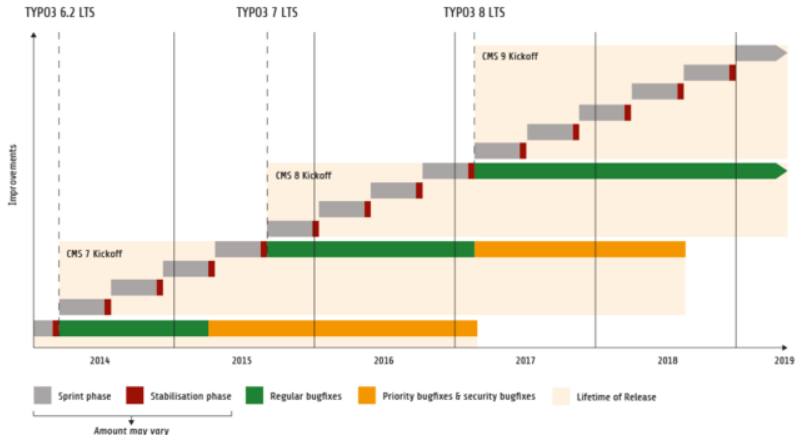
Requerimientos del Sistema

- PHP*: v5.5.0 - v5.6.x
- MySQL: v5.5.x - v5.6.x (modo no strict)
- Espacio en disco: mín 200 MB
- Ajustes PHP:
 - memory_limit >= 128M
 - max_execution_time >= 240s
 - opción de compilación `-disable-ipv6` no debe usarse
- Backend requiere IE >= 9 o cualquier otro navegador moderno

*) Más detalles: [Requisitos Mínimos de PHP para TYPO3 CMS 7](#)

Introducción

Desarrollo y Línea de tiempo de Lanzamiento



Introducción

Línea de lanzamiento de TYPO3 CMS

Fechas de lanzamiento estimadas y su foco principal:

- v7.0 → 02/Dic/2014 Revisión Backend Vol 1
- v7.1 → 24/Feb/2015 Limpieza de Núcleo & Coordinación
- v7.2 → 10/Mar/2015 Frontend
- v7.3 → 21/Abr/2015 Ecosistema Composer
- v7.4 → 09/Jun/2015 Revisión Backend Vol 2
- v7.5 → 28/Jul/2015 (*por determinar...*)
- v7.6 → 13/Oct/2015 pre-LTS inferno
- v7.7 → xx/xxx/2015 **TYPO3 CMS 7 LTS** (Soporte a Largo Plazo)

