TYPO3 CMS 6.1

A quick summary of the features, changes and improvements
Chapter Overview

- Introduction
- Backend Changes
- TypoScript
- In-Depth Changes
- Application Programming Interface (API)
- Extbase and Fluid
- Sources
Introduction

TYPO3 CMS 6.1 “Improve”

• Focus on stabilisation of existing code/features
  • Improving communication
  • Improving contributions
  • Improving the product

• Release manager:
  Benjamin Mack
  benni.mack (at) typo3.org
  @bennimack
Introduction

TYPO3 CMS 6.1 “Improve”

- Release date: 30 April 2013
- Release timeline:

![TYPO3 CMS 6.1 Release Timeline Diagram]
Introduction

TYPO3 CMS 6.1 “Improve”

- System requirements
  - PHP: v5.3.7 – v5.4.x
  - MySQL: v5.1.x – v5.5.x

- End of maintenance: 31 October 2014

- Not a Long Term Support (LTS) release (TYPO3 CMS 6.2 will be the next version with LTS)
Introduction

TYPO3 CMS 6.1 “Improve”

- TYPO3 release agenda:
Chapter 1

Backend Changes
Backend Changes

Wizard “New Content Element”:
- Content element “Header only” added
Backend Changes

Web → Templates:

- Context menu added to template record (icon)
Backend Changes

Files → Filelist:

- Delete button added to list view

![Filelist comparison between TYPO3 < 6.1 and TYPO3 >= 6.1]
Backend Changes

cObject “FORM”:

- Confirmation page can be enabled in backend (in the past: configuration in TypoScript only)

```plaintext
enctype = multipart/form-data
method = post
prefix = tx_form
confirmation = 1
postProcessor {
    1 = mail
    [...] 
}
```
Backend Changes

Extension Manager:

- Improved usability
- Function menu for manage/get extensions
- Direct link for “Show all versions”
Backend Changes

Extension Manager:
- Improved usability
- Function menu for manage/get extensions
- Direct link for “Show all versions”
Backend Changes

Systemextension “sys_note”:
  • Re-developed in Extbase/Fluid
  • Edit button appears on mouse over
Backend Changes

Systemextension “sys_note”:

- Very easy to customise visual appearance

Constants:

```php
module.tx_sysnote {
    view {
        # cat=Internal Notes/general/1000; type=string; label=Layout Root Path: Path to layouts
        layoutRootPath = EXT:sys_note/Resources/Private/Layouts/

        # cat=Internal Notes/general/1010; type=string; label=Partial Root Path: Path to partials
        partialRootPath = EXT:sys_note/Resources/Private/Partials/

        # cat=Internal Notes/general/1020; type=string; label=Template Root Path: Path to templates
        templateRootPath = EXT:sys_note/Resources/Private/Templates/
    }
}
```
Backend Changes

Systemextension “sys_note”:
• Very easy to customise visual appearance

Setup:

```php
module.tx_sysnote {
  view {
    layoutRootPath = {$module.tx_sysnote.view.layoutRootPath}
    templateRootPath = {$module.tx_sysnote.view.templateRootPath}
    partialRootPath = {$module.tx_sysnote.view.partialRootPath}
  }
}
```
Backend Changes

Scheduler:

- Re-worked overview of tasks in Scheduler
- Execute a task with one click

![Scheduler Comparison](image.png)
Backend Changes

Scheduler:

- Task “System Status Update”
- Multiple email addresses are possible now

TYPO3 CMS 6.1 - What's New
Backend Changes

System Environment Check:

• Re-development of system checks
  • All checks separated in a “standalone script”
  • Independent to TYPO3
• Module “Reports” accesses and shows check results

⚠️ System

⚠️ System environment check 2 Test(s)
The system environment check returned warnings. Those errors might have a negative affect to the functionality and stability of your TYPO3 CMS instance. Please check the install tool “System environment” for all details.
Backend Changes

System Environment Check:

- New section in Install Tool
  - Green: ok
  - Yellow: warning
  - Red: fatal error
Chapter 2

TypoScript
TypoScript

strPad:

• New stdWrap function “strPad” pads a string to a certain length with another string or character

```typo3
page.10 = TEXT
page.10.value = HELLO WORLD!
page.10.strPad {
  length = 15
  padWith = .
  type = left
}
```

Output:

```none
...HELLO WORLD!
```
TypoScript

disableBodyTag:

