Field rearrangement

T3UXW14 – Festenburg

Team members

- Patrick Broens
- Felix Kopp
- Mathias Schreiber

Goal

Since the T3UXW14 is all about user experience, we are focussing on a clean backend by default, regarding TCEFORMS. The main user groups are editors and beginners. Cleanup "under the hood" is also considered for backend configuration and frontend output.

Problems

- Beginners are overwhelmed with the amount of fields displayed, especially in their main task - creating and editing content. They will start as administrator when trying TYPO3 CMS for the first time.
- Database contains unused fields for backwards compatibility.
- Fields do not have a proper use case.
- Non logical grouping of fields in tabs and palettes.
- Fields are misused.
  Layout field is used to change functionality.
- No logical field functionality.
  Target field is a combination of an input field and a dropdown with one value – New window. Since frames are not in use anymore, this can be changed to a simple checkbox.
- Extend field functionality for better usability.
  Add functionality where it will help the editor, like autocomplete.
- Too much fields for styling of frontend output.
  tt_content contains a lot of fields for changing styles. These fields should be centralized.
- Mixing database fields and FlexForms are not considered best practice.
- Labels contain the wrong information or are not descriptive.
- Language labels for fields are taken from several core extensions.
- There is no clear distinction between core and css_styled_content.
  The core is providing the database fields for css_styled_content, where it should only provide the basic fields like uid, pid, creation date and user, language and versioning.

- Redundant content elements.
  Functionality of some content elements is provided by other extensions, like forms and login. There is no proper use case for the search content element, since mostly it will be added to a page with TypoScript. The search content element does not work with all search extensions in the TER.
- css_styled content is hard to maintain.

Solutions

Content elements (core / tt_content / css_styled_content)

Separate core and css_styled_content

Make it possible to replace css_styled_content with another extension providing content elements without cluttering database and unused definitions and methods. It 's becoming normal for companies to use only their own content elements. Therefor the current fields are redundant. We can imagine there can be extensions providing content elements adapted for mobile devices or other purposes.

Keep only the necessary basic properties for Content Elements:

- Authoring (user, tstamp, time)
- Hierarchy (position, sorting, parent page)
- Visibility (frontend user access, access time)
- Channel (language, workspace)

Tasks

- Move tt_content table definition to css_styled_content, except for fields mentioned above.
- Move relevant TCA to css_styled_content. Keep the basic properties in "frontend".
- Move language labels, used in TCA, to css_styled_content.
- Move user functions, used in TCA, to css_styled_content.
- Move images, used in TCA, to css_styled_content.
- Move the content preview for content elements in tx_cms_layout::tt_content_drawItem to css_styled_content

Remove content elements

- Form
  Forms are provided as TER extensions. The system extension "form" will be moved to TER.
- Search
  Only provides search field for indexed_search and should be done with TypoScript or provided by search extension.
- Login
  Provided by the extension fe_login

Field changes in tt_content table

Name	Action	Comments
accessibility_bypass	hide	
accessibility_bypass_text	hide	
accessibility_title	hide	
altText	deprecate	Superseded by FAL
cols	change / hide	change label to "table_columns" supply migration
date	hide	
editor_information	new	This field is used to supply informational instructions for editor. The field will receive a migration so that hidden headers will have their content transfered info the field Editor Information The field will add a marker to the page module indicating there is actual information. It will be shown on mouseover, onclick on-something...
header_layout	change	Remove option “Hidden”. The context is gone when a header is hidden. Checkout field “editor_information”
header_position	deprecate	Should be done with “layout” field since the position of the header is within the scope of the content element layout
image	to be done	See Benni's “big MM plan”
imageborder	deprecate	Should be done by CSS style defined in the layout field of the CE
imagecaption	deprecate	Provided by FAL
imageheight	deprecate	
imageorient	change	The following options can be done with multiple content elements, so should be hidden: Above right Above center Above left Below right Below center Below left
imagewidth	deprecate	Superseded by image_field_percentage
imagecaption_position	deprecate	Should be done by CSS style defined in the layout field of the CE
image_compression	deprecate	
image_effects	deprecate	
image_frames	deprecate	Should be done by CSS style defined in the layout field of the CE
image_link	deprecate	Superseded by FAL
image_noRows	deprecate	Too expensive to calculate