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

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

Introducción

Instalación

- Procedimiento de instalación oficial bajo Linux/Mac OS X (DocumentRoot por ejemplo /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
```

- Enlaces simbólicos bajo Microsoft Windows:
 - Use junction bajo Windows XP/2000
 - Use mlink bajo Windows Vista and Windows 7

Introducción

Actualización a TYPO3 CMS 7.x

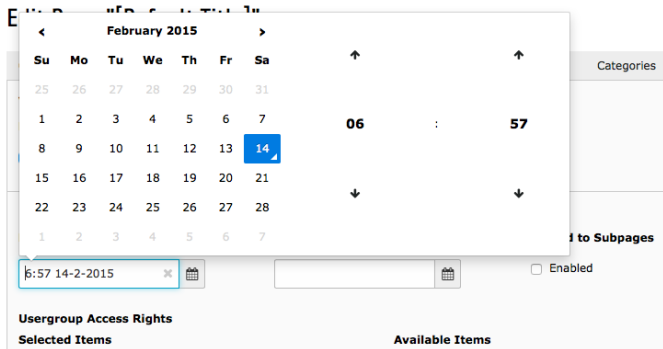
- Actualizaciones sólo posibles desde TYPO3 CMS 6.2 LTS
- TYPO3 CMS < 6.2 debe ser actualizado a TYPO3 CMS 6.2 LTS primero
- Instrucciones de Actualización:
http://wiki.typo3.org/Upgrade#Upgrading_to_7.1
- Guía oficial de TYPO3 "Instalación y Actualización de TYPO3":
<http://docs.typo3.org/typo3cms/InstallationGuide>
- Enfoque general:
 - Comprobar requisitos mínimos del sistema (PHP, MySQL, etc.)
 - Revisar **deprecation_*.log** en vieja instancia de TYPO3
 - Actualizar todas las extensiones a la última versión
 - Desplegar nuevas fuentes y correr la Herramienta de Instalación
→Asistente de Actualización
 - Revisar módulo de inicio para usuarios del backend (opcionalmente)

Capítulo 1: Interfaz del Usuario del Backend

Interfaz del Usuario del Backend

Apariencia: Selector de Fecha/Hora

Se ha reemplazado el selector de Fecha/Hora con una alternativa Bootstrap



The screenshot displays a Bootstrap-style date and time picker. The date picker shows a calendar for February 2015, with the 14th selected. The time picker shows 06:57. Below the picker, there is a text input field containing "6:57 14-2-2015" and a calendar icon. To the right, there is a checkbox labeled "Enabled".

Categories

to Subpages

Enabled

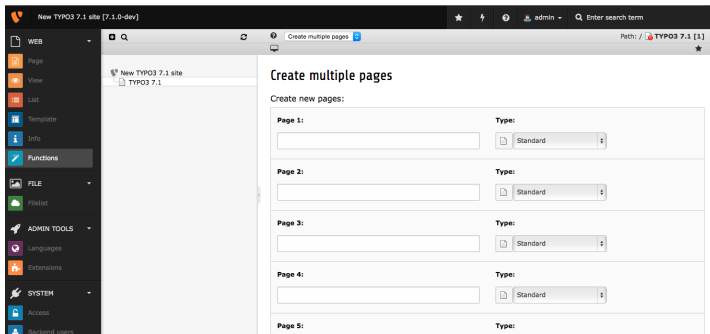
Usergroup Access Rights

Selected Items Available Items

Interfaz del Usuario del Backend

Apariencia: Módulo Funciones

"Crear Páginas" y "Ordenar Páginas" movido a: Web => Funciones
(en TYPO3 CMS < 7.1, estaban ubicadas bajo "Web => Funciones => Asistentes")



Interfaz del Usuario del Backend

Apariencia: Módulo Acceso

Module Web => Acceso permite dejar sin cambios el propietario/grupo al sobrescribir permisos

Permissions: EDIT

Owner

- leave unchanged -

Group

- leave unchanged -

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

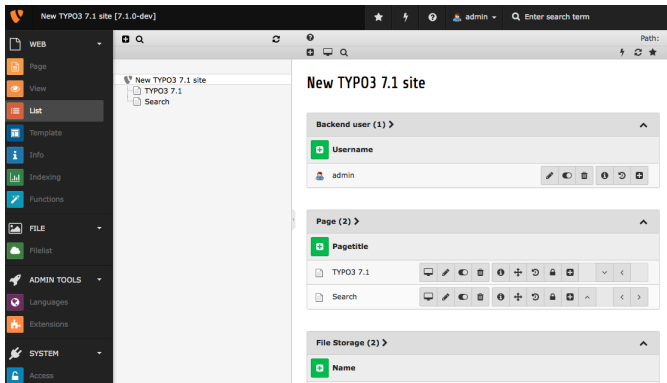
-

Save Abort

Interfaz del Usuario del Backend

Apariencia: Iconos en Módulo Lista

Iconos ("botones de acción") en módulo Lista divididos en dos grupos (acciones primarias primero (leer, actualizar, borrar), seguidas de acciones secundarias)



Capítulo 2: TSconfig & TypoScript

Tsconfig & TypeScript

StdWrap para page.headTag

- Ajuste TypeScript `page.headTag` tiene funcionalidad `stdWrap` ahora

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


TScnfig & TypoScript

Incluir ficheros JavaScript asíncronamente

- Se pueden cargar ficheros JavaScript asíncronamente

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

- Esto afecta a:

- includeJSlibs / includeJSLibs
- includeJSFooterlibs
- includeJS
- includeJSFooter

Tsconfig & TypeScript

Selección de item HMENU vía `additionalWhere`

- TypeScript `cObject` HMENU cuenta con una nueva propiedad `additionalWhere`
- Esto permite una consulta a la base de datos más específica (p.e. filtrado)
- Ejemplo:

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

TScnfig & TypoScript

Propiedades adicionales para menús HMENU browse

- Dos nuevas propiedades para el cObject HMENU (opción "special=browse") para seleccionar ítems de menú más de grano fino:
 - excludeNoSearchPages
 - includeNotInMenu

- Ejemplo:

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

Tsconfig & TypeScript

Múltiples cabeceras HTTP

- Se pueden establecer cabeceras HTTP como un array (`config.additionalHeaders`)
- Esto permite la configuración de múltiples cabeceras a la vez