- Disable HTML `<body>` tag generated by TYPO3 core
  
  ```typo3conf/config.yaml
  disableBodyTag = 1
  ```

- For example useful, if `<body>` should be created (and customised) by a template engine such as TemplaVoilá
TypoScript

FLUIDTEMPLATE – cObject:

- Source of templates for FLUIDTEMPLATE can be cObjects now (previously: only files)

Example 1: cObject “FILE”

```typo3
page.10 = FLUIDTEMPLATE
page.10 {
    template = FILE
    template.file = fileadmin/templates/MyTemplate.html

    partialRootPath = fileadmin/templates/partial/
    variables {
        mylabel = TEXT
        mylabel.value = Hello World!
    }
}
```
TypoScript

FLUIDTEMPLATE – cObject:

- Source of templates for FLUIDTEMPLATE can be cObjects now (previously: only files)

Example 2: cObject “TEXT”

```plaintext
page.10 = FLUIDTEMPLATE
page.10 {
    template = TEXT
    template.value = <h1>{data.title}</h1>
    partialRootPath = fileadmin/templates/partial/
    variables {
        mylabel = TEXT
        mylabel.value = Hello World!
    }
}
```
TypoScript

FLUIDTEMPLATE – Settings Array:

- Pass an array of values to FLUIDTEMPLATE by using the key word “settings”
- Constants are also supported

```typo3
lib.globalSettings {
  foo = value
  bar = {$bar}
}

devpage = PAGE
devpage.10 = FLUIDTEMPLATE
devpage.10 {
  file = fileadmin/templates/MyTemplate.html
  settings < lib.globalSettings
}

Access values in template file:
{settings.foo} and {settings.bar}
```
TypoScript

splitChar:

- Defines “split” character for noTrimWrap in HMENUs (default is: “|”)

```typo3ts
temp.mainMenu = HMENU
temp.mainMenu {
    [...]
    10 = TMENU
    10.NO {
        [...]
        10 = TEXT
        10 {
            text.field = title

            # line below does not work (rendered as optionSplit):
            # text.noTrimWrap = | -- | |

            text.noTrimWrap = * -- * *
            text.noTrimWrap.splitChar = *

            [...]
        }
    }
}
```
Typoscript

TSconfig - Module “List”:

- Disable links to export data in module “List”

  mod.web_list.noExportRecordsLink = 1
TypoScript

Backend Layout Configuration:

- Include external files for backend layout configuration

```
<INCLUDE_TYPOSCRIPT: source="FILE:fileadmin/layout.ts"/>
```
TypoScript

Miscellaneous

• New functions to manipulate comma-separated values: `uniqueList()`, `reverseList()` and `sortList()`

```typo3
type = PAGE
page.10 = TEXT

page.10.value = 11,14,11,7
// output: 11,14,11,7

page.10.value := uniqueList()
// output: 11,14,7

page.10.value := reverseList()
// output: 7,14,11

page.10.value := sortList(numeric,descending)
// output: 14,11,7

page.10.value := sortList(numeric)
// output: 7,11,14
```
TypoScript

Miscellaneous

- Remove all CSS generated via \_CSS\_PAGE\_STYLE by extensions (e.g. css_styled_content)
  
  ```
  config.removePageCss = 1
  ```

- Option pageTitleSeparator can be used with StdWrap functions
  
  ```
  config.pageTitleSeparator = -
  config.pageTitleSeparator.noTrimWrap = | | |
  ```

- User TSconfig “clearRTECache” renamed

  OLD:   options.clearCache.clearRTECache
  NEW:   options.clearCache.clearLangCache

This is a breaking change!
TypoScript

Miscellaneous

- Default values of cObject FORM can be used with stdWrap functions

- Add tags to pages (stored in database table “cache_pages”)
  
  ```
  stdWrap.addPageCacheTags = pagetag1,pagetag2,pagetag3
  ```

- Remove page tags by TSconfig
  
  ```
  TCEMAIN.clearCacheCmd = tag:pagetag1,tag:pagetag2,tag:pagetag3
  ```
Chapter 3

In-Depth Changes
In-Depth Changes

TCA Handling:

- Table Configuration Array (TCA) handling partly re-developed
- TCA is always loaded and cached in FE
- TCA is loaded in BE and option dynamicConfigFile became redundant (no differences between 'ctrl' and 'columns')
- Extension developers should store the TCA file at:  
  EXT:extensionkey/Configuration/TCA/tablename.php
  ...which returns an array of table “tablename”
In-Depth Changes