Name	Action	Comments
		Rendering needs explanation / not intuitive
image_width_percentage	new	Evaluates against 1-100%
layout	hide	Change type to multiselect in order to supply multiple CSS classes Values can be added on CType basis Only keep the value "default"
linkToTop	hide	Move to “Appearance” tab
longdescURL	deprecate	Should be done by FAL (@TODO)
media	to be done	See Benni's “big MM plan”
multimedia	Supply migration	
rte_enabled	deprecate	Should be set on a global level (either instance or user)
sectionIndex	hide	
section_frame	deprecate	Should be done by layout field
spaceBefore	deprecate	
spaceAfter	deprecate	
subheader	hide	Hide in all CE's Used for mailform, should be moved to new databasefield
table_bgColor	deprecate	
table_border	deprecate	
table_caption	new / hide	Instead of FlexForm String
table_cellpadding	deprecate	
table_cellspacing	deprecate	
table_csv_delimiter	new / hide	Instead of FlexForm Dropdown with predefined values
table_csv_enclosure	new / hide	Instead of FlexForm Dropdown with predefined values
table_header_position	new / hide	Instead of FlexForm Dropdown with predefined values
table_tfoot	new / hide	Instead of FlexForm Checkbox
target	change	Label becomes "open in new window"
text_align	remove	Not in use
text_color	remove	Not in use
text_face	remove	Not in use
text_properties	remove	Not in use
text_size	remove	Not in use
titleText	deprecate	Superseded by FAL
uploads_description	new	Boolean to show description
uploads_type	new	Dropdown:

Name	Action	Comments
		Only file name File name and file extension icon File name and thumbnail (if possible)

Legend:

- Deprecate: Remove from TCA. Delete in later version
- Change: Change in functionality or label
- New: New field
- Remove: Remove immediately from database
- Hide: Hide field in TCEFORM by default

FlexForms

FlexForms are not considered best practice. It mixes regular database fields with XML structures. They also add extra tabs, so we get nested tabs.

FlexForm fields should be moved to regular database fields as mentioned above.

The content elements “Table” and “Media” are using FlexForms.

Provide modern content elements

css_styled_content is heavily based on TypoScript, which is hard to understand for integrators. The current controller is not according to our standards and hard to maintain.

A new extension should be provided using FLUIDTEMPLATE for easy adaptions.

Above we are mentioning that tt_content fields should be moved to css_styled_content. The new extension can define new field names, which are according to our standards.

Pages

Field changes in tt_content table

Name	Action	Comments
abstract	hide	
alias	remove	Remove static url's and embrace EXT: realurl
author	hide	
author_email	hide	
cache_tags	hide	
cache_timeout	hide	
extendToSubpages	change	Should be moved underneath Access / User rights Rename field in database extend_to_subpages and keep Change label to "Extend User group Access Rights to subpages" Form central palette to Access Somewhat unclear what is based on this field in the wild
keywords	hide	
lastUpdated	hide	
layout	discuss	Frontend layout: Who would like to drop this field and rename backend_layout to Pages Layout?
meta_robot_index	new / hide	Dropdown: None Index No index
meta_robot_follow	new / hide	Dropdown: None Follow No follow
module	change	Move below Pages type into palette "Type". Should only be visible on a folder where records are stored. Rename to "Contains records for Module XXX" or "Folder Type"
newUntil	remove	Not in use
no_cache	hide	
php_tree_stop	discuss	Broken functionality. See https://forge.typo3.org/issues/61812
storage_pid	discuss	Use should be discouraged Most extensions are already using storage with specific configuration Records for the page should be within the page and not a remote folder

Name	Action	Comments
subtitle	hide	
target	discuss / hide	For an editor this text field is unclear
TSconfig	change / hide	Rename label and make sure that there is no "normal" TypoScript allowed. This is the configuration language (MANAGE tier, not PRESENT tier)
url_scheme	hide	

Legend:

- Deprecate: Remove from TCA. Delete in later version
- Change: Change in functionality or label
- New: New field
- Remove: Remove immediately from database
- Hide: Hide field in TCEFORM by default
- Discuss: Should be discussed with active contributors