```
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 & TypeScript

Opción "auto" añadida para config.absRefPrefix

- Se puede usar el ajuste TypeScript `config.absRefPrefix` para permitir la reescritura URL. Como una alternativa a `config.baseURL` (para configurar un dominio específico), `absRefPrefix` puede detectar la raíz del sitio automáticamente:

```
config.absRefPrefix = auto
```

```
# ...instead of:
```

```
[ApplicationContext = Production]
```

```
config.absRefPrefix = /
```

```
[ApplicationContext = Testing]
```

```
config.absRefPrefix = /my_site_root/
```

Nota: La nueva opción es también segura para entornos multi-dominios para evitar mecanismo de cacheo duplicado.

Tsconfig & TypoScript

Código de dos letras ISO para sys_language (1)

- El manejo de idiomas se hace mediante registros almacenados en la tabla de BD `sys_language`, que se referencian normalmente vía `sys_language_uid`
- En TYPO3 CMS 7.1, se ha introducido las dos letras de la ISO 639-1:
 - Nuevo campo DB: `sys_language.language_isocode`
 - Nueva opción TypoScript: `sys_language_isocode`

Nota: **ISO 639** es un conjunto de estándares de la Organización Internacional de Estandarización. La lista de códigos ISO 639-1 está disponible en :

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

Tsconfig & TypoScript

Código de dos letras ISO para sys_language (2)

■ Ejemplo:

```
# Danish by default
config.sys_language_uid = 0
config.sys_language_isocode_default = da

[globalVar = GP:L = 1]
  # ISO code stored in table sys_language (uid 1)
  config.sys_language_uid = 1
  # overwrite ISO code as required
  config.sys_language_isocode = fr
[GLOBAL]

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

TScnfig & TypoScript

Condiciones TypoScript personalizadas en Backend

- Se han introducido ya en TYPO3 CMS 7.0 soporte de condiciones personalizadas para el **frontend**
- Desde TYPO3 CMS 7.1, es también posible usar condiciones personalizadas en el **backend**
- La condición debe derivarse de `AbstractCondition` e implementar el método `matchCondition()`
- Uso de ejemplo en TypoScript:

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

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

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

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


Tsconfig & TypoScript

Iconos personalizados vía PageTsconfig

- Pueden configurarse ya pares valor/etiqueta de campos select a través de la opción PageTsconfig `addItems`
- Es también posible influenciar el **icono** de estos campos ahora
 - Opción 1: usando `addItems` y la subpropiedad `.icon`
 - Opción 2: usando `altIcons` (todos los items en general)
- Ejemplo:

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

TSconfig & TypoScript

Extender elemento browser con puntos de montaje

- Nueva opción UserTSconfig .append permite a los administradores **añadir** puntos de montaje, en lugar de reemplazar los puntos de montaje de base de datos del usuario
- Ejemplo:

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

TConfig & TypoScript

Sobrescritura de etiqueta de checkboxes y radio buttons

- Ahora pueden sobrescribirse etiquetas de radio buttons y checkboxes
- Ejemplo:

```
// field with a single checkbox (use ".default")
TCEFORM.pages.hidden.altLabels.default = new label
TCEFORM.pages.hidden.altLabels.default = LLL:path/to/languagefile.xlf:individualLabel

// field with multiple checkboxes (0, 1, 2, 3...)
TCEFORM.pages.l18n_cfg.altLabels.0 = new label of first checkbox
TCEFORM.pages.l18n_cfg.altLabels.1 = new label of second checkbox
TCEFORM.pages.l18n_cfg.altLabels.2 = new label of third checkbox
...
```

TScnfig & TypoScript

Miscelánea (1)

- Puede configurarse el ancho y alto del Elemento Browser usando UserTScnfig:

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

- PageTScnfig: puede usarse la nueva propiedad de configuración RTE para configurar un destino por defecto para enlaces de un determinado tipo:

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

Donde [type] puede ser page, file, url, mail o spec
(extensiones pueden proporcionar más tipos)

TSconfig & TypoScript

Miscelánea (2)

- Cabeceras de sección de resultados de búsqueda indexada son enlaces por defecto. Ahora es posible deshabilitar estos enlaces y desplegar secciones como textos simples

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

- `getData` puede acceder a datos de un campo ahora (no sólo arrays como `GPVar` y `TSFE`):

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

- Ajuste TypoScript `config.pageTitle` tiene funcionalidad `stdWrap` ahora