TCA Handling:

- Sections 'ctrl' can be removed from file ext_tables.php

- Functions/methods became obsolete:
  - TSFE->includeTCA()
  - TSFE->getCompressedTCAarray()
  - GeneralUtility::loadTca()

- Fallback for extensions which do not follow the new TCA scheme in typo3conf/ext_tables.php

- Directory t3lib/stdbb removed
  (all files have been implemented in the TYPO3 core now)
In-Depth Changes

TCA – Field Display Conditions:

- Option `displayCond` supports multiple conditions and 'AND' and 'OR' now

```
'link_file' => array (  
    'exclude' => 0,  
    'label' => 'LLL:EXT:filelinks/locallang_db.xml:tx_filelinks.link_file',  
    'config' => array (  
        [...]  
    ),  
    'displayCond' => array(  
        'AND' => array(  
            'FIELD:testField:>:9',  
            'FIELD:testField:<:11',  
        ),  
    )  
)
```
In-Depth Changes

RequireJS:
- JavaScript file and module loader
- Integrated in the TYPO3 core
- Available in FE and BE
- Supports Asynchronous Module Definition (AMD) (mechanism for defining modules such that the module and its dependencies can be asynchronously loaded)
- jQuery is already loaded via AMD, named “jQuery”
In-Depth Changes

RequireJS:

- Example for developers:

```javascript
loadRequireJsModule(TYPO3/CMS/PackageName/ModuleName);
```

...would load module:

```javascript
EXT:PackageName/Resources/Public/JavaScript/ModuleName.js
```
In-Depth Changes

Database:

- Main DB connection class uses "mysqli" instead of the old "mysql" PHP extension
- "Lazy" database connection implemented
  - DB connection will be established by database connection class on demand
  - Bootstrap does not take care of initialisation
  - eID scripts do not need to call `connectDB()` any more
In–Depth Changes

PHP Opcode Caches:

- XCache cache backend added to caching framework
  http://xcache.lighttpd.net/wiki/Introduction

- Checks added to Install Tool:
  - Zend Optimizer Plus
  - Zend OPcache
  - wincache
In-Depth Changes

Miscellaneous

- Domain records are cached, when typolink generates links, if the following option has been set:
  
  \[
  \text{config\.typolinkCheckRootline} = 1
  \]

- Images from the Rich Text Editor (RTE) moved from: \text{uploads/RTEmagicC_*} to: \text{fileadmin/_migrated_/RTE/}

When updating to TYPO3 6.1, the update wizard moves the files and adds them to the storage index.
In-Depth Changes

Removed Components:

- Systemextension “statictemplates” removed from TYPO3 core
  Install Tool → Upgrade Wizard imports and installs the extension from TER if required

- Flash library “swfupload” and Flash Uploader interface removed
  The more robust HTML5 file uploader (introduced in TYPO3 6.0) works perfectly fine
Chapter 4

Application Programming Interface (API)
Application Programming Interface (API)

Database API:

- Method `searchQuery()` supports “AND” and “OR” now
- 4th parameter controls constraint (default: AND)

```php
\TYPO3\CMS\Core\Database\DatabaseConnection

class DatabaseConnection {

    public function searchQuery(
        $searchWords,
        $fields,
        $table,
        $constraint = self::AND_Constraint
    ) { // or: self::OR_Constraint
        [...]
    }
}
```
Application Programming Interface (API)

Pre- and Post SELECT Hooks:

- See class:
  
  \TYPO3\CMS\Core\Database\DatabaseConnection

- New pre-process-hook for SELECT statements
  
  public function SELECTquery_preProcessAction(...) { ... }

- New post-process-hook for SELECT statements
  
  public function exec_SELECTquery_postProcessAction(...) { ... }
Application Programming Interface (API)

**CSS Styled Content Rendering Hook:**

- See class:
  
  \TYPO3\CMS\CssStyledContent\Controller\CssStyledContentController

- New hook in CSS Styled Content allows custom rendering of a media element

  \$GLOBALS['TYPO3_CONF_VARS']['EXTCONF']...[\'css_styled_content\'][\'pi1_hooks\'][\'render_singleMediaElement\']

- Useful for FAL content such as videos or content embedded by JavaScript
Chapter 5

Extbase and Fluid
Extbase and Fluid

ClassNamingUtility:

