

t3_origuid decision

t3_origuid field has ambiguous meaning, we need to decide what to do with it.

Options:

1. provide new definition, adapt core code according to it
2. deprecate the field provide new field

What is t3_origuid (origUid)

Official documentation

OrigUid

<https://docs.typo3.org/typo3cms/TCAResource/Reference/Control/Index.html#origuid> (old link)

<https://docs.typo3.org/m/typo3/reference-tca/master/en-us/Control/Index.html?highlight=origuid#origuid> (new link)

Field name, which will contain the UID of the original record in case a record is created as a copy **or new version** of another record.

Is used when new versions are created from elements and enables **the backend to display a visual comparison between a new version and its original.**

~~versioning_followPages - this field was removed in 7~~

~~https://docs.typo3.org/typo3cms/TCAResource/singlehtml/#versioning_followpages~~

~~Description~~

~~(Only for other tables than "pages")~~

~~If set, content from this table will get copied along when a new version of a page is created.~~

~~Tracking Originals~~

~~It is highly recommended to use the "**origUid**" feature for tables whose records are copied with pages that are versioned with content or subtree since this will enable the possibility of **content comparison between current and future versions.**~~

My investigation

One of the usages of this field was page/branch versioning, which is long gone. The content comparison mentioned in the official docu is also not present in the core for the long time (was it implemented at all??).

Current usage

Currently in the core it is used just in few places related to translation wizard. The idea behind it is to use `t3_origuid` as a field pointing to translation source even when record is not in connected mode (has no `l10n_parent`).

Another appliance is to know the translation source in case a record is translated from non-default language. In most cases the field works in a way it can be used for translation mode purposes, however there are scenarios (workspaces, complex copy) when it contains unexpected value.

The field is currently filled with in following **scenarios**

1. when copying record (set to uid of the source record)
2. when translating record (set to translation source)
3. when modifying/versioning a record (delete, modify, sorting, move) in the draft workspace (set to uid of the modified live record)

Problems

1. You will get different `t3_origuid` value depending on whether you edit record in live or edit and publish through workspaces (scenario 3)
2. `t3_origuid` is not set for `sys_file_reference` on copying
3. In some places `t3_origuid` name is hardcoded, TCA `'origUid'` setting should be used instead
4. when you copy a `tt_content` which has translations 297, to another page, It will be copied together with translated versions. The translated&copied record (300) will have `l10n_parent` set to record copied&InDefaultLanguage, but `t3_origuid` set to copy source. <https://forge.typo3.org/issues/78599>

Before copy:

uid	pid	sys_language_uid	l10n_parent	t3_origuid	header
297	89	0	0	0	Regular Element #1
298	89	1	287	287	[Translate to Dansk:]Regular Element #1

After copy:

uid	pid	sys_language_uid	l10n_parent	t3_origuid	header
297	89	0	0	0	Regular Element #1

298	89	1	287	287	[Translate to Dansk:] Regular Element #1
299	90	0	0	0	Regular Element #1
300	90	1	299	298	[Translate to Dansk:] Regular Element #1

However if you copy a record before translation, and then translate it, it will have different t3_origuid.

Solution 1 - adapt the definition

New definition - origUid

Field name, which will contain the UID of the original record in case a record is created as a copy or translation of another record.

This field will have the same value as `transOrigPointerField` when a record is translated from a record in the default language. However if a record was translated from a record in non-default language the value of `origUid` (source record uid) will be different than `transOrigPointerField` (uid of the record in the default language).

The field has the same value independently on whether it was modified directly in live workspace or was modified in draft and then published.

Question: What should t3_origuid contain If a record was copied together with its translation (e.g. copying tt_content with translation to another page) ??

uid	pid	sys_language_uid	l18n_parent	t3_origuid	l10n_source	header
2	89	0	0	0	0	Regular Element #1
3	89	1	2	2	2	[Translate to Dansk:] Regular Element #1
4	89	2	2	3	2	[Translate to German:] [Translate to Dansk:] Regular Element #1
5	90	0	0	?? / 2	0	Regular Element #1 (copy of 2)

6	90	1	5	?? / 3	5	[Translate to Dansk:] Regular Element #1
7	90	2	5	?? / 4	5	[Translate to German:] [Translate to Dansk:] Regular Element #1

Solution 2 - introduce a new field

Provide a new field called `translationSource` in v8 and a new TCA setting

`\$TCA[\$tableName][ctrl][translationSource]`.