```
# make value of <title> upper case  
page = PAGE  
page.config.pageTitle.case = upper
```

Capítulo 3: Cambios En Profundidad

Cambios En Profundidad

TCA: Máximo de caracteres en el elemento de texto

- TCA tipo text ahora soporta el atributo HTML5 `maxlength` para restringir la longitud de un texto (nota: usualmente se cuentan saltos de línea como dos caracteres)

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

Por favor nota que no cada navegador soporta este atributo. Ver [Lista de Soporte de Navegadores](#) para detalles.

Cambios En Profundidad

Nueva implementación SplFileInfo

- Nueva clase: TYPO3\CMS\Core\Type\File\FileInfo
- Esta clase extiende SplFileInfo, que permite recoger meta información de ficheros

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

- Implementaciones personalizadas pueden usar el siguiente hook:

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


Cambios En Profundidad

UserFunc en TCA Display Condition

- `userFunc displayCondition` hace posible chequear cada condición imaginable o estado
- Si una situación no puede ser evaluada con alguno de los chequeos existentes, los desarrolladores pueden desarrollar sus propias funciones de usuario
(devuelve TRUE/FALSE para mostrar/esconder campo TCA apropiado)

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

Cambios En Profundidad

API para modales Twitter Bootstrap (1)

- Dos nuevos métodos API para crear/borrar popups modales:
 - `TYPO3.Modal.confirm(title, content, severity, buttons)`
 - `TYPO3.Modal.dismiss()`
- Se requieren opciones `title` y `content`
- Se requieren también opciones `buttons.text` y `buttons.trigger`, si se usa `buttons`
- Ejemplo 1:

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

Cambios En Profundidad

API para modales Twitter Bootstrap (2)

■ Ejemplo 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();
            }
        }
    ]
);
```

Cambios En Profundidad

API de Almacenamiento JavaScript (1)

- Acceso a la configuración del usuario BE (`$BE_USER->uc`) puede ser manejado en JavaScript usando pares simples clave-valor
- Adicionalmente, [localStorage](#) de HTML5 puede usarse para almacenar datos en el navegador del usuario (lado del cliente)
- Dos nuevos objetos TYPO3 globales:
 - `top.TYPO3.Storage.Client`
 - `top.TYPO3.Storage.Persistent`
- Cada objeto tiene los siguientes métodos API:
 - `get(key)`: recoge data
 - `set(key, value)`: escribe datos
 - `isset(key)`: chequea, si key se está usando
 - `clear()`: vacía todos los datos de almacenamiento

Cambios En Profundidad

API de Almacenamiento JavaScript (2)

■ Ejemplo:

```
// get value of key 'startModule'  
var value = top.TYPO3.Storage.Persistent.get('startModule');  
  
// write value 'web_info' as key 'start_module'  
top.TYPO3.Storage.Persistent.set('startModule', 'web_info');
```

Cambios En Profundidad

Renderizado Inline de Checkboxes

- Ajuste inline de checkbox para "cols" puede usarse para renderizar checkboxes directamente uno junto al otro para reducir el total de espacio usado

```
'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'  
  )  
)  
),  
...
```

Cambios En Profundidad

Registro de Objetos de Contenido

- Se ha introducido nueva opción global para registrar y/o extender/sobreescribir cObjects tales como TEXT
- Una lista de todos los cObjects disponibles está disponible como:
`$GLOBALS['TYPO3_CONF_VARS']['FE']['ContentObjects']`
- Ejemplo: registrar un nuevo cObject EXAMPLE
`$GLOBALS['TYPO3_CONF_VARS']['FE']['ContentObjects']['EXAMPLE'] = Vendor\MyExtension\ContentObject\ExampleContentObject::class;`
- La clase registrada debe ser una subclase de
`TYPO3\CMS\Frontend\ContentObject\AbstractContentObject`
- Almacene su clase en el directorio
`typo3conf/myextension/Classes/ContentObject/`
para estar preparado para futuros mecanismos de autocarga