- New class offers several functions related to naming and conversions of names

\TYPO3\CMS\Core\Utility\ClassNamingUtility

- Methods:

  static public translateModelNameToRepositoryName($modelName) { ... }
  static public translateModelNameToValidatorName($modelName) { ... }
  static public translateRepositoryNameToModelName($repositoryName) { ... }
  static public explodeObjectControllerName($controllerObjectName) { ... }
Extbase and Fluid

Extbase – Recursive Storage PID:

- Set storage PID recursively via TypoScript (default value: 0)

```typo3
tx_extension {
persistence {
    storagePid = 1,2,3
    recursive = 99
}
}
```
Extbase and Fluid

Extbase – Overwrite Object Mapping:

• Overwrite object mapping via TypoScript

```php
config.tx_extbase {
  objects {
    Tx_Extbase_Persistence_BackendInterface {
      className = Tx_MyExt_Persistence_Backend
    }
  }
}

...also works with plugin now:

plugin.tx_extension {
  objects {
    Tx_Extbase_Persistence_BackendInterface {
      className = Tx_MyExt_Persistence_Backend
    }
  }
}
```
Extbase and Fluid

Extbase – beforeCallActionMethod:

- Signal (hook) that is triggered prior execution of current action (callActionMethod)

```php
$this->signalSlotDispatcher->dispatch(
    __CLASS__,
    'beforeCallActionMethod',
    array(
        'controllerName' => get_class($this),
        'actionMethodName' => $this->actionMethodName,
        'preparedArguments' => $preparedArguments
    )
);
```
Extbase and Fluid

Extbase – beforeCallActionMethod:

- Signal (hook) that is triggered prior execution of current action (callActionMethod)
- Slot usage example:

```php
$this->getSignalSlotDispatcher()->connect(
    '\TYPO3\CMS\Extbase\Mvc\Controller\ActionController',
    'beforeCallActionMethod',
    'Tx_MyExtension_CallActionSlot',
    'beforeCallActionMethod'
);
```
Extbase and Fluid

Extbase – Property Mapper:

- New (rewritten) Property Mapper ready for production environments (has been shipped with TYPO3 CMS since 4.6)
- Feature is enabled by default now

\[
\text{config.tx_extbase.features.\texttt{rewrittenPropertyMapper}} = 1
\]

- Use TYPO3 Flow documentation as a reference

Extbase and Fluid

Extbase – TYP03_MODE detection:

- New service class to determine if FE or BE context

```php
\TYPO3\CMS\Extbase\Service\EnvironmentService

public function isEnvironmentInFrontendMode() {
    return (defined('TYPO3_MODE') && TYPO3_MODE === 'FE') ? : FALSE;
}

public function isEnvironmentInBackendMode() {
    return (defined('TYPO3_MODE') && TYPO3_MODE === 'BE') ? : FALSE;
}
```
Extbase and Fluid

Extbase – TYP03_MODE detection:

• New service class to determine if FE or BE context

• Usage example:

```php
/**
 * @var \TYPO3\CMS\Extbase\Service\EnvironmentService
 */
protected $environmentService;

/**
 * @param \TYPO3\CMS\Extbase\Service\EnvironmentService $environmentService
 * @return void
 */
public function injectEnvironmentService(
    \TYPO3\CMS\Extbase\Service\EnvironmentService $environmentService)
{
    $this->environmentService = $environmentService;
}
... if ($this->environmentService->isEnvironmentInFrontendMode()) { ... } ...
Extbase and Fluid

Extbase – FlashMessage Handling:

- FlashMessages can be added to a queue (before: set via a “container” only)
- Enable this feature with TypoScript
  config.tx_extbase.enableLegacyFlashMessageHandling = 1
- Usage Example:
  // set a FlashMessage
  $this->controllerContext->getFlashMessageQueue()->addMessage(
      new \TYPO3\CMS\Core\Messaging\FlashMessage(
          $flashMessage, '', \TYPO3\CMS\Core\Messaging\FlashMessage::ERROR
      )
  );

  // read and empty queue
  $this->controllerContext->getFlashMessageQueue()->getAllMessagesAndFlush();
Extbase and Fluid

Extbase – Performance:

- Performance increase in various areas
  - Deletion of page cache
  - Internal caching layer for `getTargetPidByPlugin`
  - Persistence of ObjectStorages optimised
Extbase and Fluid

Extbase – Property Mapper Validation:

- Object validation of Property Mapper reworked
  - Prevent storing invalid objects if validation fails
  - Modified objects have to be marked as such by executing the repository's update() function