Which will behave in a following way:

- It will contain an uid of the source record in case a record is translated (if translation source is in the default language, l10n_parent === l10n_source)
- The field has the same value independently on whether it was modified directly in live workspace or was modified in draft and then published.
- In connected mode, when you copy a tt_content ID2 which has translations(2,3), to another page (90), It will be copied together with translated versions. The translated and copied record (6) will have l10n_parent and l10n_source set to copied record in the default language (5) because it's translated from a record in the default language.
- **Record (7) will have l10n_source set to (6) because it's a copy of the record (4) which was translated from a record in non-default language (3). So the copied record (7) will also be marked as translated from language 1 (record 6).**

Connected mode:

uid	pid	sys_language_uid	l18n_parent	l10n_source	(t3_origuid)	header
2	89	0	0	0	0	Regular Element #1
3	89	1	2	2	2	[Translate to Dansk:] Regular Element #1
4	89	2	2	3	3	[Translate to German:] [Translate to Dansk:] Regular Element #1
5	90	0	0	0	2	Regular Element #1 (copy of ID 2)
6	90	1	5	5	3	[Translate to Dansk:] Regular Element #1 (copy of

						ID 3)
7	90	2	5	6	4	[Translate to German:] [Translate to Dansk:] Regular Element #1

Free mode:

uid	pid	sys_language_uid	l18n_parent	l10n_source	header
2	89	0	0	0	Regular Element #1
3	89	1	0	2	[Translate to Dansk:] Regular Element #1
4	89	2	0	3	[Translate to German:] [Translate to Dansk:] Regular Element #1

Action points:

1. Don't bother about t3_origuid any more
2. Define the behaviour of the new field
3. Add the new field in v8
 - a. db & TCA definition
 - b. support in datahandler
 - c. functional tests coverage (live, workspaces, relations)
4. Rewrite the translation wizard to use it
5. (Deprecate t3_origuid in v8)
6. Find a solution for v7
 - a. "fix" functionality of chained processing in t3_origuid - like we did in master
 - b. use \$TCA[<table>][ctrl][localizationOrigUid] as trigger - not defined in 7.6
 - c. provide an extension (e.g. friends of typo3) that introduces l10n_source database fields and enables \$TCA[<table>][ctrl][localizationOrigUid] in TCA-Overrides for localizable core tables

Discussion

Chained processing of localize and copy on for a record overrides `t3_origuid` and thus the meaning of a partial record

- * this constraint can never be resolved, since localize is internally handles as copy and thus re-uses application logic to define `t3_origUid`
- * using mixed values for `t3_origUid` like `11-14`, is not feasible since this adds more complexity and field type has to be changed (`integer` to `string`)
- * in any way, chaining the format or the meaning of `t3_origUid` is heavily breaking for third party extensions
- * introducing a new field `l10n_source` (or something similar) is the only way to keep a distinct meaning

changing `t3_origUid` is too risky and still we don't know what other extensions/applications are doing with the information in that field. thus, fiddling around with that makes it even harder to keep the database model in a proper integrity. thus, we should aim for introducing a new `l10n_source` field for the purpose of localize and the copyToLanguage actions...

for v7 i see two options:

- 1) remove the functionalities that are currently not working (the broken features)
- 2) provide an extension that fixes the behaviour by using hooks - can be installed on demand
- [3) fix it in v7 as well and introduce the new field - however, changing database in stable branch is not encouraged]

... or maybe another addition to the first option:

- * remove the functionalities that are currently not working
- * reimplement the behavior of master, which is only available if `$TCA[$tableName]['ctrl']['translationSource']` is set
- * defining `l10n_source` in database and \$TCA might be done with an extension

this way we could have similar code in v7 and v8, side-effects are removed from v7. if somebody relies on the feature in v7, it can be added by manually installing `EXT:whatever_localization`

in v9 we hopefully can introduce event-sourcing and avoid these kind of integrity problems with the directly persisted states

Voting (put your name in the column):

Solution 1	Solution 2	I Don't have opinion/I don't care
	Markus Klein	
	Mathias Schreiber	
	Susi Moog	

	Frans Saris	
	Benni Mack	
	Tymoteusz Motylewski	
	loli	
	Olly	