Cambios En Profundidad

Hooks y Señales (1)

- Se ha añadido un nuevo hook al final de `PageRepository->init()`, que permite influenciar la visibilidad de las páginas
- Registre el hook como sigue:

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

- La clase hook debe implementar la siguiente interfaz:

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


Cambios En Profundidad

Hooks y Señales (2)

- Se ha añadido un nuevo hook en el PageLayoutView para manipular el renderizado del pie de un elemento de contenido.

- Ejemplo:

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

- La clase hook debe implementar la siguiente interfaz:

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

Cambios En Profundidad

Hooks y Señales (3)

- Se ha añadido un nuevo hook como un post procesador de BackendUtility::countVersionsOfRecordsOnPage
- Éste puede usarse para visualizar estados del espacio de trabajo en el árbol de páginas por ejemplo
- Registre el hook como sigue:

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

Cambios En Profundidad

Hooks y Señales (4)

- Se ha añadido nueva señal al final del método `DataPreprocessor::fetchRecord()`
- Ésta puede usarse para manipular el array `regTableItems_data` por ejemplo, para visualizar datos manipulados en TCEForms

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

Cambios En Profundidad

Hooks y Señales (5)

- Se ha añadido una nueva señal, que permite procesamiento adicional bajo inicialización de un objeto mailer, p.e. registrando un plugin Swift Mailer

```
$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'  
);
```

Cambios En Profundidad

Múltiples UID en PageRepository::getMenu()

- Método PageRepository::getMenu() acepta ahora arrays, para definir múltiples páginas raíz

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

Capítulo 4: Extbase & Fluid

Extbase & Fluid

PaginateViewHelper

- Desde TYPO3 CMS 7.1, PaginateViewHelper acepta colecciones de entrada de los siguientes tipos:
 - QueryResultInterface
 - ObjectStorage
 - ArrayAccess
 - array

- Ejemplo:

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

ContainerViewHelper carga módulos RequireJS

- ContainerViewHelper puede cargar módulos RequireJS vía el atributo `includeRequireJsModules`
- Ejemplo:

```
<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'
  }" >
```


Método `has()` en `ObjectAccess`

- Para el uso en Fluid, `object.property` y `object.isProperty` ya soportan los siguientes métodos:
 - `isProperty()`
 - `getProperty()`
- Nuevo desde TYPO3 CMS 7.1: `hasProperty()`
- Éste llama al método `$object->hasProperty()` si se usa `object.hasProperty` en Fluid

Subir ficheros múltiples con FormUpload-ViewHelper

- FormUpload-Viewhelper soporta nuevo atributo `multiple`, que provee la opción de subir ficheros múltiples a la vez

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

Nota: desarrolladores necesitan preparar el valor de entrada para la propiedad `mapping` escribiendo su propio `TypeConverter`!

Capítulo 5: Funciones obsoletas/eliminadas

Funciones obsoletas/eliminadas

`$TYPO3_CONF_VARS[SYS][compat_version]`

- Se ha eliminado la opción `$TYPO3_CONF_VARS[SYS][compat_version]` (que fue modificada en la actualización del asistente de la Herramienta de Instalación)
- Todos los chequeos sobre `GeneralUtility::compat_version` se hacen ahora contra la constante `TYP03_branch`

Nota: ¡condiciones `TypoScript`, que chequean la vieja opción `compat_version` tienen ahora un comportamiento diferente!

Funciones obsoletas/eliminadas

Estilos inline del tag <blockquote>

- CSS Styled Content renderiza tags <blockquote> usando TypeScript `lib.parseFunc_RTE`
- Estas líneas se han eliminado sin sustitución:

```
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;"
```

- Como resultado, se han eliminado los estilos inline `"margin-bottom:0;margin-top:0;"`

Nota: estilo de tags <blockquote> posiblemente difiere tras una actualización a TYPO3 CMS 7.1.