```php
{the domain objects repositories}->update({modified domain object})
$this->blogRepository->update($blog);
```

- Side effect: performance boost
Extbase and Fluid

Extbase – Miscellaneous:

- Single Table Inheritance (STI) for users/groups removed from default TypoScript setup

  To demonstrate STI in Extbase, `tx_extbase_type` was implemented for `fe_users` and `fe_groups`. Due to various problems, this has been removed again.

- Deprecated: `ObjectManager::create` (use `ObjectManager::get instead)
Extbase and Fluid

Fluid - Pagination ViewHelper:

- Pagination ViewHelper for use in the BE

```php
<f:be.widget.paginate
    objects="{blogs}"
    as="paginatedBlogs"
    configuration="{
        itemsPerPage: 5, insertAbove: 1,
        insertBelow: 0, recordsLabel: 'MyRecords'}">

    /* use {paginatedBlogs} instead of {blogs}
    for example inside a <f:for> loop */

</f:be.widget.paginate>
```
Extbase and Fluid

Fluid – String Compare:

• Compare strings (e.g. with the if-ViewHelper, see example below)

```html
<f:if condition="\{foo\} == 'bar'">
    /* do something... */
</f:if>
```
Extbase and Fluid

Fluid – Property Mapping Configuration:

- Property Mapping relies on a whitelist (trusted properties) now
- Extbase takes care of Fluid forms automatically (current level only, not nested objects)

\TYPO3\CMS\Extbase\Property\PropertyMappingConfiguration

/* Set all options for the given $typeConverter: */
public setTypeConverterOptions($typeConverterClassName, array $options) { ... }

/* Allow a list of specific properties: */
public allowProperties() { ... }

/* Allow all properties in property mapping, even unknown ones: */
public allowAllProperties() { ... }

/* Allow all properties, but reject (blacklist) selected ones: */
public allowAllPropertiesExcept() { ... }
Extbase and Fluid

Fluid – Arrays in ViewHelper Arguments:

- Arrays are processed in ViewHelper arguments now

This is a breaking change!

- TYPO3 < 6.1:
  
  `<f:debug>{key1: 'value1', key2: 'value2'}</f:debug>`

- TYPO3 >= 6.1:
  
  `<f:debug value="{key1: 'value1', key2: 'value2'}" />`
Extbase and Fluid

Fluid – Currency ViewHelper:

- Currency ViewHelper supports decimal points now

```xml
<f:format.currency
    currencySign='$'
    decimalSeparator='.'
    thousandsSeparator=','
    prependCurrency='TRUE'
    separateCurrency='FALSE'
    decimals='2'>1234.5678</f:format.currency>
```

Output: $1,234.56
Extbase and Fluid

Fluid – Access to File References:

- Access to file references works “on the fly” now, by creation of a FAL resource object

{myEntity.fileProperty.originalResource.title}

{myEntity.fileProperty.originalResource.properties.identifier}
Sources

- **TYPO3 News**
  - http://typo3.org/news

- **Release Notes**
  - http://wiki.typo3.org/TYPO3_6.1
  - NEWS.txt

- **TYPO3 GIT Repositories**
  - https://git.typo3.org/TYPO3v4/Core.git
  - https://git.typo3.org/TYPO3v4/CoreProjects/MVC/extbase.git
  - https://git.typo3.org/TYPO3v4/CoreProjects/MVC/fluid.git
Sources

- TYPO3 Bug-/Issuetracker

- RequireJS and AMD
  - http://requirejs.org
  - https://github.com/amdjs/amdjs-api/wiki/AMD

- Further Extbase/Fluid documentation
  - http://docs.typo3.org/flow/TYPO3FlowDocumentation/
    TheDefinitiveGuide/PartIII/PropertyMapping.html
    28091_dirty_objectStorage_improvements

- Download TYPO3 CMS
  - http://typo3.org/download
During TYPO3 CMS 6.1 development phase the 10.000th core merge was done!

Thanks to all contributors :-)
Authors of this Presentation

Patrick Lobacher
typovision GmbH
Munich/Germany
http://www.typovision.de

Michael Schams
Cerebrum (Aust) Pty Ltd
Melbourne/Australia
http://schams.net

Credits also go to: Ben, Roland and Den for review and quality assurance.
Licensed under Creative Commons BY-NC-SA 3.0
Thank You!

TYP03 inspires people to share