Funciones obsoletas/eliminadas

Workspaces: campo `disable_autocreate`

- Se he eliminado el campo obsoleto `disable_autocreate` de `EXT:workspaces`
- Si una extensión TYPO3 recae en este campo, ocurre un error SQL

Funciones obsoletas/eliminadas

Funcionalidad `include_once`

- Se ha eliminado la funcionalidad para incluir ficheros PHP dentro de funciones de módulo (p.e. Info module) vía un array `include_once`
- Esto se aplica a los siguientes módulos:
 - Web => Page
 - Web => Page - New Content Element Wizard
 - Web => Functions
 - Web => Info
 - Web => Template
 - Web => Recycler
 - User => Task Center
 - System => Scheduler

Funciones obsoletas/eliminadas

Ajuste `config.meaningfulTempFilePrefix`

- En TYPO3 CMS < 7.1, nombres de ficheros de imágenes generados por el GIFBUILDER podrían ser influenciados por la opción TypoScript:
`config.meaningfulTempFilePrefix`
(GIFBUILDER usaba sólo un valor hash como el nombre de fichero)
- Se ha eliminado esta opción (nombres de ficheros en directorio `typo3temp/GB/` muestran el nombre de fichero original como el primer elemento automáticamente)

Funciones obsoletas/eliminadas

Ficheros eliminados

- Se han eliminado los siguientes **ficheros**:
 - typo3/file_edit.php
 - typo3/file_newfolder.php
 - typo3/file_rename.php
 - typo3/file_upload.php
 - typo3/show_rechis.php
 - typo3/listframe_loader.php
- Se han migrado sus funcionalidades a módulos de backend, p.e.
typo3/file_edit.php en
`BackendUtility::getModuleUrl('file_edit');`

Funciones obsoletas/eliminadas

ExtJS DatePicker

- Se he eliminado el componente ExtJS `Ext.ux.DatePicker` y reemplazado con una alternativa Twitter Bootstrap (ver capítulo "BackendUI")
- Extensiones afectadas del sistema TYPO3 CMS son `EXT:belog` o `EXT:scheduler` por ejemplo

Nota: ¡extensiones que recaen en la función obsoleta `Ext.ux.DatePicker` probablemente se romperán!

Funciones obsoletas/eliminadas

Modo de Renderizado a la Lista de Acceso

- Se ha eliminado la siguiente **variable**:
`$GLOBALS[TYPO3_CONF_VARS][BE][accessListRenderMode]`
- Campos correspondientes en tablas TCA `be_users` y `be_groups` se ajustan al valor por defecto "checkbox"
- Esto puede ser cambiado en el fichero `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';
```

Funciones obsoletas/eliminadas

Elemento de contenido "Mailform"

- Funcionalidad Mailform, que proveía el cObject FORM, ha sido eliminada del núcleo
- Todavía disponible en extensión de legado EXT:compatibility6 si se requiere
- Se han marcado las siguientes opciones como obsoletas:

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

- Se han eliminado los siguientes métodos dentro de `TypoScriptFrontendController`:

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

Funciones obsoletas/eliminadas

Funcionalidad cambiada (1)

- EXT: `indexed_search` es activada tan pronto como la extensión es instalada. Como resultado, opciones de TypoScript apropiadas `config.index_enable = 1` y `config.index_externals = 1` se vuelven activas automáticamente, también
- TSconfig `web_func.menu.wiz` cambiado a `web_func.menu.functions`
- Extensiones, que integran en la barra de herramientas superior derecha, deben implementar la nueva interfaz:
`TYPO3\CMS\Backend\Toolbar\ToolbarItemInterface`
y deben registrarse en:
`$GLOBALS['TYPO3_CONF_VARS']['BE']['toolbarItems']`

Funciones obsoletas/eliminadas

Funcionalidad cambiada (2)

- Fichero `typo3/js/tree.js` ha sido reemplazado por `EXT:backend/Resources/Public/JavaScript/LegacyTree.js` (por último está basado en jQuery)
- Variable `$GLOBALS['WEBMOUNTS']` ha sido reemplazada por `$GLOBALS['BE_USER']->returnWebmounts()`
- Se ha eliminado soporte de `.t3-table` y `.t3-button` (Clases Twitter Bootstrap implementan la apariencia visual ahora)
- Se han movido las banderas de ciudades (imágenes PNG) desde `typo3/gfx/flags/` y `typo3/sysex/t3skin/images/flags/` a: `typo3/sysex/core/Resources/Public/Icons/flags/`

Funciones obsoletas/eliminadas

Funcionalidad cambiada (3)

- Se han eliminado los templates CSS Styled Content TypoScript para TYPO3 CMS 4.5 hasta 6.1
- Los siguientes cObjects TypoScript se han movido a la extensión de legado EXT:compatibility6:

SEARCHRESULTS	COLUMNS	OTABLE	CLEARGIF
IMGTEXT	CTABLE	HRULER	

- Elemento de contenido search se ha movido a la extensión de legado EXT:compatibility6
- Las siguientes opciones del asistente TCA se han eliminado:

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

Funciones obsoletas/eliminadas

Opción TypoScript andWhere

- Opción TypoScript andWhere ha sido marcada como obsoleta
- Integradores deben usar propiedades where y markers en su lugar:

```
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
        }
    }
}
```


Funciones obsoletas/eliminadas

Puntos de entrada obsoletos

- Los siguientes puntos de entrada han sido clasificados como obsoletos:
 - typo3/tce_file.php
 - typo3/move_el.php
 - typo3/tce_db.php
 - typo3/login_frameset.php
 - typo3/sysex/cms/layout/db_new_content_el.php
 - typo3/sysex/cms/layout/db_layout.php
- En su lugar, use los siguientes:

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

Donde *<parameter>* podría ser:

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

Funciones obsoletas/eliminadas

Miscelánea (1)

- Opción TypoScript `config.xhtmlDoctype = xhtml_2` se ha marcado para ser eliminada en TYPO3 CMS 8

- Los siguientes métodos se han marcado como obsoletos:

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

- El siguiente método se ha marcado como obsoleto:

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

...y se ha reemplazado por:

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

- Propiedades `StdWrap` `textStyle` y `tableStyle` se han marcado como obsoletas

Funciones obsoletas/eliminadas

Miscelánea (2)

- Opción TypoScript `page.includeJSlibs` ha sido renombrada a `page.includeJSLibs` (mayúscula "L") y la vieja opción marcada como obsoleta
- Condición TypoScript `device` se ha marcado como obsoleta
- Método `DocumentTable::table()` ha sido marcado como obsoleto (desarrolladores deben usar Fluid para esto)
- El siguiente método se ha marcado como obsoleto:

```
TYPO3\CMS\Frontend\Controller\  
    TypoScriptFrontendController::doXHTML_cleaning()
```

...así como la opción TypoScript `config.xhtml_cleaning`

- El siguiente hook se ha marcado como obsoleto:

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

Funciones obsoletas/eliminadas

Miscelánea (3)

- Los siguientes métodos se han marcado como obsoletos:

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

- Ejecutar `contentObject->COBJECT()` se ha marcado como obsoleto (use `$cObj->cObjGetSingle('...', $conf)`; en su lugar)
- El acceso directo a `FormEngine::$renderReadonly` se ha marcado como obsoleto (use `AbstractFormElement::setRenderReadonly(TRUE)`; en su lugar)

Funciones obsoletas/eliminadas

Miscelánea (4)

- Los siguientes métodos FormEngine se han marcado como obsoletos:
 - `FormEngine::insertDefStyle`
 - `FormEngine::getAvailableLanguages()`
 - `FormEngine::sL()`
 - `FormEngine::renderVDEFDiff()`
 - `FormEngine::getLL()`
 - `FormEngine::getTSCpid()`
 - `FormEngine::getSingleField_typeFlex_langMenu()`
 - `FormEngine::getSingleField_typeFlex_sheetMenu()`
 - `FormEngine::getSpecConfFromString()`

Capítulo 6: Fuentes y Autores

Fuentes y Autores

Fuentes (1)

Noticias TYPO3:

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

Informaciones de Lanzamiento:

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

TYPO3 Bug-/Issuetracker:

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

Repositorios Git TYPO3:

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

Diapositivas TYPO3 CMS Qué hay Nuevo:

Patrick Lobacher

(Búsqueda, Recogida de Información y Versión Alemana)

Michael Schams

(Líder de Proyecto y Versión Inglesa)

Traducciones por:

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

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

Licencia bajo Creative Commons BY-NC-SA 3.0

